

ArcGIS® 9.1

Geoprocessing Commands Quick Reference Guide



Copyright © 2004, 2005 ESRI
All rights reserved.
Printed in the United States of America.

The information contained in this document is the exclusive property of ESRI. This work is protected under United States copyright law and other international copyright treaties and conventions. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or by any information storage or retrieval system, except as expressly permitted in writing by ESRI. All requests should be sent to Attention: Contracts and Legal Services Manager, ESRI, 380 New York Street, Redlands, CA 92373-8100, USA.

The information contained in this document is subject to change without notice.

CONTRIBUTING WRITERS

Melanie Harlow, Ghislain Prince, Catherine Jones, Corey Tucker, Jeff Reinhart

U.S. GOVERNMENT RESTRICTED/LIMITED RIGHTS

Any software, documentation, and/or data delivered hereunder is subject to the terms of the License Agreement. In no event shall the U.S. Government acquire greater than RESTRICTED/LIMITED RIGHTS. At a minimum, use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in FAR §52.227-14 Alternates I, II, and III (JUN 1987); FAR §52.227-19 (JUN 1987) and/or FAR §12.211/12.212 (Commercial Technical Data/Computer Software); and DFARS §252.227-7015 (NOV 1995) (Technical Data) and/or DFARS §227.7202 (Computer Software), as applicable. Contractor/Manufacturer is ESRI, 380 New York Street, Redlands, CA 92373-8100, USA.

ESRI, ArcView, ArcGIS, ArcInfo, ArcCatalog, ArcToolbox, ArcSDE, www.esri.com, 3D Analyst, and ArcEditor are trademarks, registered trademarks, or service marks of ESRI in the United States, the European Community, or certain other jurisdictions.

Other companies and products mentioned herein are trademarks or registered trademarks of their respective trademark owners.






Table of Contents

Introduction	1
Analysis Toolbox	3
Cartography Toolbox	7
Conversion Toolbox	9
Coverage Toolbox	13
Data Interoperability Toolbox	21
Data Management Toolbox	23
Geocoding Toolbox	37
Geostatistical Analyst Toolbox	39
Linear Referencing Toolbox	41
Network Analyst Toolbox	43
Spatial Analyst Toolbox	47
Spatial Statistics Toolbox	65
3D Analyst Toolbox	69
Index	75

Introduction

This reference guide is designed to provide an easy and quick reference for those wishing to use the ESRI® command language at the ArcGIS® command line and for those writing scripts.

All commands are referred to as tools, scripts, or models and are maintained in toolsets within the ArcGIS toolboxes.

-  A toolbox can contain tools, toolsets, and scripts and is organized according to the collection of geoprocessing commands it contains.
-  A toolset can contain tools, toolsets, and scripts and is organized according to the geoprocessing commands it contains.
-  A tool is a single geoprocessing command.
-  A script is a set of instructions usually stored in a file and interpreted, or compiled, at run time.
-  A model consists of one process or, more commonly, multiple processes strung together.

This guide describes the following toolboxes:

Analysis Tools
Cartography Tools
Conversion Tools
Coverage Tools
Data Management Tools
Data Interoperability Tools
Geocoding Tools

Geostatistical Analyst Tools
Linear Referencing Tools
Network Analyst Tools
Spatial Analyst Tools
Spatial Statistics Tools
3D Analyst™ Tools

Each toolbox contains a list of the toolsets and tools as they are organized within ArcToolbox™.

The Index section at the end of this guide contains an alphabetical list of each tool, script, toolset, and toolbox.

All tools are available with the ArcInfo™ license or the extension they are associated with. However, many are available for use with ArcView® or ArcEditor™ (sometimes with limited functionality). Those available with ArcView and ArcEditor are denoted with a ❖, and those available with ArcEditor are denoted with a ◆.

Some tools, such as Clip, exist in multiple toolboxes. Therefore, an alias can be added as a suffix to the tool name when more than one toolbox is available. An example of an alias usage is clip_arc, where clip is the tool and arc is the suffix representing the Coverage Toolbox, or clip_analysis, where the suffix represents the Analysis Toolbox.

The alias list

Analysis Tools	_analysis
Cartography Tools	_cartography
Conversion Tools	_conversion
Coverage Tools	_arc
Data Management Tools	_management
Geocoding Tools	_geocoding
Geostatistical Analyst Tools	_ga
Linear Referencing Tools	_lr
Network Analyst Tools	_na
Spatial Analyst Tools	_sa
Spatial Statistics Tools	_stat
3D Analyst Tools	_3d

The syntax of an example tool

```
Uni on_arc <i n_cover> <uni on_cover> <out_cover> {fuzzy_tolerance}
{JOIN | NO_JOIN}
```

Where

Union_arc is the tool; the following components are the parameters.

<> indicates required parameters.

{ } indicates optional parameters; these do not need to be included. One can be skipped using # if you need to apply only a portion of them.

The | indicates mutually exclusive arguments, and only one of the arguments in the list of options can be specified.

In some commands, there may be an ellipsis between two arguments, such as item1...item4. This indicates that you can give one or more (up to 4 in this example) names or values for that argument.

Example

```
Uni on_arc Treepol ycov Newtreecov Fi nal treecov # JOIN
```



Analysis Toolbox

A suite of geoprocessing tools used to solve spatial or statistical problems.



Extract Toolset

Contains tools used to manipulate data into manageable datasets containing only the desired features and attributes.

- ❖ **Clip:** extracts those features from an input feature class that overlap with features from a clip feature class.

```
Clip <input_features> <clip_features> <output_feature_class> {cluster_tolerance}
```

- ❖ **Select:** extracts selected features from an input feature class or layer and stores them in the output feature class.

```
Select <input_features> <output_feature_class> {where_clause}
```

Split: clips the input features and stores them in multiple output datasets.

```
Split <input_features> <split_features> <split_field> <output_workspace>  
{cluster_tolerance}
```

- ❖ **Table Select:** extracts selected attributes from an input table or table view and stores them in an output table.

```
TableSelect <input_table> <output_table> {where_clause}
```



Overlay Toolset

Contains tools for topological integration of features based on symmetry.

Erase: copies input features falling outside the erase polygon feature boundaries to the output.

```
Erase <input_features> <erase_features> <output_feature_class> {cluster_tolerance}
```

Identity: intersects two feature classes. The output contains the input features as well as those overlapping features of the identity feature class.

```
Identity <input_features> <identity_features> <output_feature_class>  
{ALL | NO_FID | ONLY_FID} {cluster_tolerance}  
{NO_RELATIONSHIPS | KEEP_RELATIONSHIPS}
```

- ❖ **Intersect:** creates an output feature class containing features that fall within the area common to both input datasets.

```
Intersect <features{Ranks}; features{Ranks}... > <output_feature_class>  
{ALL | NO_FID | ONLY_FID} {cluster_tolerance} {INPUT | LINE | POINT}
```

Symmetrical Difference: creates an output feature class containing features or portions of features common only to one of the inputs.

```
SymDifference <input_features> <update_features> <output_feature_class>
{ALL | NO_FID | ONLY_FID} {cluster_tolerance}
```

❖ **Union:** creates an output feature class containing all features from both inputs.

```
Union <features {Ranks}; features {Ranks}... > <output_feature_class>
{ALL | NO_FID | ONLY_FID} {cluster_tolerance} {GAPS | NO_GAPS}
```

Update: updates the attributes and geometry of the input using the update feature class or layer they overlap.

```
Update <input_features> <update_features> <output_feature_class>
{BORDERS | NO_BORDERS} {cluster_tolerance}
```



Proximity Toolset

Contains tools to determine spatial relationships among features, with respect to the distance relationships between features.

❖ **Buffer:** creates buffer polygons to a specified distance around the input features.

```
Buffer <input_features> <output_feature_class> <buffer_distance_or_filed>
{FULL | LEFT | RIGHT} {ROUND | FLAT} {NONE | ALL | LIST}
{dissolve_filed; dissolve_filed...}
```

🌐 ❖ **Multiple Ring Buffer:** creates a new feature class of buffer features using a set of buffer distances.

```
MultiRingBuffer <input_features> <output_feature_class>
<distances; distances... > {DEFAULT | CENTIMETERS | DECIMALDEGREES | FEET |
INCHES | KILOMETERS | METERS | MILES | MILLIMETERS | NAUTICALMILES | POINTS |
YARDS} {filed_name} {ALL | NONE}
```

Near: computes the distance from each point in the input to the nearest feature in the near feature class or layer.

```
Near <input_features> <near_features> {search_radius} {NO_LOCATION | LOCATION}
{NO_ANGLE | ANGLE}
```

Point Distance: computes the distance between each point in a feature class or layer to all points in a different feature class or layer.

```
PointDistance <input_features> <near_features> <output_table> {search_radius}
```




Statistics Toolset

Contains tools that perform standard statistical analysis on attribute data.

Frequency: calculates frequency statistics for field(s) in a table.

```
Frequency <i n_table> <out_table> <frequency_fields; frequency_fields...>  
{summary_fields; summary_fields...}
```

❖ **Summary Statistics:** calculates summary statistics for field(s) in a table.

```
Statistics <i n_table> <out_table> <fields{Statistical Type}; fields{Statistical  
Type}...> {case_fields}
```




Cartography Toolbox

Contains tools designed to produce data for maps to meet specific cartographic standards.



Masking Toolset

Contains tools to construct masking polygons for use with variable depth masking.

Cul-de-Sac Masks: creates a feature class of polygon masks from a symbolized input line layer.

```
Cul deSacMasks <i nput_l ayer> <output_feature_cl ass> <reference_scal e>
<spati al _reference> <margi n> {ONLY_FI D | NO_FI D | ALL}
```

Feature Outline Masks: creates mask polygons at a specified distance and shape around the symbolized features in the input layer.

```
FeatureOutl i neMasks <i nput_l ayers> <output_feature_cl ass> <reference_scal e>
<spati al _reference> <margi n> <CONVEX_HULL | BOX | EXACT_SI MPLI FI ED | EXACT>
<ALL_FEATURES | ONLY_PLACED> {ONLY_FI D | NO_FI D | ALL}
```

Intersecting Layers Masks: creates masking polygons at a specified shape and size at the intersections of symbolized input layers.

```
Intersecti ngLayersMasks <maski ng_l ayer> <masked_l ayer> <output_feature_cl ass>
<reference_scal e> <spati al _reference> <margi n> <CONVEX_HULL | BOX |
EXACT_SI MPLI FI ED | EXACT> <ALL_FEATURES | ONLY_PLACED>
{ONLY_FI D | NO_FI D | ALL}
```




Conversion Toolbox

Contains tools that are used to convert data into various formats.



From Raster Toolset

Contains tools to output raster datasets to other formats.

❖ **Raster to ASCII:** converts a raster dataset to an ASCII file representing raster data.

`RasterToASCII <i n_raster> <out_ascii_file>`

❖ **Raster to Float:** converts a raster dataset into a file of binary floating point values representing raster data.

`RasterToFloat <i n_raster> <out_float_file>`

❖ **Raster to Point:** converts a raster dataset to a point feature dataset.

`RasterToPoint <i n_raster> <out_point_features> {raster_filed}`

❖ **Raster to Polygon:** converts a raster dataset to a polygon feature dataset.

`RasterToPolygon <i n_raster> <out_polygon_features> {SIMPLIFY | NOSIMPLIFY} {raster_filed}`

❖ **Raster to Polyline:** converts a raster dataset to a polyline feature dataset.

`RasterToPolyline <i n_raster> <out_polyline_features> {ZERO | NODATA} {minimum_dangle_length} {SIMPLIFY | NOSIMPLIFY} {raster_filed}`



To CAD Toolset

Contains tools to assist in the mining of CAD drawings for the purpose of populating new or existing geodatabase feature classes.

Add CAD Fields: adds fields to the input table by selecting from groups of CAD-specific fields, which have the appropriate name and type recognized by the Export CAD tool.

`AddCADFields <i nput_table> <ADD_ENTITY_PROPERTIES | NO_ENTITY_PROPERTIES> {ADD_LAYER_PROPERTIES | NO_LAYER_PROPERTIES} {ADD_TEXT_PROPERTIES | NO_TEXT_PROPERTIES} {ADD_DOCUMENT_PROPERTIES | NO_DOCUMENT_PROPERTIES} {ADD_XDATA_PROPERTIES | NO_XDATA_PROPERTIES}`

Create CAD XData: creates a table formatted to be recognized by the Export CAD tool as AutoCAD Extended Entity Data.

`CreateCADXData <i n_table> <fields; fields... > <RegApp> <ADE | TRADITIONAL>`

Export to CAD: exports features from a feature class to one or more CAD drawings.

`ExportCAD <i n_features; i n_features... > <DWG-R2000 | DGN-V8 | DWG-R14 | DXF-R14 | DXF-R2000 | DWG-R2004 | DXF-R2004> <Output_File> {USE_FILENAMES_IN_TABLES | IGNORE_FILENAMES_IN_TABLES} {OVERWRITE_EXISTING_FILES | APPEND_TO_EXISTING_FILES} {Seed_File}`

Set CAD Alias: renames one or more existing field name aliases by matching columns from the input table with a list of CAD-specific fields of appropriate names, which are recognized by the Export CAD tool.

SetCADAlias <input_table> <field_info>



To Coverage Toolset

Contains tools to convert any supported feature class format to a coverage.

Feature Class to Coverage: creates a single coverage from one or more input feature classes or layers.

FeatureClassToCoverage <features{Type}; features{Type}... > <out_cover>
{cluster_tolerance} {DOUBLE | SINGLE}



To dBASE Toolset

Contains tools to convert tables into a dBASE® format.

Table to dBASE: converts INFO™, OLE DB, or geodatabase tables to dBASE tables.

TableToDBASE <input_tables; input_tables... > <output_folder>



To Geodatabase Toolset

Contains tools to convert any supported vector or raster data type to a geodatabase.



Feature Class to Feature Class: copies a feature class into a geodatabase or to a shapefile.

FeatureClassToFeatureclass <input_features> <output_location>
<output_feature_class_name> {expression} {field_info} {SAME_AS_TEMPLATE |
DISABLED | ENABLED} {SAME_AS_TEMPLATE | DISABLED | ENABLED}
{configuration_keyword} {first_spatial_grid}



Feature Class to Geodatabase (multiple): copies one or more feature classes or layers to a geodatabase feature class.

FeatureclassToGeodatabase <input_features; input_features... >
<output_geodatabase>

Import CAD Annotation: converts a collection of CAD annotation features into a geodatabase annotation feature class.


ImportCADAnnotations <input_features; input_features... >
<output_feature_class> <reference_scale> {CLASSES_FROM_LEVELS |
ONE_CLASS_ONLY} {NO_MATCH | MATCH_FIRST_INPUT} {NO_SYMBOL_REQUIRED |
REQUIRE_SYMBOL} {STANDARD | FEATURE_LINKED} {linked_feature_class}
{AUTO_CREATE | NO_AUTO_CREATE} {AUTO_UPDATE | NO_AUTO_UPDATE}

- ❖ **Import Coverage Annotation:** imports coverage annotations into a geodatabase annotation feature class.


```
ImportCoverageAnnotations <input_features; input_features. . . >
<output_feature_class> <reference_scale> {CLASSES_FROM_LEVELS |
ONE_CLASS_ONLY} {NO_MATCH | MATCH_FIRST_INPUT} {NO_SYMBOL_REQUIRED |
REQUIRE_SYMBOL} {STANDARD | FEATURE_LINKED} {linked_feature_class}
{AUTO_CREATE | NO_AUTO_CREATE} {AUTO_UPDATE | NO_AUTO_UPDATE}
```

- ❖ **Import from CAD:** imports from one or more CAD files to a geodatabase.


```
ImportCAD <input_files; input_files. . . > <out_personal_geodatabase>
{spatial_reference} {DO_NOT_EXPLODE_COMPLEX | EXPLODE_COMPLEX}
```

-  ❖ **Raster to Geodatabase (multiple):** loads multiple raster datasets into a geodatabase or raster catalog.

```
RasterToGeodatabase <input_rasters; input_rasters. . . > <output_geodatabase>
{configuration_keyword}
```

-  ❖ **Table to Geodatabase (multiple):** converts dBASE, INFO, or OLE DB tables to geodatabase tables and copies tables from one geodatabase to another.

```
TableToGeodatabase <input_table; input_table. . . > <output_geodatabase>
```

-  ❖ **Table to Table:** converts or copies dBASE, INFO, OLE DB, or geodatabase tables to a dBASE or geodatabase table.

```
TableToTable <input_table> <output_location> <output_table_name> {expression}
{field_info} {configuration_keyword}
```

To Raster Toolset

Contains tools to convert any supported raster format to either a GRID, ERDAS IMAGINE®, TIFF, or geodatabase format.

- ❖ **ASCII to Raster:** converts an ASCII file representing raster data into a raster dataset.

```
ASCIIToRaster <in_ascii_file> <out_raster> {INTEGER | FLOAT}
```

- ❖ **DEM to Raster:** converts a USGS DEM file into a raster dataset.


```
DEMToraster <in_dem_file> <out_raster> {FLOAT | INTEGER} {z_factor}
```

- ❖ **Feature to Raster:** converts a feature dataset to a raster dataset.

```
FeatureToRaster <in_features> <field> <out_raster> {cell_size}
```

- ❖ **Float to Raster:** converts a file of binary floating point values representing raster data into a raster dataset.

```
FloatToRaster <in_float_file> <out_raster>
```


 ❖ **Raster to Other Format (multiple):** converts one or more ArcGIS supported raster dataset formats to a GRID, IMAGINE, or TIFF format.

RasterToOtherFormat <input_rasters; input_rasters. . . > <output_workspace>
{GRID | IMAGINE IMAGE | TIFF}



To Shapefile Toolset

Contains tools to create shapefiles from feature classes.

 ❖ **Feature Class to Shapefile (multiple):** exports one or more feature classes to shapefiles in a designated folder.

FeatureClassToShapefile <input_feature_class; input_feature_class. . . >
<output_folder>



Coverage Toolbox

Contains the original ArcInfo Workstation commands used to perform geoprocessing tasks with coverages.



Analysis Toolset

Contains tools and toolsets used for geospatial processing.



Extract Toolset

Contains tools used to select features or parts of features to create a new coverage.

Clip: extracts, using a cookie-cutter method, those features or portions of features from an input coverage that overlap with a clip coverage polygon.

```
Clip <i n_cover> <clip_cover> <out_cover> {POLY | LINE | POINT | NET | LINK | RAW}
{fuzzy_tolerance}
```

Select: extracts map features from the input coverage and stores them in an output coverage, based on logical expressions or by applying the criteria contained in a selection file.

```
Reselect <i n_cover> <out_cover> <i nfo_express; i nfo_express. . . >
{POLY | LINE | POINT | ANNO. subclass | ROUTE. subclass | SECTI ON. subclass |
REGI ON. subclass} {sel ecti on_fi le} {out_feature_type}
```

Split: clips portions of the input coverage into multiple coverages.

```
Split <i n_cover> <split_cover> <split_item> <path> {POLY | LINE | POINT | NET |
LINK | RAW} {fuzzy_tolerance}
```



Overlay Toolset

Contains tools used to calculate the various options when overlaying two coverages.

Erase: erases the input coverage features or portions of features that overlap with the erase coverage polygons.

```
Erase <i n_cover> <erase_cover> <out_cover> {POLY | LINE | POINT | NET | LINK | RAW}
{fuzzy_tolerance}
```

Identity: computes the geometric intersection of two coverages, where all features of the input coverage and only those overlapping from the identity coverage are preserved.

```
Identity <i n_cover> <i denti ty_cover> <out_cover> {POLY | LINE | POINT}
{fuzzy_tolerance} {JOI N | NO_JOI N}
```

Intersect: computes the geometric intersection of two coverages, where only those features in the area common to both coverages will be preserved.

```
Intersect <i n_cover> <i ntersect_cover> <out_cover> {POLY | LINE | POINT}
{fuzzy_tolerance} {JOI N | NO_JOI N}
```

Union: computes the geometric intersection of two polygon coverages. All polygons from both coverages will be split at their intersections and preserved in the output coverage.

```
Union <i n_coverage> <union_coverage> <out_coverage> {fuzzy_tolerance}
{JOIN | NO_JOIN}
```

Update: replaces the input coverage areas with the update coverage polygons using a cut-and-paste type of operation.

```
Update <i n_coverage> <update_coverage> <out_coverage> {POLY | NET} {fuzzy_tolerance}
{KEEP_BORDER | DROP_BORDER}
```



Proximity Toolset

Contains tools used in geoprocessing analysis involving distance.

Buffer: creates buffer polygons around specified input coverage features.

```
Buffer <i n_coverage> <out_coverage> {LINE | POLY | POINT | NODE} {buffer_item}
{buffer_table} {buffer_distance} {fuzzy_tolerance} {ROUND | FLAT} {FULL | LEFT
| RIGHT}
```

Near: computes the distance from each point in a coverage to the nearest arc, point, or node in another coverage.

```
Near <i n_coverage> <near_coverage> <out_coverage> {LINE | POINT | NODE} {search_radius}
{NO_LOCATION | LOCATION}
```

Point Distance: computes the distances between point features in one coverage to all points in a second coverage that are within the specified search radius.

```
PointDistance <from_coverage> <to_coverage> <out_info_table> {search_radius}
```

Point Node: transfers attributes from a point feature class to a node feature class.

```
PointNode <point_coverage> <node_coverage> {search_radius}
```

Thiessen: converts a point coverage to a coverage of Thiessen or proximal polygons.

```
Thiessen <i n_coverage> <out_coverage> {proximal_tolerance}
```



Conversion Toolset

Contains tools and toolsets to convert a coverage to or from another file format.



From Coverage Toolset

Contains tools used to convert a coverage into various file formats.

Export to DLG: converts a coverage to an Optional Digital Line Graph (DLG-3) file format.

```
ArcDLG <i n_coverage> <out_dlg_file> {i n_point_coverage} {i n_projection_file}
{x_shift} {y_shift} {i n_header_file} {TRANSFORM | NO_TRANSFORM}
```

Export to Interchange File: converts a coverage to an Interchange file.

```
Export <COVER | FONT | GRID | INFO | LINESET | MAP | MARKERSET | PLOT | SHADESET |  
STACK | STACKALL | TEXT | TEXTSET | TIN> <i_n_dataset> <i_n_interchange_file> {NONE |  
PARTIAL | FULL} {max_lines}
```

Export to S57: converts a coverage to an S-57 object format.

```
ArcS57 <i_n_workspace> <i_log_file> {out_workspace}
```

Export to SDTS: converts a coverage or grid to an SDTS Topological Vector Profile (TVP) or Point Profile Transfer file.

```
SDTSEXP <TVP | POINT | RASTER> <i_n_dataset> <out_transfer_prefix>  
{i_n_point_cover} {out_DD_transfer} {conversion_control_file}
```

Export to VPF: converts a coverage into either a VPF coverage or VPF tile.

```
VPFEXP <i_n_cover> <out_file> {tile_name} {control_file} {EXTRA | NO_EXTRA}  
{NO_FIT | FIT}
```

Ungenerate: Creates a text file of xy coordinates from a coverage.

```
Ungenerate <i_n_cover> <out_generate_file> <LINE | POINT | POLY | TIC | LINK |  
REGION.subclass | ANNO.subclass> {NODES | NO_NODES} {EXPONENTIAL | FIXED}
```



To Coverage Toolset

Contains tools used to convert various file formats into coverages.

Advanced Tiger Conversion: performs the Basic Tiger Conversion, followed by advanced operations including joining, defining a projection, and building topology.

```
TigerTool <i_n_tiger_file_prefix> <out_cover_prefix> {NO_JOIN | JOIN} {UTM |  
STATE} {zone_number} {1995 | 1997 | 1998 | 1999 | 2000 | 2002} {NO_RESTART |  
RESTART}
```

Basic Tiger Conversion: converts U.S. Bureau of Census TIGER/Line™ files into one or more coverages.

```
TigerArc <i_n_tiger_file_prefix> <out_cover> {out_point_cover}  
{out_landmark_cover} {1995 | 1997 | 1998 | 1999 | 2000 | 2002}
```

Generate: creates a coverage from raw coordinates stored in a text file.

```
Generate <i_n_file> <out_cover> <LINES | ANNOTATIONS | CIRCLES | CURVES | FISHPNET |  
LINKS | POINTS | POLYGONS | TICS>
```

Import from DLG: converts a Standard or Optional formatted Digital Line Graph (DLG) file into a coverage.

```
DLGArc <i_n_dlg_file> <out_cover> {out_point_cover} {NOFIRST | ALL | ATTRIBUTED}  
{x_shift} {y_shift} {category}
```

Import from Interchange File: converts an interchange file into a coverage.

```
Import <AUTO | COVER | FONT | GRID | INFO | LINESET | PLOT | MAP | MARKERSET |  
SHADESET | STACK | TEXT | TEXTSET | TIN> <interchange_file> <out_dataset>
```

Import from S57: converts data from an S-57 file format to a coverage.

```
S57Arc <in_s57_file> <out_workspace> {CLEAN | NO_CLEAN}
```

Import from SDTS: creates coverages or grids from an SDTS Topological Vector Profile (TVP) or Point Profile Transfer file.

```
SDTSImport <in_transfer_prefix> <output> {out_point_cover} {layer_name}  
{DD | DROP_DD} {PRESERVE | CONVERT}
```

Import from VPF: converts a VPF table into an INFO table or converts a VPF coverage or tile into a coverage.

```
VPFImport <input_vpf> <output> {tile_name} {control_file} {NO_EXTRA | EXTRA}
```



Data Management Toolset

Contains tools and toolsets to manage, manipulate, and maintain coverages and their attribute tables.



Aggregate Toolset

Contains tools used to combine coverages.

Append: combines an unlimited number of coverages into one coverage.

```
Append <in_covers; in_covers. . . > <out_coverage> {FEATURES_ONLY |  
FEATURES_ATTRIBUTES} {POLY | LINE | POINT | NODE | NET | LINK | ANNO. subclass |  
SECTION. subclass | ROUTE. subclass | REGION. subclass}  
{NO | TICS_ONLY | FEATURES_ONLY | FEATURES_TICS}
```



Composite Features Toolset

Contains tools to create or convert regions within a coverage and to convert line features to routes.

Line Coverage to Region: converts arcs to preliminary regions in a new or existing coverage or appends preliminary regions to an existing region subclass.

```
RegionClass <in_coverage> {out_coverage} <out_subclass> {in_region_item}  
{out_region_item} {selection_file} {MULTIRING | SINGLERING}
```

Line Coverage to Route: creates a route system by creating whole arc sections for each arc in the input coverage. It can also be used to append arcs to an existing route system.

```
ArcRoute <in_coverage> <out_route_system> {in_route_item} {out_route_item}  
{measure_item} {UL | UR | LL | LR} {BLANK | NO_BLANK}
```

Polygon Coverage to Region: converts a polygon coverage to a region subclass. Each polygon in the `in_cover` becomes a region of the output subclass.

`Pol yRegi on <i n_cover> <out_cover> <out_subcl ass>`

Region to Polygon Coverage: converts a region subclass into a polygon coverage and creates an INFO table containing overlapping region information.

`Regi onPol y <i n_cover> <out_cover> <i n_subcl ass> {out_tabl e}`



Generalization Toolset

Contains tools to derive data with less detail and complexity from coverage features.

Aggregate Polygons: combines disjointed and adjacent polygons into new area features based on a distance.

`Aggr ega tePol ygo ns <i n_cover> <out_cover> <cel l_si ze> <di stance>
{NON_ORTHO GONAL | ORTHO GONAL}`

Collapse Dual Lines to Centerline: derives centerlines (single lines) from dual-line features, such as road casings, based on specified width tolerances.

`Col l apseDual Li neToCenter Li ne <i n_cover> <out_cover> <maxi mum_wi dth>
{mi ni mum_wi dth}`

Dissolve: merges adjacent polygons, lines, or regions that have the same value for a specified item.

`Di ssol ve <i n_cover> <out_cover> <di ssol ve_i tem>
{POL Y | LI NE | NET | REGI ON. subcl ass}`

Eliminate: merges selected polygons with neighboring polygons that have the largest shared border between them or the largest area.

`El i mi na te <i n_cover> <out_cover> <i nfo_express; i nfo_express. . . >
{NO_KEE P_EDGE | KEE P_EDGE} {POL Y | LI NE} {sel ecti on_fi l e} {BOR DER | AREA}`

Find Conflicts: searches a region coverage for overlapping and closely spaced buildings, based on a specified distance, and records the occurrences.

`Fi ndConfl i cts <i n_cover> <out_cover> <confl i ct_di stance>`

Simplify Building: simplifies the boundary or footprint of building polygons while maintaining their essential shape and size.

`Si mpl i fyBui l di ng <i n_cover> <out_cover> <si mpl i fi cati on_tol erance>
{mi ni mum_area} {sel ecti on_fi l e} {NOT_CHECK | CHECK_CONFL I CT}`

Simplify Line or Polygon: removes small fluctuations or extraneous bends from a line or polygon, while preserving its essential shape.

`Si mpl i fyLi neOrPol ygo n <i n_cover> <out_cover> <si mpl i fi cati on_tol erance>
{POI NT_REMOVE | BEND_SI MPLI FY} {NO_ERROR_CHECK | ERROR_CHECK}`



Indexes Toolset

Contains tools to add or remove attribute indexes.

Drop Index: drops an attribute index from the specified item and INFO table.

`DropIndex <i n_info_table> {index_item; index_item...}`

Index Item: creates an attribute index to increase access speed to the specified item during query operations.

`IndexItem <i n_info_table> <index_item>`



Items Toolset

Contains tools to add or remove items (fields) in INFO tables.

Add Item: adds a blank or zero item to an INFO table.

`AddItem <i n_info_table> <out_info_table> <item_name> <item_width>
<output_width> <BINARY | CHARACTER | DATE | FLOATING | INTEGER | NUMERIC>
{decimal_places} {start_item}`

Drop Item: deletes one or more items from an INFO table.

`DropItem <i n_info_table> <out_info_table> <drop_item; drop_item...>`



Joins Toolset

Contains a tool to join INFO tables.

Join Info Tables: joins the item definitions and values of two tables based on a shared item.

`JoinItem <i n_info_table> <join_info_table> <out_info_table> <related_item>
{start_item} {LINEAR | ORDERED | LINK}`



Projections Toolset

Contains tools to set a projection or reproject or transform a coverage.

Define Projection: creates or modifies the coordinate system information (including projection parameters, such as datum and spheroid) stored in the coverage's projection definition file (.prj).

`DefineProjection <i n_cover> <projection_file>`

Project: changes the coordinate system of a coverage including its datum or spheroid.

`Project <i n_cover> <out_cover> <projection_file>`

Transform: moves all features in the coverage based on a set of from and to control points.

`Transform <i n_cover> <out_cover> {AFFINE | PROJECTIVE | SIMILARITY}`



Tables Toolset

Contains tools used for editing the associated attribute tables.

Add XY Coordinates: calculates and adds xy coordinates of labels or points to the coverage PAT or xy coordinates of nodes to the coverage NAT.

AddXY <i n_cover> {POINT | NODE}

Renumber Nodes: updates arc–node topology by renumbering nodes for coverage arcs and identifies arcs that share the same node locations.

Renode <i n_cover> {from_item} {to_item}

Update IDs: updates User-IDs in a coverage after they have been modified in a feature attribute table.

IDedit <i n_cover> <POLY | LINE | POINT | ANNO_subclass>



Tolerances Toolset

Contains a tool to adjust coverage-associated tolerances.

Tolerance: sets the tolerances associated with a coverage.

Tolerance <i n_cover> {FUZZY | DANGLE | TIC_MATCH | EDIT | NODESNAP | WEED | GRAIN | SNAP} {tolerance_value}



Topology Toolset

Contains tools used to develop the topologic relationship within a coverage.

Build: creates or updates feature attribute tables and polygon topology.

Build <i n_cover> <POINT | LINE | POLY | NODE | ANNO> {anno_subclass}

Clean: generates a coverage with correct polygon or arc–node topology.

Clean <i n_cover> {out_cover} {dangle_length} {fuzzy_tolerance} {POLY | LINE}

Create Labels: creates label points for polygons that have no labels and assigns each a User-ID.

CreateLabels <i n_cover> {id_base}

VPF Tile Topology: creates cross-tile topology for all tiled coverages in a Vector Product Format (VPF) database library or creates topology for an individual tile in a VPF library.

VPFTile <VPF_Library> {sig_digits} {93 | 96} {ALL | VPF_cover}



Workspace Management Toolset

Contains a tool to manage coverages within a workspace.

Create Coverage: creates a new, empty coverage.

Create <out_cover> {template_cover}



Data Interoperability Toolbox

Contains tools to import and export data with the ArcGIS Data Interoperability extension.

Quick Export: Converts one or more input feature classes or feature layers into any format supported by the ArcGIS Data Interoperability extension.

`QuickExport <Input; Input... > <Output>`

Quick Import: Converts data from any format supported by the ArcGIS Data Interoperability extension into feature classes..

`QuickImport <Input> <Output>`



Data Management Toolbox

Contains the tools to develop, manage, and manipulate feature classes, datasets, and layers.



Database Toolset

Contains tools that improve database performance.

❖ **Compact:** reduces the size and optimizes the performance of a personal geodatabase.

`Compact <i n_workspace>`

◆ **Compress:** removes all unreferenced database states and redundant rows in a version.

`Compress <i n_workspace>`



Disconnected Editing Toolset

Contains tools to check out or update data from a shared data system.

◆ **Check In:** checks datasets back in to the parent ArcSDE® geodatabase from a checked out ArcSDE or personal geodatabase.

`CheckIn <i n_workspace> <dest_workspace> {NON_RECONCI LE | RECONCI LE}`

◆ **Check In from Delta:** checks in a geodatabase from a delta geodatabase or an XML file. A delta geodatabase contains only the changes exported from a check-out geodatabase.

`CheckInDel ta <i n_del ta_database> <dest_workspace>
{NON_RECONCI LE | RECONCI LE}`

◆ **Check Out:** checks out data from an ArcSDE geodatabase to an ArcSDE or personal geodatabase for offline editing.

`CheckOut <i n_data; i n_data. . . > <out_workspace> <out_name>
<DATA | SCHEMA_ONLY> <NO_REUSE | REUSE> <RELATED | NO_RELATED>`

◆ **Export to Delta:** exports changes in a check-out geodatabase to a delta database or XML file. A delta geodatabase contains changes exported from a check-out geodatabase.

`ExporttoDel ta <i n_workspace> <dest_del ta_database>`



Domains Toolset

Contains tools for the management of domains, both coded and attribute, within a workspace.

❖ **Add Coded Value to Domain:** adds a new value to a domain's coded value list.

`AddCodedVal ueToDomai n <i n_workspace> <domai n_name> <code> <code_descri pti on>`

❖ **Assign Domain to Field:** sets the domain for a particular field and optionally for a subtype.

`Assi gnDomai nToFi el d <i n_tabl e> <fi el d_name> <domai n_name>
{subtype_code; subtype_code. . . }`

❖ **Create Domain:** creates an attribute domain in the specified workspace.

```
CreateDomain <i_n_workspace> <domain_name> <domain_description>  
<SHORT | LONG | FLOAT | DOUBLE | TEXT | DATE> {CODED | RANGE}  
{DEFAULT | DUPLICATE | GEOMETRY_RATIO} {DEFAULT | SUM_VALUES | AREA_WEIGHTED}
```

❖ **Delete Coded Value from Domain:** removes a value from a coded value domain.

```
DeleteCodedValueFromDomain <i_n_workspace> <domain_name> <code; code... >
```

❖ **Delete Domain:** deletes a domain from a workspace.

```
DeleteDomain <i_n_workspace> <domain_name>
```

❖ **Domain to Table:** creates a table from an attribute domain.

```
DomainToTable <i_n_workspace> <domain_name> <output_table> <code_field>  
<description_field> {configuration_keyword}
```

❖ **Remove Domain from Field:** removes an attribute domain association from a feature class field.

```
RemoveDomainFromField <i_n_table> <field_name> {subtype_code; subtype_code... }
```

❖ **Set Value for Range Domain:** sets the minimum and maximum values for an existing range domain.

```
SetValueForRangeDomain <i_n_workspace> <domain_name> <min_value> <max_value>
```

❖ **Table to Domain:** creates or updates a coded value domain from a table.

```
TableToDomain <i_n_table> <code_field> <description_field> <i_n_workspace>  
<domain_name> <domain_description> <APPEND | REPLACE>
```



Feature Class Toolset

Contains tools for basic feature class management including appending and integrating multiple feature classes.

❖ **Append Annotation Feature Classes:** combines annotation from multiple input feature classes to create a new annotation feature class.

```
AppendAnnotation <input_features; input_features... > <output_feature_class>  
<reference_scale> {CREATE_CLASSES | ONE_CLASS_ONLY}  
{NO_SYMBOL_REQUIRED | REQUIRE_SYMBOL} {AUTO_CREATE | NO_AUTO_CREATE}  
{AUTO_UPDATE | NO_AUTO_UPDATE}
```

❖ **Calculate Default Cluster Tolerance:** calculates a default cluster tolerance value.

```
CalculateDefaultClusterTolerance <i_n_features>
```

❖ **Calculate Default Spatial Grid Index:** calculates a spatial grid value, used to quickly locate features in a dataset that match the criteria of a spatial search.

```
CalculateDefaultGridIndex <i_n_features>
```

❖ **Create Feature Class:** creates a new feature class.

```
CreateFeatureClass <out_path> <out_name> <POLYGON | POINT | MULTIPOINT |  
POLYLINE> {template; template...} {DISABLED | SAME_AS_TEMPLATE | ENABLED}  
{DISABLED | SAME_AS_TEMPLATE | ENABLED} {spatial_reference}  
{configuration_keyword} {spatial_grid_1} {spatial_grid_2} {spatial_grid_3}
```

❖ **Integrate:** compares one or more feature classes and makes any lines, points, or vertices within a specified distance range identical or coincident.

```
Integrate <features{Ranks}; features{Ranks}... > {cluster_tolerance}
```

❖ **Update Annotation Feature Class:** updates the input annotation feature class with text attribute fields and optionally populates the value of each new field for every feature in the feature class.

```
UpdateAnnotation <i_n_features> {POPULATE | DO_NOT_POPULATE}
```



Features Toolset

Contains tools to manage and enrich feature classes, such as inspecting and correcting potential errors with data and creating different geometries.

❖ **Add XY Coordinates:** adds the field POINT_X and POINT_Y for labels, points, tics, or nodes to the input feature class.

```
AddXY <i_n_features>
```

❖ **Check Geometry:** checks the validity of the geometries of features.

```
CheckGeometry <i_n_features; i_n_features... > <out_table>
```

❖ **Copy Features:** copies the selected features to a new feature class.

```
CopyFeatures <i_n_features> <out_feature_class> {configuration_keyword}  
{spatial_grid_1} {spatial_grid_2} {spatial_grid_3}
```

❖ **Delete Features:** deletes features from the input features.

```
DeleteFeatures <i_n_features>
```

Feature Envelope to Polygon: creates polygons from the envelopes of each feature in the input features.

```
FeatureEnvelopeToPolygon <i_n_features> <out_feature_class>  
{SINGLEPART | MULTIPART}
```

Feature to Line: creates an output line feature class from input polygon features.

```
FeatureToLine <i_n_features; i_n_features... > <out_feature_class>  
{cluster_tolerance} {ATTRIBUTES | NO_ATTRIBUTES}
```

Feature to Point: creates a point feature class from the centroids or midpoints of the input features.

`FeatureToPoint <i_n_features> <out_feature_class> {CENTROID | INSIDE}`

Feature to Polygon: creates a new polygon feature class from input line features.

`FeatureToPolygon <i_n_features; i_n_features... > <out_feature_class> {cluster_tolerance} {ATTRIBUTES | NO_ATTRIBUTES} {Label_features}`

Feature Vertices to Points: creates points from the vertex locations of the input features.

`FeatureVerticesToPoints <i_n_features> <out_feature_class> {ALL | MID | START | END | BOTHENDS}`

❖ **Multipart to Singlepart:** breaks any multipart features into single features.

`MultipartToSinglepart <i_n_features> <out_feature_class>`

Polygon to Line: generates a new line feature class from input polygon features.

`PolygonToLine <i_n_features> <out_feature_class>`

❖ **Repair Geometry:** repairs geometry problems in a feature class or layer.

`RepairGeometry <i_n_features>`

Split Line at Vertices: splits a feature class at any intersection contained in that feature.

`SplitLine <i_n_features> <out_feature_class>`



Fields Toolset

Contains tools to add and make changes to the fields in the tables of a feature class.

❖ **Add Field:** adds a field to the table of a feature class, layer or raster catalog.

`AddField <i_n_table> <field_name> <LONG | TEXT | FLOAT | DOUBLE | SHORT | DATE | BLOB> {field_precision} {field_scale} {field_length} {field_alias} {NULLABLE | NON_NULLABLE} {NON_REQUIRED | REQUIRED} {field_domain}`

❖ **Assign Default to Field:** creates a default value for a specified field and automatically applies a user-determined value to a certain field for every row added to the table or feature class.

`AssignDefaultToField <i_n_table> <field_name> <default_value> {subtype_code; subtype_code...}`

❖ **Calculate Field:** calculates the value of a field using an expression.

`CalculateField <i_n_table> <field> <expression>`

❖ **Delete Field:** deletes one or more fields from a table or feature class.

`DeleteField <i_n_table> <drop_field; drop_field... >`



General Toolset

Contains tools allowing for some simple dataset changes.

❖ **Append:** combines many feature classes into one feature class.

`Append <i nputs; i nputs. . . > <target> {TEST | NO_TEST}`

❖ **Copy:** copies feature datasets, feature classes, or tables and pastes them to another location.

`Copy <i n_data> <out_data> {data_type}`

❖ **Delete:** deletes feature datasets, feature classes, rasters, or tables.

`Delete <i n_data> {data_type}`

❖ **Rename:** changes the name of data, such as feature datasets, feature classes, rasters, tables, or toolboxes.

`Rename <i n_data> <out_data> {data_type}`

❖ **Select Data:** selects any data type on disk as input. This tool is intended for use within ModelBuilder™, not at the command line.

`SelectData <i n_data_element> <out_data_element>`



Generalization Toolset

Contains tools to derive data with less detail and complexity from a dataset.

❖ **Dissolve:** aggregates features based on one or more specified attributes.

`Dissolve <i n_features> <out_feature_class> {dissolve_field; dissolve_field. . . } {field{Statistics_Type}; field{Statistics_Type}. . . } {MULTI_PART | SINGLE_PART}`

Eliminate: eliminates features.

`Eliminate <i n_features> <out_feature_class> {LENGTH | AREA}`

◆ **Simplify Line:** removes small fluctuations or extraneous bends from a line in a feature class.

`SimplifyLine <i n_features> <out_feature_class> <POINT_REMOVE | BEND_SIMPLIFY> <tolerance> {FLAG_ERRORS | RESOLVE_ERRORS} {KEEP_COLLAPSED_POINTS | NO_KEEP}`

◆ **Smooth Line:** reduces the number of segments used to represent a line in a feature class.

`SmoothLine <i n_features> <out_feature_class> <PAEK | BEZIER_INTERPOLATION> <tolerance> {FIXED_CLOSED_ENDPOINT | NO_FIXED}`



Indexes Toolset

Contains tools to create, alter, and remove indexes.

- ❖ **Add Attribute Index:** adds an index to an existing table, feature class, shapefile, coverage, or attributed relationship class.

```
AddIndex <i n_table> <field ds; field ds. . . > {index_name} {NON_UNIQUE | UNIQUE}
{NON_ASCENDING | ASCENDING}
```

- ❖ **Add Spatial Index:** creates a new spatial index for a feature class in an ArcSDE geodatabase.

```
AddSpatialIndex <i n_features> {spatial_grid_1} {spatial_grid_2}
{spatial_grid_3}
```

- ❖ **Remove Attribute Index:** deletes an index from an existing table, feature class, shapefile, coverage, or attributed relationship class.

```
RemoveIndex <i n_table> <index_name; index_name. . . >
```

- ❖ **Remove Spatial Index:** deletes the spatial index for a feature class in an ArcSDE geodatabase.

```
RemoveSpatialIndex <i n_features>
```



Joins Toolset

Contains tools to add or remove a table join.

- ❖ **Add Join:** links a layer to a table (or a table to a table) based on a common field.

```
AddJoin <i n_layer_or_view> <i n_field> <join_table> <join_field> {KEEP_ALL |
KEEP_COMMON}
```

- ❖ **Remove Join:** removes the link between two tables.

```
RemoveJoin <i n_layer_or_view> <join_name>
```



Layers and Table Views Toolset

Contains tools for creating and manipulating layers, layer files, and table views.

- ❖ **Make Feature Layer:** creates a temporary layer.

```
MakeFeatureLayer <i n_features> <out_layer> {where_clause} {workspace}
{field_info}
```

- ❖ **Make Query Table:** represents the results of a SQL query to a database in a layer or table view.

```
MakeQueryTable <i n_table; i n_table. . . > <out_table> {USE_KEY_FIELDS |
ADD_VIRTUAL_KEY_FIELD | NO_KEY_FIELD} {i n_key_field; i n_key_field. . . }
<field{Alias}; field{Alias}. . . > {where_clause}
```

❖ **Make Raster Catalog Layer:** makes a temporary raster catalog layer.

```
MakeRasterCatalogLayer <i n_raster_catalog> <layer_name> {where_clause}  
{workspace} {file_info}
```

❖ **Make Raster Layer:** makes a temporary raster layer.

```
MakeRasterLayer <i n_raster> <out_raster_layer> {where_clause} {envelope}
```

❖ **Make Table View:** creates a temporary table.

```
MakeTableView <i n_table> <out_name> {where_clause} {workspace} {file_info}
```

❖ **Make XY Event Layer:** creates a temporary event layer with xy coordinates.

```
MakeXYEventLayer <table> <i n_x_filed> <i n_y_filed> <out_layer>  
{spatial_reference}
```

❖ **Save to Layer File:** creates a layer file on disk.

```
SaveToLayerFile <i n_layer> <out_layer>
```

❖ **Select Layer by Attribute:** creates, updates, or removes the selection on a layer or table view using an attribute query.

```
SelectLayerByAttribute <i n_layer_or_view> {NEW_SELECTION | ADD_TO_SELECTION |  
REMOVE_FROM_SELECTION | SUBSET_SELECTION | SWITCH_SELECTION |  
CLEAR_SELECTION} {where_clause}
```

❖ **Select Layer by Location:** creates, updates, or removes the selection of features in a layer based on a spatial relationship.

```
SelectLayerByLocation <i n_layer> {INTERSECT | WITHIN_A_DISTANCE |  
COMPLETELY_CONTAINS | COMPLETELY_WITHIN | HAVE_THEIR_CENTER_IN |  
SHARE_A_LINE_SEGMENT_WITH | BOUNDARY_TOUCHES | ARE_IDENTICAL_TO |  
CROSSED_BY_THE_OUTLINE_OF | CONTAINS | CONTAINED_BY} {select_features}  
{search_distance} {NEW_SELECTION | ADD_TO_SELECTION | REMOVE_FROM_SELECTION |  
SUBSET_SELECTION | SWITCH_SELECTION}
```



Projections and Transformations Toolset

Contains tools to set the projection as well as reproject or transform a dataset.


❖ **Define Projection:** sets the projection information for a dataset.

```
DefineProjection <i n_dataset> <coordinate_system>
```



Feature (Projections and Transformations) Toolset

Contains tools to convert a geographic dataset from one coordinate system to another.

 ❖ **Batch Project:** changes the coordinate system of one or more feature classes including the datum or spheroid.

```
BatchProject <i n_input_feature_class_or_dataset;  
input_feature_class_or_dataset... > <output_workspace>  
{output_coordinate_system} {template_dataset} {transformation}
```

❖ **Create Spatial Reference:** creates spatial reference and domains.

```
CreateSpatialReference {spatial_reference} {spatial_reference_template}  
{xy_domain} {z_domain} {m_domain} {template; template...} {expand_ratio}
```

❖ **Project:** changes the coordinate system of a feature class including its datum or spheroid.

```
Project <input_dataset> <out_dataset> <output_coordinate_system>  
{transform_method; transform_method...}
```



Raster (Projections and Transformations) Toolset

Contains tools to set the projection, reproject, reorient, or relocate a raster dataset.

❖ **Flip:** flips a raster dataset along a horizontal axis.

```
Flip <input_raster> <out_raster>
```

❖ **Mirror:** flips a raster dataset along the vertical axis.

```
Mirror <input_raster> <out_raster>
```

❖ **Project Raster:** converts a raster dataset between two coordinate systems.

```
ProjectRaster <input_raster> <out_raster> <output_coordinate_system>  
{NEAREST | BILINEAR | CUBIC} {cell_size}
```

❖ **Rescale:** scales a raster by the specified x and y scale factors.

```
Rescale <input_raster> <out_raster> <x_scale> <y_scale>
```

❖ **Rotate:** rotates a raster dataset around a specified point by a specified angle.

```
Rotate <input_raster> <out_raster> <angle> {pivot_point} {NEAREST | BILINEAR |  
CUBIC}
```

❖ **Shift:** shifts a raster by the specified x and y shift values.

```
Shift <input_raster> <out_raster> <x_value> <y_value> {input_snap_raster}
```


❖ **Warp:** transforms or rubber sheets a raster dataset along a set of links using a polynomial transformation.

```
Warp <input_raster> <source_control_points; source_control_points...>  
<target_control_points; target_control_points...> <out_raster> {POLYORDER1 |  
POLYORDER2 | POLYORDER3} {NEAREST | BILINEAR | CUBIC}
```




Raster Toolset

Contains tools to create and manage raster datasets and raster catalogs.

 ❖ **Batch Build Pyramids:** allows you to build pyramids on multiple raster datasets.

```
BatchBuildPyramids <input_raster_datasets; input_raster_datasets...>
```

 **❖ Batch Calculate Statistics:** allows you to calculate statistics on multiple raster datasets.

```
BatchCalculateStatistics <input_raster_datasets; input_raster_datasets... >  
{number_of_columns_to_skip} {number_of_rows_to_skip}  
{ignore_values; ignore_values... }
```

❖ Build Pyramids: builds raster pyramids for a raster dataset.

```
BuildPyramids <in_raster_dataset>
```

❖ Calculate Statistics: calculates statistics for a raster dataset.

```
CalculateStatistics <in_raster_dataset> {x_skip_factor} {y_skip_factor}  
{ignore_values; ignore_values... }
```

❖ Clip: creates a rectangular spatial subset of a raster dataset.

```
Clip <in_raster> <rectangle> <out_raster>
```

❖ Composite Bands: creates a multiband raster dataset from one or more raster datasets.

```
CompositeBands <in_rasters; in_rasters... > <out_raster>
```

❖ Copy Raster: converts a raster dataset into a GRID, TIFF, IMAGINE, or geodatabase raster dataset or loads raster datasets into a raster catalog.

```
CopyRaster <in_raster> <out_raster_dataset> {configuration_keyword}  
{background_value} {nodata_value} {NONE | ONEBITTO8BIT} {NONE | COLORMAPTORG  
{8_BIT_UNSIGNED | 1_BIT | 2_BIT | 4_BIT | 8_BIT_SIGNED | 16_BIT_UNSIGNED |  
16_BIT_SIGNED | 32_BIT_UNSIGNED | 32_BIT_SIGNED | 32_BIT_FLOAT | 64_BIT}
```

❖ Copy Raster Catalog Items: makes a copy of a raster catalog including all its contents.

```
CopyRasterCatalogItems <in_raster_catalog> <out_raster_catalog>  
{configuration_keyword} {spatial_grid_1} {spatial_grid_2} {spatial_grid_3}
```

❖ Create Raster Catalog: creates an empty raster catalog in a geodatabase.

```
CreateRasterCatalog <out_path> <out_name> {raster_spatial_reference}  
{spatial_reference} {configuration_keyword} {spatial_grid_1}  
{spatial_grid_2} {spatial_grid_3} {MANAGED | UNMANAGED}
```

❖ Create Raster Dataset: creates an empty raster dataset in a geodatabase.

```
CreateRasterDataset <out_path> <out_name> {cell_size} {8_BIT_UNSIGNED | 1_BIT |  
2_BIT | 4_BIT | 8_BIT_SIGNED | 16_BIT_UNSIGNED | 16_BIT_SIGNED |  
32_BIT_UNSIGNED | 32_BIT_SIGNED | 32_BIT_FLOAT | 64_BIT}  
{raster_spatial_reference} {number_of_bands} {configuration_keyword}  
{pyramids} {tile_size} {compression} {pyramid_origin}
```

❖ Delete Raster Catalog Items: deletes the raster catalogs contained in a raster catalog.

```
DeleteRasterCatalogItems <in_raster_catalog>
```

❖ **Mosaic:** mosaics multiple rasters into a single raster.

```
Mosaic <i nputs; i nputs. . . > <target> {FIRST | LAST | BLEND | MEAN | MINIMUM |  
MAXIMUM} {REJECT | FIRST | LAST | MATCH} {background_value} {nodata_value}  
{NONE | ONEBITTO8BIT} {mosack_tolerance}
```

 ❖ **Mosaic to New Raster:** mosaics multiple rasters into a new, single raster dataset.

```
MosaicToNewRaster <i nput_rasters; i nput_rasters. . . > <output_location>  
<raster_dataset_name_with_extent> {coordinate_system_for_the_raster}  
{8_BIT_UNSIGNED | 1_BIT | 2_BIT | 4_BIT | 8_BIT_SIGNED | 16_BIT_UNSIGNED |  
16_BIT_SIGNED | 32_BIT_UNSIGNED | 32_BIT_SIGNED | 32_BIT_FLOAT | 64_BIT}  
{cell_size} <number_of_bands> {FIRST | LAST | BLEND | MEAN | MINIMUM | MAXIMUM}  
{REJECT | FIRST | LAST | MATCH}
```

❖ **Resample:** changes the cell size of a grid.

```
Resample <i n_raster> <out_raster> <cell_size> {NEAREST | BILINEAR | CUBIC}
```

Relationship Classes Toolset

Contains tools to create associations between feature classes as well as feature classes and tables.

❖ **Create Relationship Class:** creates a relationship class to store an association between features in two feature classes or tables.

```
CreateRelationship <origin_table> <destination_table>  
<output_relationship_class> <SIMPLE | COMPOSITE> <forward_label >  
<backward_label > <NONE | FORWARD | BACKWARD | BOTH> <ONE_TO_ONE | ONE_TO_MANY |  
MANY_TO_MANY> <NONE | ATTRIBUTED> <origin_primary_key> <origin_foreign_key>  
{destination_primary_key} {<destination_foreign_key>
```

❖ **Table To Relationship Class:** creates an attributed relationship class from the Origin, Destination and Relationship tables.

```
TableToRelationship <origin_table> <destination_table>  
<output_relationship_class> <SIMPLE | COMPOSITE> <forward_label >  
<backward_label > <NONE | FORWARD | BACKWARD | BOTH> <ONE_TO_ONE | ONE_TO_MANY |  
MANY_TO_MANY> <relationship_table> <attribute_fields; attribute_fields. . . >  
<origin_primary_key> <origin_foreign_key> <destination_primary_key>  
<destination_foreign_key>
```

Subtypes Toolset

Contains tools to manage the subtypes of a feature class or a table.

❖ **Add Subtype:** adds a new subtype to the set of subtypes in a feature class or table.

```
AddSubtype <i n_table> <subtype_code> <subtype_description>
```

❖ **Remove Subtype:** deletes a subtype from the set of associated subtypes in a feature class or table.

```
RemoveSubtype <i n_table> <subtype_code; subtype_code. . . >
```

❖ **Set Default Subtype:** sets the default subtype value.

`SetDefaultSubtype <i n_table> <subtype_code>`

❖ **Set Subtype Field:** defines the field in the feature class or table that stores the subtype codes.

`SetSubtypeField <i n_table> <field>`



Table Toolset

Contains tools to help you create and evaluate tabular data from a variety of sources.

◆ **Analyze:** updates relational database management system statistics for a dataset.

`Analyze <i n_dataset> <BUSINESS | FEATURE | RASTER | ADDS | DELETES>`

◆ **Change Privileges:** changes the user privileges associated with a dataset.

`ChangePrivileges <i n_dataset> <user> {AS_IS | GRANT | REVOKE} {AS_IS | GRANT | REVOKE}`

❖ **Copy Rows:** copies rows from a feature class, layer, table, or table view.

`CopyRows <i n_rows> <out_table> {configuration_keyword}`

❖ **Create Table:** creates an empty geodatabase or dBASE table.

`CreateTable <out_path> <out_name> {template; template...} {configuration_keyword}`

❖ **Delete Rows:** deletes rows from a feature class, layer, table, or table view.

`DeleteRows <i n_rows>`

◆ **Get Count:** returns the number of rows in the feature class, layer, table, or table view.

`GetCount <i n_rows>`

Pivot Table: sorts and summarizes table fields, based on selected fields to reduce redundancy.

`PivotTable <i n_table> <fields; fields...> <pivot_field> <value_field> <out_table>`



Topology Toolset

Contains tools to establish and manage topological relationships between features.

◆ **Add Feature Class to Topology:** adds a new feature class to a topological relationship.

`AddFeatureClassToTopology <i n_topology> <i n_feature_class> <xy_rank> <z_rank>`

- ◆ **Add Rule to Topology:** adds a rule to the management of the topological relationship within a feature dataset.

```
AddRuleToTopology <in_topology> <Must Not Have Gaps (Area) | Must Not Overlap (Area) | Must Be Covered By Feature Class Of (Area-Area) | Must Cover Each Other (Area-Area) | Must Be Covered By (Area-Area) | Must Not Overlap With (Area-Area) | Must Be Covered By Boundary Of (Line-Area) | Must Be Covered By Boundary Of (Point-Area) | Must Be Properly Inside (Point-Area) | Must Not Overlap (Line) | Must Not Intersect (Line) | Must Not Have Dangles (Line) | Must Not Have Pseudo-Nodes (Line) | Must Be Covered By Feature Class Of (Line-Line) | Must Not Overlap With (Line-Line) | Must Be Covered By (Point-Line) | Must Be Covered By Endpoint Of (Point-Line) | Boundary Must Be Covered By (Area-Line) | Boundary Must Be Covered By Boundary Of (Area-Area) | Must Not Self-Overlap (Line) | Must Not Self-Intersect (Line) | Must Not Intersect Or Touch Interior (Line) | Endpoint Must Be Covered By (Line-Point) | Contains Point (Area-Point) | Must Be Single Part (Line)> <in_feature_class> {subtype} {in_feature_class2} {subtype2}
```

- ◆ **Create Topology:** creates a topology within a feature dataset.

```
CreateTopology <in_dataset> <out_name> <in_cluster_tolerance>
```

- ◆ **Remove Feature Class from Topology:** removes a feature class from the topological relationship.

```
RemoveFeatureClassFromTopology <in_topology> <in_feature_class>
```

- ◆ **Remove Rule from Topology:** removes a rule from a topological relationship.

```
RemoveRuleFromTopology <in_topology> <in_rule>
```

- ◆ **Set Cluster Tolerance:** alters the cluster tolerance in a topological relationship.

```
SetClusterTolerance <in_topology> <cluster_tolerance>
```

- ◆ **Validate Topology:** evaluates the features against the rules and finds any new errors related to new rules or feature classes.

```
ValidateTopology <in_topology> {FULL_EXTENT | VISIBLE_EXTENT}
```



Versions Toolset

Contains tools to make adjustments to the versions of the data.

- ◆ **Alter Version:** alters the properties of any of the versions of the dataset including name, description, and access permissions.

```
AlterVersion <in_workspace> <in_version> {name} {description} {PRIVATE | PUBLIC | PROTECTED}
```

- ◆ **Create Version:** creates a new version of the specified database.

```
CreateVersion <in_workspace> <parent_version> <version_name>
```


◆ **Delete Version:** deletes the specified version from the input workspace.

`DeleteVersion <i n_workspace> <version_name>`

◆ **Post Version:** applies the current edit session to the reconciled target version during versioned geodatabase editing.

`PostVersion <i n_workspace> <version_name>`

◆ **Reconcile Version:** reconciles an ArcSDE version against a parent version in its lineage.

`ReconcileVersion <i n_workspace> <version_name> <target_name>`

◆ **Register as Versioned:** registers an ArcSDE dataset as versioned in ArcCatalog™.

`RegisterAsVersioned <i n_dataset>`

◆ **Unregister as Versioned:** unregisters a dataset as versioned in ArcCatalog.

`UnregisterAsVersioned <i n_dataset> {KEEP_EDIT | NO_KEEP_EDIT}`



Workspace Toolset

Contains tools to create the storage models used with ArcGIS.

◆ **Create ArcInfo Workspace:** creates a workspace with an INFO subdirectory.

`CreateArcInfoWorkspace <out_folder_path> <out_name>`

◆ **Create Feature Dataset:** creates an empty feature dataset within an existing geodatabase.

`CreateFeatureDataset <out_dataset_path> <out_name> {spatial_reference}`

◆ **Create Folder:** creates a new folder.

`CreateFolder <i n_folder_path> <out_name>`

◆ **Create Personal GDB:** creates a new personal geodatabase.

`CreatePersonalGDB <out_folder_path> <out_name>`



Geocoding Toolbox

Contains tools used to manage a geocoding service and run geocoding actions.

- ❖ **Automate Geocoding Indexes:** creates an automatically updating relationship between the reference data and the geocoding index(es) of an address locator.

```
AutomateGeocodingIndexes <i n_address_locator>
```

- ❖ **Create Address Locator:** creates a new address locator.

```
CreateAddressLocator <i n_address_locator_style>  
<reference_data{Role}; reference_data{Role}... > <i n_field_map>  
<out_address_locator>
```

- ❖ **Deautomate Geocoding Indexes:** removes the automatically updating relationship between the reference data and geocoding index of an address locator.

```
DeautomateGeocodingIndexes <i n_address_locator>
```

- ❖ **Delete Address Locator:** deletes an address locator.

```
DeleteAddressLocator <i n_address_locator>
```

- ❖ **Geocode Addresses:** creates a point feature class from a table of addresses.

```
GeocodeAddresses <i n_table> <address_locator> <i n_address_fields>  
<out_feature_class> {STATIC | DYNAMIC}
```

- ❖ **Rebuild Geocoding Indexes:** rebuilds the indexes of an address locator.

```
RebuildGeocodingIndex <i n_address_locator>
```

- ❖ **Standardize Addresses:** standardizes the address information in a table or feature class.

```
StandardizeAddresses <i n_address_data>  
<i n_input_address_fields; i n_input_address_fields... >  
<i n_address_locator_style>  
<i n_output_address_fields; i n_output_address_fields... > <out_address_data>  
{STATIC | DYNAMIC}
```




Geostatistical Analyst Toolbox

Contains tools for exploratory spatial data analysis.

GA Layer to Grid: exports a Geostatistical layer to an ArcInfo GRID.

```
GALayerToGrid <in_geostat_layer> <out_surface_grid> <cell_size>  
<points_per_block_horz> <points_per_block_vert>
```




Linear Referencing Toolbox

Contains tools to model relative locations along linear features and associate multiple sets of attributes to portions of linear features.

- ❖ **Calibrate Routes:** adjusts route measures by reading measure information stored as an attribute in a point feature class.

```
CalibrateRoutes <i n_route_features> <route_id_fiel d> <i n_point_features>
<point_id_fiel d> <measure_fiel d> <out_feature_class> {DISTANCE | MEASURES}
{search_radius} {BETWEEN | NO_BETWEEN} {BEFORE | NO_BEFORE} {AFTER | NO_AFTER}
{IGNORE | NO_IGNORE} {KEEP | NO_KEEP} {INDEX | NO_INDEX}
```

- ❖ **Create Routes:** creates routes from existing lines.

```
CreateRoutes <i n_line_features> <route_id_fiel d> <out_feature_class>
<LENGTH | ONE_FIELD | TWO_FIELDS> <from_measure_fiel d> <to_measure_fiel d>
{UPPER_LEFT | LOWER_LEFT | UPPER_RIGHT | LOWER_RIGHT} {measure_factor}
{measure_offset} {IGNORE | NO_IGNORE} {INDEX | NO_INDEX}
```

- ❖ **Dissolve Route Events:** removes redundant information from event tables or separates event tables having more than one descriptive attribute into separate tables.

```
DissolveRouteEvents <i n_events> <i n_event_properties> <dissolve_fiel d>
dissolve_fiel d... > <out_table> <out_event_properties>
{DISSOLVE | CONCATENATE} {INDEX | NO_INDEX}
```

- ❖ **Locate Features Along Routes:** computes the intersection of input features (point, line, or polygon) and route features and writes the route and measure information to a new event table.

```
LocateFeaturesAlongRoutes <i n_features> <i n_routes> <route_id_fiel d>
<radius_or_tolerance> <out_table> <out_event_properties> {FIRST | ALL}
{DISTANCE | NO_DISTANCE} {ZERO | NO_ZERO} {FIELDS | NO_FIELDS}
```

- ❖ **Make Route Event Layer:** creates a temporary route event layer.

```
MakeRouteEventLayer <i n_routes> <route_id_fiel d> <i n_table>
<i n_event_properties> <out_layer> {offset_fiel d} {NO_ERROR_FIELD |
ERROR_FIELD} {NO_ANGLE_FIELD | ANGLE_FIELD} {NORMAL | TANGENT}
{ANGLE | COMPLEMENT} {LEFT | RIGHT} {POINT | MULTIPOINT}
```

- ❖ **Overlay Route Events:** combines two input event tables to create a single output event table, using either a union or intersection operation.

```
OverlayRouteEvents <i n_table> <i n_event_properties> <overlay_table>
<overlay_event_properties> <INTERSECT | UNION> <out_table>
<out_event_properties> {ZERO | NO_ZERO} {FIELDS | NO_FIELDS} {INDEX | NO_INDEX}
```

- ❖ **Transform Route Events:** transforms the measures of events from one route reference to another and writes them to a new event table.

```
TransformRouteEvents <i n_table> <i n_event_properties> <i n_routes>  
<route_id_field> <target_routes> <target_route_id_field> <out_table>  
<out_event_properties> <cluster_tolerance> {FIELD | NO_FIELD}
```


Network Analyst Toolbox

Contains tools for network analyst, building networks and creating and editing turns.

Analysis toolset

Contains tools used to perform analysis when using the Network Analyst extensions.

Add Field To Analysis Layer: adds a field to a network analyst layer.

```
AddFieldToAnalysisLayer <i_n_network_anal_ysi_s_l_ayer> <sub_l_ayer> <fi_el_d_name>
  <LONG | TEXT | FLOAT | DOUBLE | SHORT | DATE | BLOB> {fi_el_d_preci_si_on}
  {fi_el_d_scal_e} {fi_el_d_l_ength} {fi_el_d_al_i_as} {NULLABLE | NON_NULLABLE}
```

Add Locations: adds network locations to a network analysis layer.

```
AddLocations <i_n_network_anal_ysi_s_l_ayer> <sub_l_ayer> <i_n_tabl_e>
  <fi_el_d_mappi_ngs> <search_tol_erance> {sort_fi_el_d} {source {snap_type}; source
  {snap_type}... } {MATCH_TO_CLOSEST | PRI_ORI_TY} {APPEND | CLEAR}
```

Calculate Location Fields: calculates the network location fields for a point feature class.

```
CalculateLocationFields <i_n_poi_nt_features> <i_n_network_dataset>
  <search_tol_erance> <source {snap_type}; source {snap_type}... >
  {MATCH_TO_CLOSEST | PRI_ORI_TY} {source_ID_fi_el_d} {source_OD_fi_el_d}
  {posi_ti_on_fi_el_d} {si_de_fi_el_d} {snap_X_fi_el_d} {snap_Y_fi_el_d} {di_stance_fi_el_d}
```

Directions: generates directions information for a network analysis layer with routes. The directions information is written to a file in either XML or text format.

```
Directions <i_n_network_anal_ysi_s_l_ayer> <XML | TEXT> <out_di_recti_ons_fi_l_e> <Feet
  | Yards | Mi_l_es | Meters | Ki_l_ometers> {REPORT_TI_ME | NO_REPORT_TI_ME}
  {ti_me_attri_bute}
```

Make Closest Facility Layer: makes a closest facility network analysis layer and sets its navigation properties.

```
MakeClosestFacilityLayer <i_n_network_dataset> <out_network_anal_ysi_s_l_ayer>
  <i_mpedance_attri_bute> {TRAVEL_TO | TRAVEL_FROM} {defaul_t_cutoff}
  {defaul_t_number_facil_i_ties_to_fi_nd} {accumul_ate_attri_bute_name;
  accumul_ate_attri_bute_name... } {ALLOW_UTURNS | NO_UTURNS |
  ALLOW_DEAD_ENDS_ONLY} {restri_cti_on_attri_bute_name;
  restri_cti_on_attri_bute_name... } {NO_HI_ERARCHY | USE_HI_ERARCHY}
  {hi_erarchy_setti_ngs} {PATH_SHAPE | STRAI_GHT_LI_NE | NO_SHAPE}
```

Make OD Cost Matrix Layer: makes an origin and destination cost matrix layer and sets its navigation properties.

```
MakeODCostMatrixLayer <i_n_network_dataset> <out_network_anal_ysi_s_l_ayer>
  <i_mpedance_attri_bute> {defaul_t_cutoff}
  {defaul_t_number_desti_nati_ons_to_fi_nd} {accumul_ate_attri_bute_name;
  accumul_ate_attri_bute_name... } {ALLOW_UTURNS | NO_UTURNS |
  ALLOW_DEAD_ENDS_ONLY} {restri_cti_on_attri_bute_name;
  restri_cti_on_attri_bute_name... } {NO_HI_ERARCHY | USE_HI_ERARCHY}
  {hi_erarchy_setti_ngs} {STRAI_GHT_LI_NE | NO_SHAPE}
```

Make Route Layer: makes a route network analysis layer and sets its navigation properties.

```
MakeRouteLayer <i n_network_dataset> <out_network_analysis_layer>
  <i mpedance_attribute> {USE_INPUT_ORDER | FIND_BEST_ORDER} {PRESERVE_BOTH |
  PRESERVE_NONE | PRESERVE_FIRST | PRESERVE_LAST} {USE_TIMEWINDOWS |
  NO_TIMEWINDOWS} {accumulated_attribute_name; accumulated_attribute_name...}
  {ALLOW_UTURNS | NO_UTURNS | ALLOW_DEAD_ENDS_ONLY}
  {restriction_attribute_name; restriction_attribute_name...} {NO_HIERARCHY |
  USE_HIERARCHY} {hierarchy_settings} {PATH_SHAPE | STRAIGHT_LINE | NO_SHAPE}
```

Make Service Area Layer: makes a service area network analysis layer and sets its navigation properties.

```
MakeServiceAreaLayer <i n_network_dataset> <out_network_analysis_layer>
  <i mpedance_attribute> {TRAVEL_FROM | TRAVEL_TO} {default_t_break_values}
  {SIMPLE_POLYGONS | DETAILED_POLYGONS | NO_POLYS} {NO_MERGE | MERGE} {RINGS |
  DISKS} {NO_LINES | LINES | LINES_MEASURED} {OVERLAP | NON_OVERLAP} {NO_SPLIT |
  SPLIT} {out_line_feature_class}
  {excluded_source_name; excluded_source_name...}
  {accumulated_attribute_name; accumulated_attribute_name...} {ALLOW_UTURNS |
  NO_UTURNS | ALLOW_DEAD_ENDS_ONLY}
  {restriction_attribute_name; restriction_attribute_name...}
```

Solve: performs the analysis appropriate to the network analysis layer on which it is executed.

```
Solve_na <i n_network_analysis_layer>
```



Network Dataset toolset

Contains tools used to build network datasets.

Build Network: reconstructs the network connectivity and attribute information of a network dataset.

```
BuildNetwork_na <i n_network_dataset>
```



Turn Feature Class toolset

Contains tools used to build and edit turn data.

Create Turn Feature Class: creates a new turn feature class.

```
CreateTurnFeatureClass <out_location> <out_feature_class_name> {maximum_edges}
  {i n_network_dataset} {i n_template_feature_class} {spatial_reference}
  {config_keyword} {spatial_grid_1} {spatial_grid_2} {spatial_grid_3}
```

Increase Maximum Edges: increases the maximum number of edges in a turn feature class.

```
IncreaseMaximumEdges <i n_turn_features> <maximum_edges>
```

Populate Alternate ID Fields: creates and populates additional fields on the turn feature class(es) in a network dataset that reference the edges by alternate IDs.

```
PopulateAlternateIDFields <i n_network_dataset> <alternate_id_field_name>
```

Turn Table to Turn Feature Class: converts an ArcView GIS turn table or ArcInfo Workstation coverage turn table to an ArcGIS turn feature class.

```
TurnTableToFeatureClass <i n_turn_table> <reference_line_features>  
<out_feature_class_name> {reference_nodes_table} {maximum_edges}  
{config_keyword} {spatial_grid_1} {spatial_grid_2} {spatial_grid_3}
```

Update by Alternate ID Fields: updates all the edge references in each turn feature class in a network dataset using an alternate ID field.

```
UpdateByAlternateIDFields <i n_network_dataset> <alternate_ID_field_name>
```

Update by Geometry: updates all the edge references in the turn table using the geometry of the feature.

```
UpdateByGeometry <i n_turn_features>
```


Spatial Analyst Toolbox

This toolbox provides tools for performing cell-based (raster) spatial analysis.

Conditional Toolset

Contains tools to control the output values based on the conditions placed on the input values.

Con: performs one or more conditional if/else evaluations on a cell-by-cell basis.

```
Con <i n_conditional_raster> <i n_true_raster_or_constant> <out_raster>
  {i n_false_raster_or_constant} {where_clause}
```

Pick: uses the values of the cell to determine which expression in a list will be used to compute the output raster dataset.

```
Pick <i n_position_raster> <i n_rasters_or_constants;
  i n_rasters_or_constants... > <out_raster>
```

Set Null: defines the null cells within a raster dataset.

```
SetNull <i n_conditional_raster> <i n_false_raster_or_constant> <out_raster>
  {where_clause}
```

Density Toolset

Contains tools used to calculate the spread of values over a surface.

Kernel Density: calculates a magnitude per unit area from point or polyline feature using a kernel function to fit a smoothly tapered surface to each point or polyline.

```
KernelDensity <i n_features> <population_field> <out_raster> {cell_size}
  {search_radius} {SQUARE_MAP_UNITS | SQUARE_MILES | SQUARE_KILOMETERS | ACRES |
  HECTARES | SQUARE_YARDS | SQUARE_FEET | SQUARE_INCHES | SQUARE_METERS |
  SQUARE_CENTIMETERS | SQUARE_MILLIMETERS}
```

Line Density: calculates a magnitude per unit area from polyline features that fall within a radius around each cell.

```
LineDensity <i n_polyline_features> <population_field> <out_raster> {cell_size}
  {search_radius} {SQUARE_MAP_UNITS | SQUARE_MILES | SQUARE_KILOMETERS | ACRES |
  HECTARES | SQUARE_YARDS | SQUARE_FEET | SQUARE_INCHES | SQUARE_METERS |
  SQUARE_CENTIMETERS | SQUARE_MILLIMETERS}
```

Point Density: calculates a magnitude per unit area from point features that fall within a neighborhood around each cell.

```
PointDensity <i n_point_features> <population_field> <out_raster> {cell_size}
  {neighborhood} {SQUARE_MAP_UNITS | SQUARE_MILES | SQUARE_KILOMETERS | ACRES |
  HECTARES | SQUARE_YARDS | SQUARE_FEET | SQUARE_INCHES | SQUARE_METERS |
  SQUARE_CENTIMETERS | SQUARE_MILLIMETERS}
```



Distance Toolset

Contains tools used to compute distance across a raster dataset with respect to cost or along a path.

Corridor: computes the sum of two accumulative cost raster datasets.

```
Corridor <i_n_distance_raster1> <i_n_distance_raster2> <out_raster>
```

Cost Allocation: identifies for each cell the zone of each source cell that could be reached with the least accumulative cost.

```
CostAllocation <i_n_source_data> <i_n_cost_raster> <out_allocation_raster>
{maximum_distance} {i_n_value_raster} {source_file_id} {out_distance_raster}
{out_backlink_raster}
```

Cost Back Link: defines the neighbor cell on the least accumulative cost path from a cell to a set of source cells.

```
CostBackLink <i_n_source_data> <i_n_cost_raster> <out_backlink_raster>
{maximum_distance} {out_distance_raster}
```

Cost Distance: calculates the least accumulative cost distance over a cost surface.

```
CostDistance <i_n_source_data> <i_n_cost_raster> <out_distance_raster>
{maximum_distance} {out_backlink_raster}
```

Cost Path: calculates the least cost paths from a source to a destination over a surface.

```
CostPath <i_n_destination_data> <i_n_cost_distance_raster>
<i_n_cost_backlink_raster> <out_raster>
{EACH_CELL | EACH_ZONE | BEST_SINGLE} {destination_file_id}
```

Euclidean Allocation: assigns each cell the value of the sources to which it is closest.

```
EucAllocation <i_n_source_data> <out_allocation_raster> {maximum_distance}
{i_n_value_raster} {cell_size} {source_file_id} {out_distance_raster}
{out_direction_raster}
```

Euclidean Direction: computes for each cell the direction to the nearest source, measured in degrees.

```
EucDirection <i_n_source_data> <out_direction_raster> {maximum_distance}
{cell_size} {out_distance_raster}
```

Euclidean Distance: calculates for each cell the straight-line distance from each cell to the closest source.

```
EucDistance <i_n_source_data> <out_distance_raster> {maximum_distance}
{cell_size} {out_direction_raster}
```

Path Distance: calculates for each cell the least accumulative cost distance over a cost surface from a source cell or a set of source cells while accounting for surface distance and horizontal and vertical cost factors.

```
PathDistance <i_n_source_data> <out_distance_raster> {i_n_cost_raster}
{i_n_surface_raster} {i_n_horizontal_raster} {horizontal_factor}
{i_n_vertical_raster} {vertical_factor} {maximum_distance}
{out_backlink_raster}
```

Path Distance Allocation: calculates for each cell its nearest source based on the least accumulative cost over a cost surface, while accounting for surface distance, horizontal cost factors, and vertical cost factors.

```
PathAllocation <i_n_source_data> <out_allocation_raster> {i_n_cost_raster}
{i_n_surface_raster} {i_n_horizontal_raster} {horizontal_factor}
{i_n_vertical_raster} {vertical_factor} {maximum_distance} {i_n_value_raster}
{source_field} {out_distance_raster} {out_backlink_raster}
```

Path Distance Back Link: defines the neighbor that is the next cell on the least accumulative cost path to the nearest source, while accounting for surface distance, horizontal cost factors, and vertical cost factors.

```
PathBackLink <i_n_source_data> <out_backlink_raster> {i_n_cost_raster}
{i_n_surface_raster} {i_n_horizontal_raster} {horizontal_factor}
{i_n_vertical_raster} {vertical_factor} {maximum_distance}
{out_distance_raster}
```



Extraction Toolset

Contains tools to extract a subset of cells either by the attributes or spatial location of each cell.

Extract by Attributes: extracts the cells of a raster dataset based on a logical query.

```
ExtractByAttributes <i_n_raster> <where_clause> <out_raster>
```

Extract by Circle: extracts the cell values of a raster dataset based on the boundaries of a circle.

```
ExtractByCircle <i_n_raster> <center_point> <radius> <out_raster>
{INSIDE | OUTSIDE}
```

Extract by Mask: extracts the raster dataset cell values based on a mask dataset.

```
ExtractByMask <i_n_raster> <i_n_mask_data> <out_raster>
```

Extract by Points: extracts the cells of a raster dataset based on a set of points.

```
ExtractByPoints <i_n_raster> <points; points... > <out_raster>
{INSIDE | OUTSIDE}
```

Extract by Polygon: extracts the cells of a raster dataset based on the boundaries within a polygon feature.

```
ExtractByPolygon <i n_raster> <polygon; polygon. . . > <out_raster>  
{INSIDE | OUTSIDE}
```

Extract by Rectangle: extracts the cells of a raster dataset based on the boundaries of a rectangle.

```
ExtractByRectangle <i n_raster> <rectangle> <out_raster> {INSIDE | OUTSIDE}
```

Extract Values to Points: extracts the cell values from a raster at the locations of points in a feature class.

```
ExtractValuesToPoints <i n_point_features> <i n_raster> <out_point_features>  
{NONE | INTERPOLATE} {VALUE_ONLY | ALL}
```

Sample: writes a sample of cell values from a group of rasters to a table.

```
Sample <i n_rasters; i n_rasters. . . > <i n_location_data> <out_table>  
{NEAREST | BILINEAR | CUBIC}
```



Generalization Toolset

Contains tools to remove or reduce erroneous or irrelevant data within a raster through aggregation, edge smoothing, intelligent noise removal, and so forth.

Aggregate: generates a reduced resolution raster dataset.

```
Aggregate <i n_raster> <out_raster> <cell_factor>  
{SUM | MAXIMUM | MEAN | MEDIAN | MINIMUM} {EXPAND | TRUNCATE}  
{DATA | NODATA}
```

Boundary Clean: smoothes the boundary between zones by expanding and shrinking the boundary.

```
BoundaryClean <i n_raster> <out_raster> {NO_SORT | DESCEND | ASCEND}  
{TWO_WAY | ONE_WAY}
```

Expand: expands the selected zones of a raster dataset by a specified number of cells.

```
Expand <i n_raster> <out_raster> <number_cells> <zone_values; zone_values. . . >
```

Majority Filter: replaces cell values within a raster dataset based upon the majority of their contiguous neighboring cells.

```
MajorityFilter <i n_raster> <out_raster> {FOUR | EIGHT} {MAJORITY | HALF}
```

Nibble: replaces cell values in a raster dataset corresponding to a mask with the values of the nearest neighbors.

```
Nibble <i n_raster> <i n_mask_raster> <out_raster> {ALL_VALUES | DATA_ONLY}
```


Region Group: records for each cell of a raster the identity of the connected region to which it belongs. A unique number is assigned to each region.

`RegionGroup <i_n_raster> <out_raster> {FOUR | EIGHT} {WITHIN | CROSS} {ADD_LINK | NO_LINK} {excluded_value}`

Shrink: shrinks the selected zones by a specified number of cells.

`Shrink <i_n_raster> <out_raster> <number_cells> <zone_values; zone_values... >`

Thin: thins rasterized linear features in a raster dataset.

`Thin <i_n_raster> <out_raster> {ZERO | NODATA} {NO_FILTER | FILTER} {ROUND | SHARP} {maximum_thickness}`



Groundwater Toolset

Contains tools used to measure hydrodynamic movement within or along a surface.

Darcy Flow: calculates the groundwater volume balance residual and other outputs for steady flow in an aquifer.

`DarcyFlow <i_n_head_raster> <i_n_porosity_raster> <i_n_thickness_raster> <i_n_transmissivity_raster> <out_volume_raster> {out_direction_raster} {out_magnitude_raster}`

Darcy Velocity: calculates the groundwater seepage velocity vector (direction and magnitude) for steady flow in an aquifer.

`DarcyVelocity <i_n_head_raster> <i_n_porosity_raster> <i_n_thickness_raster> <i_n_transmissivity_raster> <out_direction_raster> <out_magnitude_raster>`

Particle Track: calculates the path of a particle through a velocity field.

`ParticleTrack <i_n_direction_raster> <i_n_magnitude_raster> <source_point> <out_track_file> {step_length} {tracking_time} {out_track_polygon_features}`

Porous Puff: calculates the hydrodynamic dispersion of an instantaneous point release of a constituent as it is advected along the flow path.

`PorousPuff <i_n_track_file> <i_n_porosity_raster> <i_n_thickness_raster> <out_raster> <mass> {dispersion_time} {longitudinal_dispersion} {dispersion_ratio} {retardation_factor} {decay_coefficient}`



Hydrology Toolset

Contains tools providing hydrology functions to simulate the flow of water over an elevation surface and creates either a stream network or a watershed.

Basin: creates a raster dataset delineating all drainage basins.

`Basin <i_n_flow_direction_raster> <out_raster>`

Fill: fills sinks in a surface raster to remove small imperfections in the data.

`Fill <i_n_surface_raster> <out_surface_raster> {z_limit}`

Flow Accumulation: creates a raster dataset of accumulated flow to each cell by accumulating the weight for all cells that flow into each downslope cell.

```
FlowAccumulation <in_flow_direction_raster> <out_accumulation_raster>  
{in_weight_raster}
```

Flow Direction: creates a raster dataset of flow direction from each cell to its steepest downslope neighbor.

```
FlowDirection <in_surface_raster> <out_flow_direction_raster>  
{NORMAL | FORCE} {out_drop_raster}
```

Flow Length: calculates upstream or downstream distance or weighted distance along a flow path for each cell.

```
FlowLength <in_flow_direction_raster> <out_raster> {DOWNSTREAM | UPSTREAM}  
{in_weight_raster}
```

Sink: creates a raster dataset identifying all sinks or areas of internal drainage.

```
Sink <in_flow_direction_raster> <out_raster>
```

Snap Pour Point: snaps pour points to the cell of highest flow accumulation within a specified distance.

```
SnapPourPoint <in_pour_point_data> <in_accumulation_raster> <out_raster>  
<snap_distance> {pour_point_field}
```

Stream Link: assigns unique values to sections of a raster linear network between intersections.

```
StreamLink <in_stream_raster> <in_flow_direction_raster> <out_raster>
```

Stream Order: assigns a numeric order to segments of a raster dataset representing branches of a linear network.

```
StreamOrder <in_stream_raster> <in_flow_direction_raster> <out_raster>  
{STRAHLER | SHREVE}
```

Stream to Feature: converts a raster dataset representing a linear network to features representing the linear network.

```
StreamToFeature <in_stream_raster> <in_flow_direction_raster>  
<out_polyline_features> {SIMPLIFY | NO_SIMPLIFY}
```

Watershed: determines the contributing area above a set of cells in a raster dataset.

```
Watershed <in_flow_direction_raster> <in_pour_point_data> <out_raster>  
{pour_point_field}
```

Interpolation Toolset

Contains tools to create a raster surface from point features.

IDW: interpolates a surface from points using an inverse distance weighted (IDW) technique.

```
IDW <i n_poi nt_features> <z_fi el d> <out_raster> {cel l _si ze} {power}
  {search_radi us} {i n_barri er_pol yl i ne_features}
```

Krige: interpolates a raster dataset from a set of points using kriging.

```
Kri gi ng_sa <i n_poi nt_features> <z_fi el d> <out_surface_raster>
  <semi Vari ogram_props> {cel l _si ze} {search_radi us}
  {out_vari ance_predi cti on_raster}
```

Natural Neighbor: interpolates a surface from points using a natural neighbor technique.

```
Natural Nei ghbor <i n_poi nt_features> <z_fi el d> <out_raster> {cel l _si ze}
```

Spline: interpolates a surface from points using a minimum curvature spline technique.

```
Spl i ne <i n_poi nt_features> <z_fi el d> <out_raster> {cel l _si ze}
  {REGULARI ZED | TENSI ON} {wei ght} {number_poi nts}
```

Topo to Raster: generates a hydrologically correct raster dataset of elevation.

```
TopoToRaster <feature_l ayer{Fi el d} {Type}; feature_l ayer{Fi el d} {Type}. . . >
  <out_surface_raster> {cel l _si ze} {extent} {Margi n} {mi ni mum_z_val ue}
  {maxi mum_z_val ue} {ENFORCE | NO_ENFORCE | ENFORCE_WI TH_SI NK}
  {CONTOUR | SPOT} {maxi mum_i terati ons} {roughness_penal ty}
  {di screte_error_factor} {verti cal _standard_error} {tol erance_1}
  {tol erance_2} {out_stream_features} {out_si nk_features}
  {out_di agnosti c_fi le} {out_parameter_fi le}
```

Topo to Raster by File: generates a hydrologically correct raster dataset of elevation.

```
TopoToRasterByFi le <i n_parameter_fi le> <out_surface_raster>
  {out_stream_features} {out_si nk_features}
```

Trend: interpolates a surface from points using a trend technique.

```
Trend <i n_poi nt_features> <z_fi el d> <out_raster> {cel l _si ze} {order}
  {LI NEAR | LOGI STI C}
```

Local Toolset

Contains tools to compute an output raster dataset where the output value at each location is a function of the value associated with that location on one or more raster datasets.

Cell Statistics: calculates a per cell statistic from multiple raster datasets.

```
Cell Stati sti cs <i n_rasters_or_constants; i n_rasters_or_constants. . . >
  <out_raster> {MEAN | MAJORI TY | MAXI MUM | MEDI AN | MI NI MUM | MI NORI TY | RANGE |
  STD | SUM | VARI ETY}
```

Combine: combines multiple raster datasets on a cell-by-cell basis.

`Combine <input_rasters; input_rasters... > <output_raster>`

Equal to Frequency: evaluates the number of times the input raster dataset values are equal to a specified value on a cell-by-cell basis.

`EqualToFrequency <input_value_raster> <input_rasters; input_rasters... > <output_raster>`

Greater Than Frequency: evaluates the number of times the input raster dataset values are greater than a specified value on a cell-by-cell basis.

`GreaterThanFrequency <input_value_raster> <input_rasters; input_rasters... > <output_raster>`

Highest Position: determines the position of a raster dataset with the maximum value in a set of raster datasets.

`HighestPosition <input_rasters_or_constants; input_rasters_or_constants... > <output_raster>`

Less Than Frequency: evaluates the number of times the input raster dataset values are less than a specified value on a cell-by-cell basis.

`LessThanFrequency <input_value_raster> <input_rasters; input_rasters... > <output_raster>`

Lowest Position: determines the position for each cell of the input raster dataset with the minimum value in the argument list.

`LowestPosition <input_rasters_or_constants; input_rasters_or_constants... > <output_raster>`

Popularity: determines the value that is at a specified level of popularity on a cell-by-cell basis.

`Popularity <input_popularity_raster_or_constant> <input_rasters; input_rasters... > <output_raster>`

Rank: returns the value of a set of raster datasets based on a rank level specified by another raster dataset on a cell-by-cell basis.

`Rank <input_rank_raster_or_constant> <input_rasters; input_rasters... > <output_raster>`



Map Algebra Toolset

Contains the tool to create expressions using any of the Map Algebra statements. Map Algebra is the analysis language from Spatial Analyst.

Multi Output Map Algebra: executes GRID's map algebra statements.

`Mul ti OutputMapAl gebra <expressi on_stri ng>`

Single Output Map Algebra: executes GRID's map algebra statement to produce a raster dataset.

`Si ngl eOutputMapAl gebra <expressi on_stri ng> <out_raster> {i n_data; i n_data. . . }`



Math Toolset

Contains tools to implement math functions, which apply a specified mathematical operation or function to each cell location on an input raster or series of raster datasets.

Abs: calculates the absolute value of the input raster dataset on a cell-by-cell basis.

`Abs <i n_raster_or_constant> <out_raster>`

Divide: divides the values of two input raster datasets on a cell-by-cell basis.

`Di vi de <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Exp: calculates the base e exponential of cells in a raster dataset.

`Exp <i n_raster_or_constant> <out_raster>`

Exp10: calculates the base 10 exponential of cells in a raster dataset.

`Exp10 <i n_raster_or_constant> <out_raster>`

Exp2: calculates the base 2 exponential of cells in a raster dataset.

`Exp2 <i n_raster_or_constant> <out_raster>`

Float: converts each cell value in a raster dataset to floating-point values.

`Fl oat <i n_raster_or_constant> <out_raster>`

Int: converts each cell value in a raster dataset to an integer by truncation.

`I nt <i n_raster_or_constant> <out_raster>`

Ln: calculates the natural logarithm (base e) of cells in a raster dataset.

`Ln <i n_raster_or_constant> <out_raster>`

Log10: calculates the base 10 logarithm of cells in a raster dataset.

`Log10 <i n_raster_or_constant> <out_raster>`

Log2: calculates the base 2 logarithm of cells in a raster dataset.

Log2 <i n_raster_or_constant> <out_raster>

Minus: subtracts the values of the second input from the values of the first input on a cell-by-cell basis.

Mi nus <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Mod: divides the values of the first input by the values of the second input and returns the remainder.

Mod <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Negate: changes the sign of the cell values of the input raster dataset (multiplies by -1).

Negate <i n_raster_or_constant> <out_raster>

Plus: adds the values of two raster datasets on a cell-by-cell basis.

Pl us <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Power: calculates the “nth” power of the input raster or number on a cell-by-cell basis.

Power <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Round Down: returns the next lower whole number value for each cell in a raster dataset.

RoundDown <i n_raster_or_constant> <out_raster>

Round Up: returns the next highest whole number value that is greater than or equal to the input value for each cell in a raster dataset.

RoundUp <i n_raster_or_constant> <out_raster>

Square: calculates the square of cell values in a raster dataset.

Square <i n_raster_or_constant> <out_raster>

Square Root: calculates the square root of the input grid or number for each cell.

SquareRoot <i n_raster_or_constant> <out_raster>

Times: multiplies the values of two raster datasets on a cell-by-cell basis.

Ti mes <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Bitwise (Math) Toolset

Contains tools to implement bitwise operators, which treat the operands as bits (binary representations) and calculate the output by applying the logical operation (e.g., 3 BITWISEAND 5 = 0011 && 0101 = 0001 = 1).

Bitwise And: performs the bitwise AND operation on the binary values of two inputs on a cell-by-cell basis (e.g., 0011 && 0101 = 0001).

`BitwiseAnd <input_raster_or_constant1> <input_raster_or_constant2> <output_raster>`

Bitwise Left Shift: shifts the bits to the left using the number specified (e.g., 1<<2 = 0001 << 2 = 0100 = 4).

`BitwiseLeftShift <input_raster_or_constant1> <input_raster_or_constant2> <output_raster>`

Bitwise Not: performs the bitwise COMPLEMENT operation on the binary values of two inputs on a cell-by-cell basis (flips the bits; e.g., 5 = 0101 ~ 1010 = 10).

`BitwiseNot <input_raster_or_constant> <output_raster>`

Bitwise Or: performs a bitwise OR operation on the binary values of two inputs on a cell-by-cell basis (e.g., 0101 || 1100 = 1101).

`BitwiseOr <input_raster_or_constant1> <input_raster_or_constant2> <output_raster>`

Bitwise Right Shift: shifts the bits to the right using the number specified (e.g., 6>>1 = 0110 >> 1 = 0011 = 3).

`BitwiseRightShift <input_raster_or_constant1> <input_raster_or_constant2> <output_raster>`

Bitwise XOR: performs a bitwise exclusive OR operation on the binary values of two inputs on a cell-by-cell basis (e.g., 0101 !! 1100 = 1001).

`BitwiseXor <input_raster_or_constant1> <input_raster_or_constant2> <output_raster>`

Logical (Math) Toolset

Contains tools to evaluate the values of an input raster or rasters relative to a conditional statement, the values in another raster, a constant value, or a specific value. Also contains tools that can produce an output that tracks the unique combinations of the input values between two rasters or constants.

Boolean And: performs the Boolean AND operator on the cell values of two input raster datasets.

`BooleanAnd <input_raster_or_constant1> <input_raster_or_constant2> <output_raster>`

Boolean Not: performs the Boolean COMPLEMENT operator on the cell values of two input raster datasets.

`BooleanNot <input_raster_or_constant> <output_raster>`

Boolean Or: performs the Boolean OR operator on the cell values of two input raster datasets.

`BooleanOr <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Boolean XOR: performs the Boolean exclusive OR operator on the cell values of two input raster datasets.

`BooleanXOr <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Combinatorial And: performs a combinatorial AND operation on two input raster datasets.

`CombinatorialAnd <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Combinatorial Or: performs a combinatorial OR operation on the cell values of two input raster datasets.

`CombinatorialOr <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Combinatorial XOR: performs a combinatorial exclusive OR operation on the cell values of two input raster datasets.

`CombinatorialXOr <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Equal To: returns 1 for cells where the first raster equals the second raster and 0 if it does not.

`EqualTo <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Greater Than: returns 1 for cells where the first raster dataset is greater than the second raster dataset and returns 0 where it is not.

`GreaterThan <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Greater Than Equal: returns 1 for cells where the first raster dataset is greater than or equal to the second raster dataset and returns 0 where it is not.

`GreaterThanEqual <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Is Null: returns 1 for cells in the input raster dataset that have a value of NoData and returns 0 where they do not.

`IsNull <i n_raster> <out_raster>`

Less Than: returns 1 for cells where the first raster dataset is less than the second raster dataset and returns 0 where it is not.

`LessThan <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>`

Less Than Equal: returns 1 for cells where the first raster dataset is less than or equal to the second raster dataset and returns 0 where it is not.

LessThanEqual <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Not Equal: returns 1 for cells where the first raster dataset is not equal to the second raster dataset and returns 0 where it is not.

NotEqual <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Test: returns 1 for cells that evaluate to true based on a logical expression and returns 0 for cells that evaluate to false.

Test <i n_raster> <where_clause> <out_raster>



Trigonometric (Math) Toolset

Contains tools for the trigonometric functions that are applied on a per cell basis to an input raster dataset.

ACos: calculates the inverse cosine of the input raster dataset or number on a cell-by-cell basis.

ACos <i n_raster_or_constant> <out_raster>

ACosH: calculates the inverse hyperbolic cosine of cells in an input raster dataset.

ACosH <i n_raster_or_constant> <out_raster>

ASin: calculates the inverse sine of cells in an input raster dataset.

ASin <i n_raster_or_constant> <out_raster>

ASinH: calculates the inverse hyperbolic sine of cells in an input raster dataset.

ASinH <i n_raster_or_constant> <out_raster>

ATan: calculates the inverse tangent of cells in an input raster dataset.

ATan <i n_raster_or_constant> <out_raster>

ATan2: calculates the inverse tangent (based on y/x) of cells in an input raster dataset.

ATan2 <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

ATanH: calculates the inverse hyperbolic tangent of cells in an input raster dataset.

ATanH <i n_raster_or_constant> <out_raster>

Cos: calculates the cosine of cells in an input raster dataset.

Cos <i n_raster_or_constant> <out_raster>

CosH: calculates the hyperbolic cosine of cells in an input raster dataset.

CosH <i n_raster_or_constant> <out_raster>

Sin: calculates the sine of cells in an input raster dataset.

`Sin <i n_raster_or_constant> <out_raster>`

Sinh: calculates the hyperbolic sine of cells in an input raster dataset.

`Sinh <i n_raster_or_constant> <out_raster>`

Tan: calculates the tangent of cells in an input raster dataset.

`Tan <i n_raster_or_constant> <out_raster>`

Tanh: calculates the hyperbolic tangent of cells in an input raster dataset.

`Tanh <i n_raster_or_constant> <out_raster>`



Multivariate Toolset

Contains tools to allow the statistical analysis of a series of raster datasets (independent variables) that, based on the values within the rasters, produces a predictable result for some phenomena (the dependent variable).

Band Collection Statistics: calculates the statistics for a set of raster bands.

`BandCollectionStats <i n_raster_bands; i n_raster_bands. . . > <out_stat_file>
{ BRIEF | DETAILED }`

Class Probability: creates probability layers for each class in a signature file.

`ClassProbability <i n_raster_bands; i n_raster_bands. . . > <i n_signature_file>
<out_multiband_raster> { maximum_output_value } { EQUAL | SAMPLE | FILE }
{ i n_a_prio ri _file }`

Create Signatures: creates an ASCII signature file of classes defined by input sample data and a set of raster bands.

`CreateSignatures <i n_raster_bands; i n_raster_bands. . . > <i n_sample_data>
<out_signature_file> { COVARIANCE | MEAN_ONLY } { sample_file_id }`

Dendrogram: constructs a tree diagram showing attribute distances between sequentially merged classes in a signature file.

`Dendrogram <i n_signature_file> <out_dendrogram_file> { VARIANCE | MEAN_ONLY }
{ line_width }`

Edit Signatures: edits and updates a signature file by merging, renumbering, and deleting class signatures.

`EditSignatures <i n_raster_bands; i n_raster_bands. . . > <i n_signature_file>
<i n_signature_remap_file> <out_signature_file> { sample_interval }`

Iso Cluster: uses isodata clustering to determine the characteristics of natural cell groupings in multidimension attribute space.

```
IsoCluster <i_n_raster_bands; i_n_raster_bands... > <out_signature_file>  
<number_classes> {number_iterations} {min_class_size} {sample_interval}
```

Maximum Likelihood Classification: performs a maximum likelihood classification on a set of raster dataset bands.

```
MLClassification <i_n_raster_bands; i_n_raster_bands... > <i_n_signature_file>  
<out_classified_raster> {0.0 | 0.005 | 0.01 | 0.025 | 0.05 | 0.1 | 0.25 | 0.5 |  
0.75 | 0.9 | 0.95 | 0.975 | 0.99 | 0.995} {EQUAL | SAMPLE | FILE}  
{i_n_a_prio_raster} {out_confidence_raster}
```

Principal Components: performs principal components analysis on a set of raster bands.

```
PrincipalComponents <i_n_raster_bands; i_n_raster_bands... >  
<out_multiband_raster> {number_components} {out_data_file}
```



Neighborhood Toolset

Contains tools to calculate a statistic or value based on the values at each processing cell and the values of the cells within an identified neighborhood.

Block Statistics: calculates statistics for a nonoverlapping neighborhood.

```
BlockStatistics <i_n_raster> <out_raster> {neighborhood} {MEAN | MAJORITY |  
MAXIMUM | MEDIAN | MINIMUM | MINORITY | RANGE | STD | SUM | VARIETY}  
{DATA | NODATA}
```

Filter: performs a preset focal filter on a raster.

```
Filter <i_n_raster> <out_raster> {LOW | HIGH} {DATA | NODATA}
```

Focal Flow: determines the flow of the values in the surface raster dataset within each cell's immediate neighborhood.

```
FocalFlow <i_n_surface_raster> <out_raster> {threshold_value}
```

Focal Statistics: calculates a statistic for each raster dataset cell value within a specified neighborhood.

```
FocalStatistics <i_n_raster> <out_raster> {neighborhood} {MEAN | MAJORITY |  
MAXIMUM | MEDIAN | MINIMUM | MINORITY | RANGE | STD | SUM | VARIETY}  
{DATA | NODATA}
```

Line Statistics: calculates a statistic on the attributes of lines in a circular neighborhood around each output cell in a raster dataset.

```
LineStatistics <i_n_polyline_features> <field> <out_raster> {cell_size}  
{search_radius} {MEAN | MAJORITY | MAXIMUM | MEDIAN | MINIMUM | MINORITY | RANGE  
| STD | SUM | VARIETY}
```

Point Statistics: calculates a statistic on the points in a neighborhood outputting a raster dataset.

```
PointStatistics <i n_poi nt_features> <fi el d> <out_raster> {cel l_si ze}  
{nei ghborhood} {MEAN | MAJORI TY | MAXI MUM | MEDI AN | MI NI MUM | MI NORI TY | RANGE |  
STD | SUM | VARI ETY}
```



Overlay Toolset

Contains the tool to create an output surface by adding a series of weighted input raster datasets together.

Weighted Overlay: overlays several rasters using a common scale and weighing each according to its importance.

```
WeightedOverlay <raster {i nfl uence} {fi el d} {remap}; raster {i nfl uence} {fi el d}  
{remap}... > <out_raster>
```



Raster Creation Toolset

Contains tools to create raster datasets based on a constant, random values, or a normal distribution using the existing cell size, extent, and other analysis properties.

Create Constant Raster: creates a raster dataset from a constant value.

```
CreateConstantRaster <out_raster> <constant_val ue> {I NTEGER | FLOAT}  
{cel l_si ze} {extent}
```

Create Normal Raster: creates a raster dataset of random values from a normal distribution.

```
CreateNormalRaster <out_raster> {cel l_si ze} {extent}
```

Create Random Raster: creates a raster dataset of random numbers between 0 and 1.

```
CreateRandomRaster <out_raster> {seed_val ue} {cel l_si ze} {extent}
```



Reclass Toolset

Contains tools to change the values assigned to cells in a thematic raster dataset.

Lookup: creates a new raster dataset by looking up values found in another field in the table of the input raster dataset.

```
Lookup <i n_raster> <l ookup_fi el d> <out_raster>
```

Reclass by ASCII File: reclassifies (or changes) the values of the input cells of a raster dataset by using an ASCII remap file.

```
Recl assByASCI I Fi le <i n_raster> <i n_remap_fi le> <out_raster> {DATA | NODATA}
```

Reclass by Table: reclassifies (or changes) the values of the input cells of a raster dataset by using a remap table.

```
Recl assByTabl e <i n_raster> <i n_remap_tabl e> <from_val ue_fi el d>  
<to_val ue_fi el d> <output_val ue_fi el d> <out_raster> {DATA | NODATA}
```

Reclassify: reclassifies (or changes) the value of the cells in a raster dataset.

Reclassify <i n_raster> <reclass_field> <remap> <out_raster> {DATA | NODATA}

Slice: slices a range of values of the input cells by zones of equal area or equal interval.

Slice <i n_raster> <out_raster> <number_zones> {EQUAL_INTERVAL | EQUAL_AREA | NATURAL_BREAKS} {base_output_zone}



Surface Toolset

Contains tools to analyze the surface of the shapes represented by the raster values.

Aspect: identifies the direction of the maximum rate of change in z-value from each cell.

Aspect <i n_raster> <out_raster>

Contour: creates contours or isolines from a raster dataset surface.

Contour <i n_raster> <out_polyline_features> <contour_interval> {base_contour} {z_factor}

Contour List: creates contours or isolines based on a list of contour values.

ContourList <i n_raster> <out_polyline_features> <contour_values; contour_values...>

Curvature: calculates the curvature of a surface at each cell center.

Curvature <i n_raster> <out_curvature_raster> {z_factor} {out_profile_curve_raster} {out_plan_curve_raster}

Cut/Fill: calculates cut and fill areas.

CutFill <i n_before_surface> <i n_after_surface> <out_raster> {z_factor}

Hillshade: creates a shaded relief raster by considering the illumination angle and shadows.

HillShade <i n_raster> <out_raster> {azimuth} {altitude} {NO_SHADOWS | SHADOWS} {z_factor}

Observer Points: identifies exactly which observer points are visible from each surface location.

ObserverPoints <i n_raster> <i n_observer_point_features> <out_raster> {z_factor} {FLAT_EARTH | CURVED_EARTH} {refractive_coefficient}

Slope: identifies the rate of maximum change in z-value from each cell.

Slope <i n_raster> <out_raster> {DEGREE | PERCENT_RISE} {z_factor}

Viewshed: derives the raster dataset surface locations visible to a set of observation points.

Viewshed <i n_raster> <i n_observer_features> <out_raster> {z_factor} {FLAT_EARTH | CURVED_EARTH} {refractive_coefficient}



Zonal Toolset

Contains tools that can be applied to the zones within the raster dataset.

Tabulate Area: calculates cross-tabulated areas between two datasets.

```
TabulateArea <i n_zone_data> <zone_file_d> <i n_class_data> <class_file_d>  
<out_table> {processing_cell_size}
```

Zonal Fill: fills zones in a dataset using the minimum cell value from another raster dataset along the zone boundary.

```
ZonalFill <i n_zone_raster> <i n_weight_raster> <out_raster>
```

Zonal Geometry: calculates area, perimeter, thickness, or the characteristics of ellipse and records the information as a raster dataset.

```
ZonalGeometry <i n_zone_data> <zone_file_d> <out_raster>  
{AREA | PERIMETER | THICKNESS | CENTROID} {cell_size}
```

Zonal Geometry as Table: calculates area, perimeter, thickness, or the characteristics of ellipse and records the information in an output table.

```
ZonalGeometryAsTable <i n_zone_data> <zone_file_d> <out_table>  
{processing_cell_size}
```

Zonal Statistics: calculates a statistic on values of a raster within the zones of another dataset.

```
ZonalStatistics <i n_zone_data> <zone_file_d> <i n_value_raster> <out_raster>  
{MEAN | MAJORITY | MAXIMUM | MEDIAN | MINIMUM | MINORITY | RANGE | STD | SUM |  
VARIETY} {DATA | NODATA}
```

Zonal Statistics as Table: summarizes values of a raster dataset within the zones of another dataset and reports the results to a table.


```
ZonalStatisticsAsTable <i n_zone_data> <zone_file_d> <i n_value_raster>  
<out_table> {DATA | NODATA}
```

Spatial Statistics Toolbox

Contains statistical tools for analyzing the distribution of geographic features.

Analyzing Patterns Toolset


Contains tools to calculate statistical values used to quantify patterns.

 ❖ **Average Nearest Neighbor:** calculates a nearest neighbor index based on the average distance from each feature to its nearest neighboring feature.

```
AverageNearestNeighbor <input_feature_class> <EUCLIDEAN_DISTANCE | MANHATTAN_DISTANCE> {FALSE | TRUE} {area}
```

 ❖ **High/Low Clustering (Getis–Ord General G):** measures concentrations of high or low values for a study area.


```
HighLowClustering <input_feature_class> <input_field> {FALSE | TRUE} <INVERSE_DISTANCE | INVERSE_DISTANCE_SQUARED | FIXED_DISTANCE_BAND | ZONE_OF_INDIFFERENCE | GET_SPATIAL_WEIGHTS_FROM_FIELD> <EUCLIDEAN_DISTANCE | MANHATTAN_DISTANCE> <NONE | ROW | GLOBAL> <distance_band_or_threshold_distance> {weights_matrix_file}
```

 ❖ **Spatial Autocorrelation (Morans I):** measures spatial autocorrelation based on feature locations and attribute values.


```
SpatialAutocorrelation <input_feature_class> <input_field> {FALSE | TRUE} <INVERSE_DISTANCE | INVERSE_DISTANCE_SQUARED | FIXED_DISTANCE_BAND | ZONE_OF_INDIFFERENCE | GET_SPATIAL_WEIGHTS_FROM_FIELD> <EUCLIDEAN_DISTANCE | MANHATTAN_DISTANCE> <NONE | ROW | GLOBAL> <distance_band_or_threshold_distance> {weights_matrix_file}
```

Mapping Clusters Toolset


Contains tools for cluster analysis, such as identifying the locations of statistically significant hot spots or areas of significant diversity.

 ❖ **Cluster and Outlier Analysis (Anselin Local Morans I):** identifies those clusters of points with values similar in magnitude and clusters of points with very heterogeneous values, within a set of weighted data points.


```
ClustersOutliers <input_feature_class> <input_field> <output_feature_class> <INVERSE_DISTANCE | INVERSE_DISTANCE_SQUARED | FIXED_DISTANCE_BAND | ZONE_OF_INDIFFERENCE | GET_SPATIAL_WEIGHTS_FROM_FIELD> <EUCLIDEAN_DISTANCE | MANHATTAN_DISTANCE> <NONE | ROW | GLOBAL> <distance_band_or_threshold_distance> {weights_matrix_file}
```

 ❖ **Cluster/Outlier Analysis with Rendering:** identifies clusters of points with values similar in magnitude and clusters of points with very heterogeneous values, within a given set of weighted data points and applies a “cold to hot” type of rendering.

```
ClustersOutliersRendered <input_feature_class> <input_field> <output_layer_file> <output_feature_class>
```

 ❖ **Hot Spot Analysis (Getis–Ord G_i^*):** identifies spatial clusters of statistically significant high or low attribute values by calculating the Getis–Ord G_i^* statistics.


```
HotSpot <i nput_feature_cl ass> <i nput_fi el d> <output_feature_cl ass> <I NVERSE
DI STANCE | I NVERSE DI STANCE SQUARED | FI XED DI STANCE BAND | ZONE OF I NDI FFERENCE
| GET SPATI AL WEI GHTS FROM FI LE> <EUCLI DEAN DI STANCE | MANHATTAN DI STANCE> <NONE
| ROW | GLOBAL> <di stance_band_or_threshol d_di stance> {sel f_potentia l _fi el d}
{wei ghts_matri x_fi le}
```

 ❖ **Hot Spot Analysis with Rendering:** calculates G_i^* statistics and applies a “cold to hot” type of rendering to the output Z scores.


```
HotSpotsRendered <i nput_feature_cl ass> <i nput_fi el d> <output_l ayer_fi l e>
<output_feature_cl ass> <di stance_band_or_threshol d_di stance>
```

Measuring Geographic Distributions Toolset


Contains tools to calculate a value that represents a characteristic of the distribution of a set of features, such as the center, compactness, or orientation.

 ❖ **Central Feature:** identifies the most centrally located feature in a point, line, or polygon feature.


```
Central Feature <i nput_feature_cl ass> <output_feature_cl ass> <EUCLI DEAN
DI STANCE | MANHATTAN DI STANCE> {wei ght_fi el d} {sel f_potentia l _wei ght_fi el d}
```

 ❖ **Directional Distribution (Standard Deviational Ellipse):** measures whether a distribution of features exhibits a directional trend.


```
Di recti onal Di stri buti on <i nput_feature_cl ass> <output_el l i pse_feature_cl ass>
<1 STANDARD DEVI ATI ON | 2 STANDARD DEVI ATI ONS | 3 STANDARD DEVI ATI ONS>
{wei ght_fi el d} {case_fi el d}
```

 ❖ **Linear Directional Mean:** identifies the general (mean) direction for a set of vectors.

```
Di recti onal Mean <i nput_feature_cl ass> <output_feature_cl ass>
<ori entati on_onl y> {case_fi el d}
```

 ❖ **Mean Center:** identifies the geographic center (or the center of concentration) for a set of features.

```
MeanCenter <i nput_feature_cl ass> <output_feature_cl ass> {wei ght_fi el d}
{case_fi el d} {di mensi on_fi el d}
```

 ❖ **Standard Distance:** measures the degree to which features are concentrated or dispersed around the points (or feature centroids) in an input feature class.

```
StandardDi stance <i nput_feature_cl ass>
<ouput_standard_di stance_feature_cl ass> <1 STANDARD DEVI ATI ON | 2 STANDARD
DEVI ATI ONS | 3 STANDARD DEVI ATI ONS> {wei ght_fi el d} {case_fi el d}
```




Utilities Toolset

Contains tools to perform a variety of data rendering tasks that can be used in conjunction with other tools in the Spatial Statistics toolbox.

❖ **Calculate Areas:** Calculates AREA values for each feature in a polygon feature class.

`CalculateAreas <input_feature_class> <output_feature_class>`



❖ **Collect Events:** collects event data into weighted point data.

`CollectEvents <input_incident_features> <output_weighted_point_feature_class>`



❖ **Collect Events with Rendering:** collects event data into weighted point data and applies a graduated circle rendering to the count field.

`CollectEventsRendered <input_incident_features> <output_layer_file>
<output_weighted_point_feature_class>`



❖ **Count Rendering:** applies graduated circle rendering to a count type field of a point feature class.

`CountRenderer <input_feature_class> <field_to_render> <output_layer_file>
<number_of_classes> <MANGO | BRIGHT_RED | DARK_GREEN | GREEN | DARK_BLUE |
BRIGHT_PINK | LIGHT_YELLOW | SKY_BLUE> {maximum_field_value}`

❖ **Export Feature Attribute to ASCII:** Exports feature class coordinates and attribute values to a space, comma, or semicolon delimited ASCII text file.

`ExportXYv <input_feature_class> <value_field> <SPACE | COMMA | SEMI-COLON>
<output_ascii_file>`



❖ **Z Score Rendering:** applies a “cold or hot” graduated color rendering to a field of Z scores.

`ZRenderer <input_feature_class> <field_to_render> <output_layer_file>`



3D Analyst Toolbox

Contains tools to create and modify TIN and raster surfaces, then extract information and features from them.



Conversion Toolset

Contains tools used to convert to and from a TIN.

Raster to TIN: creates a TIN from a raster dataset.

`RasterTin <i n_raster> <out_tin> {z_tolerance} {max_points} {z_factor}`

TIN Domain: extracts the interpolation zone from an input TIN into an output feature class.

`TinDomain <i n_tin> <out_feature_class> <LINE | POLYGON>`

TIN Edge: extracts the triangle edge from an input TIN into an output feature class.

`TinEdge <i n_tin> <out_feature_class> {DATA | SOFT | HARD | ENFORCED | REGULAR | OUTSIDE | ALL}`

TIN Node: extracts nodes from an input TIN into an output feature class.

`TinNode <i n_tin> <out_feature_class> {spot_filed} {tag_filed}`

TIN Polygon Tag: extracts polygon tag information from an input TIN into an output feature class.

`TinPolygonTag <i n_tin> <out_feature_class> {tag_filed}`

TIN to Raster: converts a TIN to a raster.

`TinRaster <i n_tin> <out_raster> {FLOAT | INT} {LINEAR | NATURAL_NEIGHBORS} {OBSERVATIONS | CELLSIZE} {z_factor}`

TIN Triangle: extracts triangles as polygons from an input TIN into an output feature class.

`TinTriangle <i n_tin> <out_feature_class> {PERCENT | DEGREE} {z_factor} {HILLSHADE} {tag_filed}`



Functional Surface Toolset

Contains tools to produce output providing knowledge about height information that is contained in surfaces.

Interpolate Shape: interpolates z-values for a feature class based on an underlying surface.

`InterpolateShape <i n_surface> <i n_feature_class> <out_feature_class> {sample_distance} {z_factor}`

Line of Sight: calculates the visibility across a surface between points.

`LineOfSight <i n_surface> <i n_line_feature_class> <out_line_feature_class> {out_obstruction_feature_class} {use_curvature} {use_refraction} {refraction_factor}`

Surface Length: calculates the surface length of each line in a feature class based on a functional surface.

```
SurfaceLength <i n_surface> <i n_feature_class> {out_length_file}
{sample_distance} {z_factor}
```

Surface Spot: calculates surface values for each point of a point feature class by interpolating from a functional surface.

```
SurfaceSpot <i n_surface> <i n_feature_class> {out_spot_file} {z_factor}
```

Surface Volume: calculates the area and volume of a functional surface above or below a given reference plane.

```
SurfaceVolume <i n_surface> {out_text_file} {ABOVE | BELOW} {base_z} {z_factor}
```



Raster Interpolation Toolset

Contains tools to create a raster surface from point features.

IDW: interpolates a surface from points using an inverse distance weighted technique.

```
IDW <i n_point_features> <z_file> <out_raster> {cell_size} {power}
{search_radius} {i n_barrier_polyline_features}
```

Krige: interpolates a raster dataset from a set of points using kriging.

```
Kriging <i n_point_features> <z_file> <out_surface_raster>
<semi_Variogram_props> {cell_size} {search_radius}
{out_variance_predictor_raster}
```

Natural Neighbor: interpolates a surface from points using a natural neighbor technique.

```
NaturalNeighbor <i n_point_features> <z_file> <out_raster> {cell_size}
```

Spline: interpolates a surface from points using a minimum curvature spline technique.

```
Spline <i n_point_features> <z_file> <out_raster> {cell_size}
{REGULARIZED | TENSION} {weight} {number_points}
```

Topo to Raster: generates a hydrologically correct raster dataset of elevation.

```
TopoToRaster <feature_layer{File} {Type}; feature_layer{File} {Type} . . . >
<out_surface_raster> {cell_size} {extent} {Margin} {minimum_z_value}
{maximum_z_value} {ENFORCE | NO_ENFORCE | ENFORCE_WITH_SINK}
{CONTOUR | SPOT} {maximum_iterations} {roughness_penalty}
{discrete_error_factor} {vertical_standard_error} {tolerance_1}
{tolerance_2} {out_stream_features} {out_sink_features}
{out_diagnostics_file} {out_parameter_file}
```

Topo to Raster by File: generates a hydrologically correct raster dataset of elevation.

```
TopoToRasterByFile <i n_parameter_file> <out_surface_raster>
{out_stream_features} {out_sink_features}
```

Trend: interpolates a surface from points using a trend technique.

Trend <i n_poi nt_features> <z_fi el d> <out_raster> {cel l_si ze} {order}
{LI NEAR | LOGI STIC}



Raster Math Toolset

Contains the tools used to perform mathematics with raster datasets.

Divide: divides the values of two inputs on a cell-by-cell basis.

Di vi de <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Float: converts each cell value of a raster dataset into a floating-point value.

Fl oat <i n_raster_or_constant> <out_raster>

Int: converts each cell value of a raster dataset into an integer value through truncation.

Int <i n_raster_or_constant> <out_raster>

Minus: subtracts the values of the second input from the values of the first input.

Mi nus <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Plus: adds the values of two raster datasets on a cell-by-cell basis.

Pl us <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>

Times: multiplies the values in raster datasets on a cell-by-cell basis.

Ti mes <i n_raster_or_constant1> <i n_raster_or_constant2> <out_raster>



Raster Reclass Toolset

Contains tools to alter the classes within a raster dataset.

Lookup: creates a new raster dataset by looking up values found in another field in the table of the input raster dataset.

Look up <i n_raster> <l ookup_fi el d> <out_raster>

Reclass by ASCII File: reclassifies (or changes) the values of the input cells of a raster dataset by using an ASCII remap file.

Recl assByASCI I Fi l e <i n_raster> <i n_remap_fi l e> <out_raster> {DATA | NODATA}

Reclass by Table: reclassifies (or changes) the values of the input cells of a raster dataset by using a remap table.

Recl assByTabl e <i n_raster> <i n_remap_tabl e> <from_val ue_fi el d>
<to_val ue_fi el d> <output_val ue_fi el d> <out_raster> {DATA | NODATA}

Reclassify: reclassifies (or changes) the value in a raster dataset.

Recl assi fy <i n_raster> <recl ass_fi el d> <remap> <out_raster> {DATA | NODATA}

Slice: slices a range of values of the input cells by zones of equal area or equal interval.

`Slice <i n_raster> <out_raster> <number_zones> {EQUAL_INTERVAL | EQUAL_AREA | NATURAL_BREAKS} {base_output_zone}`



Raster Surface Toolset

Contains tools to analyze the surface of a raster dataset.

Aspect: identifies the direction of the maximum rate of change in z-value from each cell.

`Aspect <i n_raster> <out_raster>`

Contour: creates contours or isolines from a raster dataset surface.

`Contour <i n_raster> <out_polygon_features> <contour_interval> {base_contour} {z_factor}`

Contour List: creates contours or isolines from a list of contour values.

`ContourList <i n_raster> <out_polygon_features> <contour_values; contour_values...>`

Curvature: calculates the curvature of a surface at each cell center.

`Curvature <i n_raster> <out_curvature_raster> {z_factor} {out_profile_curve_raster} {out_plan_curve_raster}`

Cut/Fill: calculates cut and fill areas.

`CutFill <i n_before_surface> <i n_after_surface> <out_raster> {z_factor}`

Hillshade: creates a shaded relief raster dataset by considering the illumination angle and shadows.

`HillShade <i n_raster> <out_raster> {azimuth} {altitude} {NO_SHADOWS | SHADOWS} {z_factor}`

Observer Points: identifies exactly which observer points are visible from each surface location within the raster dataset.

`ObserverPoints <i n_raster> <i n_observer_point_features> <out_raster> {z_factor} {FLAT_EARTH | CURVED_EARTH} {refractive_coefficient}`

Slope: identifies the rate of maximum change in z-value from each cell.

`Slope <i n_raster> <out_raster> {DEGREE | PERCENT_RISE} {z_factor}`

Viewshed: determines the raster dataset surface locations visible to a set of observer features.

`Viewshed <i n_raster> <i n_observer_features> <out_raster> {z_factor} {FLAT_EARTH | CURVED_EARTH} {refractive_coefficient}`



TIN Creation Toolset

Contains tools to create and edit TINs.

Create TIN: creates an empty TIN with an extent based on an input geodataset or coordinates.

CreateTin <out_tin> <spatial_reference>

Edit TIN: edits the TIN geoprocessing function.

EditTin <in_tin> <in_feature_class {Height_Field} {Tag_Field} {SF_Type} {Use_Z}; in_feature_class {Height_Field} {Tag_Field} {SF_Type} {Use_Z}... > {out_tin}



TIN Surface Toolset

Contains tools to analyze the surface of a TIN dataset.

Interpolate Polygon to Multipatch: converts a polygon feature class to a multipatch feature class whose heights are based on a surface.

InterpolatePolyToPatch <in_tin> <in_feature_class> <out_feature_class> {max_strip_size} {z_factor}

TIN Aspect: calculates aspect for the surface using each triangle in a TIN.

TinAspect <in_tin> <out_feature_class> {class_breaks_table} {aspect_field}

TIN Contour: derives contour lines for a surface from a TIN.

TinContour <in_tin> <out_feature_class> <interval> {base_contour} {contour_field} {contour_field_precision} {index_interval} {index_interval_field} {z_factor}

TIN Slope: calculates the slope of the surface as the maximum rate of change in elevation across each triangle.

TinSlope <in_tin> <out_feature_class> {PERCENT | DEGREE} {class_breaks_table} {slope_field} {z_factor}

Index

3D Analyst toolbox 69

A

Abs command 55

Absolute value. *See* Abs command

ACos command 59

ACosH command 59

AddCadFields command 9

AddCodedValueToDomain command 23

AddFeatureClassToTopology command 33

AddField command 26

AddFieldToAnalysisLayer command 43

AddIndex command 28

Adding. *See also* Plus command

 attribute index 28

 CAD fields 9

 coded value to domain 23

 fields 26

 indexes 28

 items 18

 joins 28

 rules to topology 34

 spatial indexes 28

 xy coordinates 19, 25

AddItem command 18

AddJoin command 28

AddLocations command 43

Address

 geocoding 37

 standardizing 37

Address locator

 creating 37

 deleting 37

AddressBuild. *See* RebuildGeocodingIndex command

AddressCreate. *See* CreateAddressLocator command

AddressMatch. *See* GeocodeAddresses command

AddressParse. *See* StandardizeAddresses command

AddRouteMeasure. *See* LocateFeaturesAlongRoutes command

AddRuleToTopology command 34

AddSpatialIndex command 28

AddSubtype command 32

AddXY command 19, 25

Adjust. *See* Transform command

ADRGGrid. *See* CopyRaster command

Advanced Tiger Conversion. *See* TigerTool command

Aggregate 17

 command 50

 toolset 16

AggregatePolygons command 17

Alias

 described 2

 list 2

AlterVersion command 34

Analysis

 toolbox 3

 toolset 13, 43

Analyze command 33

Analyzing

 clusters and outliers 65

 hot spots 66

Analyzing Patterns toolset 65

Annotation

 importing from CAD 10

 importing from coverage 11

Append command 16, 27

AppendAnnotations command 24

ArcDLG command 14

ArcDXF. *See* ExportCAD command

ArcIGDS. *See* ExportCAD command

ArcInfo workspace

 creating 35

ArcRoute command 16

ArcS57 command 15

ArcSDE

 checking in 23

 checking out 23

ArcSection. *See* CreateRoutes command

ArcShape. *See* FeatureClassToShapefile command

ArcTiger. *See* TigerArc command; TigerTool command

ArcTIN. *See* Editing: TINs

Area

 calculating 67

 creating polygon features 26

AreaAggregate. *See* AggregatePolygons command

Arguments 2

ASCII

 converting to raster 11

ASCIIGrid. *See* ASCII: converting to raster

ASCIIToRaster command 11

ASin command 59

ASinH command 59

Aspect command 63, 72

- AssignDefaultToField command 26
- AssignDomainToField command 23
- Assigning
 - domain to field 23
- ATan command 59
- ATan2 command 59
- ATanH command 59
- Attributes
 - joining tables 18
- Autocorrelation 65
- AutomateGeocodingIndexes command 37
- AverageNearestNeighbor command 65

- B**
- Band Collection Statistics. *See* BandCollectionStats command
- BandCollectionStats command 60
- Base 10. *See* Log10 command. *See* Exp10 command
- Base 2. *See* Exp2 command; Log2 command
- Base e. *See* Exp command
- Basic Tiger Conversion. *See* TigerArc: command
- Basin command 51
- BatchBuildPyramids command 30
- BatchCalculateStatistics command 31
- Bitwise
 - and 57
 - exclusive or 57
 - left shift 57
 - not 57
 - or 57
 - right shift 57
 - toolset 57
- Bitwise Complement. *See* BitwiseNot command
- Bitwise Exclusive Or. *See* BitwiseXOr command
- BitwiseAnd command 57
- BitwiseLeftShift command 57
- BitwiseNot command 57
- BitwiseOr command 57
- BitwiseRightShift command 57
- BitwiseXOr command 57
- BlockMajority. *See* BlockStatistics command
- BlockMax. *See* BlockStatistics command
- BlockMean. *See* BlockStatistics command
- BlockMedian. *See* BlockStatistics command
- BlockMin. *See* BlockStatistics command
- BlockMinority. *See* BlockStatistics command
- BlockRange. *See* BlockStatistics command
- BlockStatistics command 61
- BlockStd. *See* BlockStatistics command
- BlockSum. *See* BlockStatistics command
- BlockVariety. *See* BlockStatistics command
- Boolean
 - and 57
 - exclusive or 58
 - not 57
 - or 58
- Boolean Complement. *See* BooleanNot command
- BooleanAnd command 57
- BooleanNot command 57
- BooleanOr command 58
- BooleanXOr command 58
- BoundaryClean command 50
- Buffer command 4, 14
- Build command 19
- Building
 - pyramids 30, 31
 - simplification 17
- BuildNetwork command 44
- BuildPyramids command 31
- BuildSta. *See* CellStatistics command

- C**
- CAD
 - annotation
 - importing 10
 - importing 11, 12
- CalculateAreas command 67
- CalculateDefaultClusterTolerance command 24
- CalculateDefaultGridIndex command 24
- CalculateField command 26
- CalculateLocationFields command 43
- CalculateStatistics command 31
- Calculating
 - cluster tolerance 24
 - fields 26
 - kernel density 47
 - spatial grid index 24
 - statistics 29, 31
 - summary statistics 5
- CalibrateRoutes command 41
- CAnd. *See* CombinatorialAnd command
- Cartography toolbox 7
- Cell groupings
 - using isodata clustering 61
- CellStatistics command 53

Centerline	
creating	17
CentralFeature command	66
CentroidLabels. <i>See</i> FeatureToPoint command	
ChangePrivileges command	33
CheckGeometries command	25
CheckIn command	23
CheckInDelta command	23
Checking in	
to ArcSDE	23
Checking out	
from ArcSDE	23
CheckOut command	23
ClassProb. <i>See</i> ClassProbability command	
ClassProbability command	60
ClassSig. <i>See</i> CreateSignatures command	
Clean command	19
Cleaning	
boundaries	50
Clip command	3, 13, 31
Cluster and outlier analysis	
analysis	65
Cluster tolerance	
calculating	24
setting	34
Clustering	
High/Low	65
ClustersOutliersRendered command	65
CollapseDualLineToCenterline command	17
CollectEvents command	67
CollectEventsRendered command	67
Combinatorial	
and	58
exclusive or	58
or	58
CombinatorialAnd command	58
CombinatorialOr command	58
CombinatorialXOr command	58
Combine command	54
Combining features. <i>See</i> Append command	
Compact command	21, 23
Composite Feature toolset	16
CompositeBands command	31
Compress command	21, 23
Con command	47
Conditional toolset	47
Conflicts	
finding	17
Contour command	63, 72
Contouring	
from a TIN	73
ContourList command	63, 72
Conversion	
toolbox	9
toolset	14, 69
ConvertImage. <i>See</i> CopyRaster command	
ConvertRemap. <i>See</i> ReclassByASCIIFile command	
Coordinates	
adding xy	19
Copy command	27
CopyFeatures command	25
Copying	
features	25
raster catalog items	31
raster datasets	31
rows in a table	33
tables	11
CopyRaster command	31
CopyRasterCatalogItems command	31
CopyRows command	33
COR. <i>See</i> CombinatorialOr command	
Corridor command	48
Cos command	59
CosH command	59
Cost	
calculating least accumulative distance	48
least accumulative	48
least accumulative distance	
path	49
least accumulative path	
define neighbor cell	48
least accumulative source	49
retracing path	49
CostAllocation command	48
CostBackLink command	48
CostDistance command	48
CostPath command	48
Counting	
table rows. <i>See</i> GetCount command	
CountRenderer command	67
Coverages	
combining	16
creating	19
erasing	13
exporting as text file	15
exporting to DLG	14
generating	15

Coverages (continued)	
generating topology	19
importing from DLG	15
projecting	18
setting tolerances	19
toolbox	13
transforming	18
updating	14
updating attribute table and topology	19
Create command	19
CreateAddressLocator command	37
CreateArcInfoWorkspace command	35
CreateCADXData command	9
CreateConstantRaster command	62
CreateDomains command	24
CreateFeatureClass command	25
CreateFeatureDataset command	35
CreateFolder command	35
CreateGeodatabaseRasterDataset command	31
CreateLabels command	19
CreateNormalRaster command	62
CreatePersonalGDB command	35
CreateRandomRaster command	62
CreateRasterCatalog command	31
CreateRelationshipClass command	32
CreateRoutes command	41
CreateSignatures command	60
CreateSpatialReference command	30
CreateTable command	33
CreateTIN command	73
CreateTopology command	34
CreateTurnFeatureClass command	44
CreateVersion command	34
CreateWorkspace. <i>See</i> CreateArcInfoWorkspace command	
Creating	
address locators	37
ArcInfo workspaces	35
CAD x data	9
centerlines	17
coverages	19
domains	24
envelopes	25
feature classes	25
folders	35
labels	19
layers	29
multiband raster datasets	31
Creating (continued)	
personal geodatabases	35
raster catalogs	31
raster datasets	31, 62
spatial references	30
tables	33
tables from domains	24
TINs	73
topology	34
versions	34
CuldeSacMasks command	7
Curvature command	63, 72
CutFill command	63, 72
CXOr. <i>See</i> CombinatorialXOr command	
D	
DarcyFlow command	51
DarcyVelocity command	51
Data	
selecting	27
Data Interoperability toolbox	21
Data Management toolbox	23
Data Management toolset	16
Database toolset	23
Datasets	
copying	27
deleting	27
renaming	27
DBaseInfo. <i>See</i> TableToGeodatabase command	
DBMSInfo. <i>See</i> TableToGeodatabase command	
DeautomateGeocodingIndexes command	37
DefineProjection command	18, 29
Delete command	27
DeleteAddressLocator command	37
DeleteCodedValueFromDomain command	24
DeleteDomain command	24
DeleteFeatures command	25
DeleteField command	26
DeleteRasterCatalogItems command	31
DeleteRows command	33
DeleteVersion command	35
Deleting	
address locators	37
coded values from domains	24
domains	24
feature classes	27
feature datasets	27

Deleting (continued)	
features	25
fields in a table	26
geodatabase versions	35
items in a raster catalog	31
raster datasets	27
rows in a table	33
tables	27
Delta geodatabase	
checking in	23
checking out	23
DEM	
converting to raster	11
DEMGrid. <i>See</i> DEM: converting to raster	
DEMLattice. <i>See</i> DEM: converting to raster	
DEMToRaster command	11
Dendrogram command	60
Density	
calculating kernel	47
calculating using line features	47
calculating using point features	47
toolset	47
DFADArc. <i>See</i> Feature class: converting to coverage	
DirectionalDistribution command	66
DirectionalMean command	66
Directions command	43
Disconnected Editing toolset	23
Dissolve command	17, 27
DissolveRouteEvents command	41
Dissolving	
route events	41
Distance toolset	48
Divide command	55, 71. <i>See also</i> Mod command
DLG	
importing to coverage	15
DLGArc command	15
Domain	
creating	24
creating from table	24
deleting	24
removing from field	24
setting value for range	24
toolset	23
DomainToTable command	24
Drainage. <i>See</i> Hydrology toolset	
DropIndex command . 18. <i>See also</i> RemoveIndex command	
DropItem command	18

Dropping	
indexes	18
items	18
DTEDGrid. <i>See</i> CopyRaster command	
DXFArc. <i>See</i> ImportCAD command	
DXFINFO. <i>See</i> ImportCAD command	

E

Editing	
TINs	73
EditSig. <i>See</i> EditSignatures command	
EditSignatures command	60
Eliminate command	17, 27
Ellipse	2
Envelopes	
creating polygons from	25
EqualityTest. <i>See</i> EqualTo command	
EqualTo command	58
EqualToFrequency command	54
Erase command	3, 13
ETAKArc. <i>See</i> FeatureClassToCoverage command	
EucAllocation command	48
EucDirection command	48
EucDistance command	48
Euclidean	
allocating	48
direction	48
distance	48
Euclidean Allocation. <i>See</i> EucAllocation command.	
Euclidean Direction. <i>See</i> EucDirection command	
Euclidean Distance. <i>See</i> EucDistance command.	
EventArc. <i>See</i> MakeRouteEventLayer command	
EventPoint. <i>See</i> MakeRouteEventLayer command	
Events	
dissolving for routes	41
transforming routes	42
EventTransform. <i>See</i> TransformRouteEvents command	
Example tool	2
Exclusive arguments	2
Exp command	55
Exp10 command	55
Exp2 command	55
Expand command	50
Exponential. <i>See also</i> Exp command	
base 10	55
base 2	55
base e	55
Export command	15

- ExportCAD command 9
 - Exporting
 - feature attribute to ASCII 67
 - to CAD 9
 - to Delta geodatabase 21, 23
 - to Interchange file 15
 - to S57 15
 - to SDTS 15
 - to VPF 15
 - ExportToDelta command 23
 - ExportXYv command 67
 - Extract toolset 3, 13
 - ExtractByAttributes command 49
 - ExtractByCircle command 49
 - ExtractByMask command 49
 - ExtractByPoints command 49
 - ExtractByPolygon command 50
 - ExtractByRectangle command 50
 - Extraction toolset 49
 - ExtractValuesToPoints command 50
- F**
- Feature class
 - combining 27
 - converting to coverage 10
 - converting to feature class 10
 - converting to shapefile 12
 - copy to geodatabase 10
 - creating 25
 - intersecting 3
 - remove from topology 34
 - toolset 24
 - updating attributes and geometry 4
 - Feature toolset 25, 29
 - FeatureClassToCoverage command 10
 - FeatureClassToFeatureClass command 10
 - FeatureClassToGeodatabase command 10
 - FeatureClassToShapefile command 12
 - FeatureEnvelopeToPolygon command 25
 - Features
 - breaking apart 26
 - buffering 4
 - center of concentration 66
 - centrally located 66
 - concentration or dispersion 66
 - converting to raster 11
 - copying 25
 - Features (continued)
 - dissolving 27
 - eliminating 27
 - erasing 3
 - extracting 3
 - locating along routes 41
 - splitting 3
 - unioning 4
 - FeatureToLine command 25
 - FeatureToPoint command 26
 - FeatureToPolygon command 26
 - FeatureToRaster command 11
 - FeatureVerticesToPoints command 26
 - Field toolset 26
 - Fields
 - adding 26
 - assigning default to 26
 - calculating 26
 - deleting 26
 - Fill command 51
 - Filter command 61
 - FindConflicts command 17
 - Finding conflicts 17
 - Flip command 30
 - Float
 - command 55, 71
 - convert to raster 11
 - FloatGrid. *See* FloatToRaster command
 - FloatToRaster command 11
 - Flow. *See* DarcyFlow command
 - FlowAccumulation command 52
 - FlowDirection command 52
 - FlowLength command 52
 - FMod. *See* Mod command
 - FocalFlow command 61
 - FocalMajority. *See* FocalStatistics command
 - FocalMax. *See* FocalStatistics command
 - FocalMean. *See* FocalStatistics command
 - FocalMedian. *See* FocalStatistics command
 - FocalMin. *See* FocalStatistics command
 - FocalMinority. *See* FocalStatistics command
 - FocalRange. *See* FocalStatistics command
 - FocalStatistics command 61
 - FocalStd. *See* FocalStatistics command. *See* FocalStatistics command
 - FocalSum. *See* FocalStatistics command
 - FocalVariety. *See* FocalStatistics command
 - Folder
 - creating 35

Frequency command	5
From Coverage toolset	14
From Raster toolset	9
Functional Surface toolset	69

G

GALayerToGrid command	39
General toolset	27
Generalization toolset	17, 27, 50
Generalize. <i>See</i> Line: simplification	
Generate command	15
GeocodeAddresses command	37
Geocoding	
addresses	37
deautomating indexes	37
toolbox	37
Geocoding index	
rebuilding	37
Geodatabase	
creating	35
Geometries	
checking	25
repairing	26
Geostatistical Analyst toolbox	39
GetCount command	33
Getis-Ord General G	65
Getis-Ord Gi*	66
GreaterThan command	58
GreaterThanEqual command	58
GreaterThanFrequency command	54
GridASCII. <i>See</i> RasterToASCII command	
GridClip. <i>See</i> ExtractByRectangle command	
GridComposite. <i>See</i> CompositeBands command	
GridFlip. <i>See</i> Flip command	
GridFloat. <i>See</i> RasterToFloat command	
GridImage. <i>See</i> CopyRaster command	
GridLine. <i>See</i> RasterToPolyline command	
GridLineShape. <i>See</i> RasterToPolyline command	
GridMirror. <i>See</i> Mirror	
GridPoint. <i>See</i> RasterToPoint command	
GridPointShape. <i>See</i> RasterToPoint command	
GridPoly. <i>See</i> RasterToPolygon command	
GridRotate. <i>See</i> Rotate command	
GridShape. <i>See</i> RasterToPolygon command	
GridShift. <i>See</i> Shift command	
GridWarp. <i>See</i> Warp command	

Groundwater	
calculating steady flow	51
dispersion of point	51
toolset	51

H

HighestPosition command	54
HighLowClustering command	65
Hillshade command	63, 72
HotSpot command	66
HotSpotsRendered command	66
Hydrodynamic dispersion. <i>See</i> PorousPuff command	
Hydrology toolset	51

I

IDEdit command	19
Identity command	3, 13
IDs	
updating	19
IDW command	53, 70
IGDSArc. <i>See</i> ImportCAD command	
IGDSInfo. <i>See</i> ImportCAD command	
ImageGrid. <i>See</i> CopyRaster command	
Import command	16
ImportCAD command	11
ImportCADAnnotations command	10
ImportCoverageAnnotations command	11
Importing	
CAD annotation	10
from CAD	11, 12
from Interchange file	16
from S57	16
from SDTS	16
from VPF	16
IncreaseMaximumEdges command	44
Index	
adding	28
adding spatial index	28
automating geocoding index	37
dropping	18
item	18
rebuilding for geocoding	37
removing	28
Indexes toolset	18, 28
IndexItem command . 18. <i>See also</i> AddIndex command	

- Info
 - joining tables 18
- InfoDBASE. *See* TableToDBASE command
- Int command 55, 71
- Integer
 - converting from. *See* Int command
- Integrate command 25
- Interoperability tools 21
- InterpolateShape command 69
- Interpolating
 - a raster dataset 53, 70
 - raster surfaces 53, 70
 - z-values 69
- Interpolation
 - calculating the trend on a point dataset 53
 - of a point coverage 14
 - toolset 53
- Intersect command 3, 13
- Intersecting
 - coverages 13, 14
- Inverse distance weighted 53, 70
- IsNull command 58
- IsoCluster command 61
- Item
 - adding 18
 - dropping 18
 - indexing 18
- Items toolset 18

- J**
- Joining
 - attribute tables 18
 - info tables 18
- JoinItem command 18
- Joins
 - adding 28
 - removing 28
 - toolset 18, 28

- K**
- KernelDensity command 47
- Kill. *See* Delete command
- Krige command 70
- Kriging. *See* Krige command

- L**
- Labels
 - creating 19
- LatticeClip. *See* ExtractByPolygon command
- LatticeContour. *See* Contour command
- Layer
 - making raster 29
 - making raster catalog 29
- Layers and Table Views toolset 28
- LessThan command 58
- LessThanEqual command 59
- LessThanFrequency command 54
- Linear directional mean 66
- Linear Referencing toolbox 39, 41
- LineDensity command 47
- LineGrid. *See* FeatureToRaster command
- LineOfSight command 69
- Lines
 - converting
 - coverage to region 16
 - coverage to route 16
 - creating from polygon 25
 - generating for polygon 26
 - simplifying 17
 - splitting at vertices 26
- LineStatistics command 61
- LineStats. *See* LineStatistics command
- Ln command 55
- Local toolset 53
- LocateFeaturesAlongRoutes command 41
- Locating
 - features along routes 41
- Log10 command 55
- Log2 command 56
- Logarithm. *See also* Ln command
 - base 10 55
 - base 2 56
 - natural 55
- Logical toolset 57
- Lookup command 62, 71
- LowestPosition command 54

- M**
- Majority. *See* CellStatistics command
- MajorityFilter command 50
- MakeClosestFacilityLayer command 43
- MakeFeatureLayer command 28

- MakeODCostMatrixLayer command 43
 - MakeQueryTable command 28
 - MakeRasterCatalogLayer command 29
 - MakeRasterLayer command 29
 - MakeRouteEventLayer command 41
 - MakeRouteLayer command 44
 - MakeServiceAreaLayer command 44
 - MakeTableView command 29
 - MakeXYEventLayer command 29
 - Management toolbox 21
 - Map Algebra
 - statements 55
 - Map Algebra toolset 55
 - Mapping Clusters toolset 65
 - Mask
 - creating cul-de-sac 7
 - creating from feature outline 7
 - Masking toolset 7
 - MasksFromFeatureOutlineMasks command 7
 - MasksFromIntersectingLayersMasks command 7
 - Math toolset 55
 - Max. *See* CellStatistics command
 - Maximum Likelihood Classification. *See* MLClassify command
 - Mean. *See* CellStatistics command
 - MeanCenter command 66
 - MeasureRoute. *See* CreateRoutes command
 - Measuring Geographic Distributions toolset 66
 - Med. *See* CellStatistics command
 - Min. *See* CellStatistics command
 - Minimum curvature. *See* Spline command
 - Minority. *See* CellStatistics command
 - Minus command 56
 - Mirror 30
 - MLClassify command 61
 - Mod command 56
 - Model
 - described 1
 - Morans I 65
 - Mosaic command 32
 - MosaicToNewRaster command 32
 - Multiband raster dataset
 - creating 31
 - MultiOutputMapAlgebra command 55
 - Multipart feature
 - converting to singlepart feature 26
 - MultipartToSinglepart command 26
 - MultipleRingBuffer command 4
 - Multiply. *See* Times command
 - Multivariate toolset 60
- ## N
- NaturalNeighbor command 53, 70
 - Near command 4, 14
 - Nearest neighbor 65
 - Negate command 56
 - Neighborhood toolset 61
 - Network Analyst toolbox 43
 - Network Dataset toolset 44
 - Nibble command 50
 - Nodes
 - renumber 19
 - Normal. *See* CreateNormalRaster command
 - NotEqual command 59
- ## O
- ObserverPoints command 63, 72
 - Outlier and cluster analysis 65
 - Overlay toolset 3, 13, 62
 - Overlaying
 - route events 41
 - OverlayRouteEvents command 41
- ## P
- Parameters 2
 - optional 2
 - required 2
 - Particle
 - calculating path 51
 - ParticleTrack command 51
 - PathAllocation command 49
 - PathBackLink command 49
 - PathDistance command 49
 - Pick command 47
 - PivotTable command 33
 - Plus command 56, 71
 - PointDensity command 47
 - PointDistance command 4, 14
 - PointGrid. *See* FeatureToRaster command
 - PointNode command 14
 - Points
 - calculating at surface. *See* SurfaceSpot command.
 - calculating distance between 4, 14
 - calculating distance from 4, 14

- Points
 - computing distance to 14
 - creating from features 26
 - creating from vertices 26
 - transferring attributes to node features 14
- PointStatistics command 62
- Polygons
 - aggregating 17
 - converting coverages to regions 17
 - converting to lines 26
 - merging 17
- PolygonToLine command 26
- PolyGrid. *See* FeatureToRaster command
- PolyRegion command 17
- Popularity command 54
- PopulateAlternateIDFields command 44
- PorousPuff command 51
- PostVersion command 35
- Power command 56
- PrincipalComponents command 61
- PrinComp. *See* PrincipalComponents command
- Project command 18, 30
- ProjectDefine. *See* DefineProjection command
- ProjectGrid. *See* ProjectRaster command
- Projecting rasters 30
- Projection
 - defining 18, 29
- Projections and Transformations toolset 29
- Projections toolset 18
- ProjectRaster command 30
- Proximity toolset 4, 14
- Pyramids
 - building 30, 31

- Q**
- QuickExport command 21
- QuickImport command 21

- R**
- Range. *See* CellStatistics command
- Rank command 54
- Raster catalogs
 - copying items 31
 - creating 31
 - deleting items 31
 - making layers 29
- Raster Creation toolset 62
- Raster datasets
 - calculating accumulated flow 52
 - calculating aspect 72
 - calculating band statistics 60
 - calculating drainage basins 51
 - calculating flow direction 52
 - calculating flow distance 52
 - calculating slope 63, 72
 - calculating statistics 61, 62
 - calculating surface length 70
 - calculating surface values 70
 - calculating surface visibility 69
 - calculating surface volume 70
 - calculating weighted overlay 62
 - changing cell values. *See* Reclassify command. *See* Reclassify command
 - changing scale 30
 - clipping 31
 - combining 54
 - converting from TINs 69
 - converting to ASCII 9
 - converting to linear network features 52
 - converting to points 9
 - converting to polygons 9
 - converting to polylines 9
 - converting values to float 9, 55
 - converting values to integer 55. *See also* Int command. *See also* Int command
 - copying 31
 - creating 31, 62, 71
 - creating ASCII file of selected cells 50
 - creating contours 72
 - creating surface. *See* Raster Interpolation toolset. *See* Raster Interpolation toolset
 - creating TINs 69
 - creating using elevation data 53
 - extracting cell values
 - using points 50
 - extracting cells
 - using a mask 49
 - using attributes 49
 - using points 49
 - within a circle 49
 - within a polygon 50
 - within a rectangle 50
 - generating for elevation 70

Raster datasets (continued)	
interpolating	53, 70, 71
majority filter	50
making layers	29
mosaicking	32
performing a classification	61
principal components	61
projecting	30
reclassifying	62, 63, 71, 72
recording least cost path	48
reducing resolution	50. <i>See also</i> Aggregate
replacing cell values. <i>See</i> Nibble command	
resampling	32
rotating	30
surface	
calculating viewable areas	72
removing sinks	51
surface flow values	61
transforming	30
viewshed	63
Raster Interpolation toolset	70
Raster Math toolset	71
Raster Reclass toolset	71
Raster Surface toolset	72
Raster toolset	30
RasterTIN command	69
RasterToASCII command	9
RasterToFloat command	9
RasterToPoint command	9
RasterToPolygon command	9
RasterToPolyline command	9
RebuildGeocodingIndex command	37
Rebuilding	
geocoding indexes	37
Reclass. <i>See</i> ReclassByTable command	
toolset	62
ReclassByASCIIFile command	62, 71
ReclassByTable command	62, 71
Reclassify command	63, 71
ReconcileVersion command	35
Reconciling	
versions	35
Rectify. <i>See</i> ProjectRaster command	
Reducing database sizes	23
RegionClass command	16
RegionGroup command	51
RegionPoly command	17

Regions	
converting to polygon coverages	17
grouping	51
RegisterAsVersioned command	35
Relationship Classes toolset	32
RemoveDomainFromField command	24
RemoveFeatureClassFromTopology command	34
RemoveIndex command	28
RemoveJoin command	28
RemoveRuleFromTopology command	34
RemoveSpatialIndex command	28
RemoveSubtype command	32
Removing	
attribute indexes	28
feature classes from topologies	34
joins	28
rules from topologies	34
spatial indexes	28
Rename command	27
Rendering	
and collecting events	67
counts	67
with hot spot analysis	66
Renode command	19
Renumbering	
nodes	19
RepairGeometries command	26
Resample command	32
Rescale command	30
Reselect command	13. <i>See also</i> Select command
Rotate command	30
RoundDown command	56
RoundUp command	56
RouteARC. <i>See</i> CreateRoutes command	
Routes	
creating from line coverages	16
locating features along	41
overlaying events	41
transforming events	42
Rows	
copying	33
deleting	33
Rubber sheeting rasters	30, 31
Rule	
adding to topology	34
removing from topology	34

S

- S57Arc command 16
- Sample command 50
- SaveToLayerFile command 29
- Saving
 - layer files 29
- Scale
 - changing 30
- Script
 - described 1
- SDTSExport command 15
- SDTSImport command 16
- Select command 3
- SelectBox. *See* ExtractByRectangle command
- SelectCircle. *See* ExtractByCircle command
- SelectData command 27
- Selecting 13
 - by attributes 29
 - by locations 29
 - data 27
 - layers by attributes 29
 - layers by locations 29
- SelectLayerByAttribute command 29
- SelectLayerByLocation command 29
- SelectMask. *See* ExtractByMask command
- SelectPoint. *See* ExtractByPoints command
- SelectPolygon command. *See* ExtractByPolygon command
- SetCADAlias command 10
- SetClusterTolerance command 34
- SetDefaultSubtype command 33
- SetNull command 47
- SetSubtypeField command 33
- Setting
 - CAD aliases 10
- SetValueForRangeDomain command 24
- Shaded relief. *See* Hillshade command
- ShapeGrid. *See* FeatureToRaster command
- Shift command 30
- Shrink command 51
- Signatures files
 - creating 60
 - creating probability layers 60
 - editing 60
- SimplifyBuilding command 17
- Simplifying
 - buildings 17
 - lines 17
- SimplifyLine command 27
- SimplifyLineOrPolygon command 17
- Sin command 60
- SingleOutputMapAlgebra command 55
- SinH command 60
- Sink command 52
- Slice command 63, 72
- Slope command 63, 72
- SmoothLine command 27
- SnapPourPoint command 52
- SnapPour. *See* SnapPourPoint command
- Solve command 44
- Spatial Analyst toolbox 47
- Spatial grid index
 - calculating 24
- Spatial reference
 - creating 30
- Spatial Statistics toolbox 65
- SpatialAutocorrelation command 65
- Spline command 53, 70
- Split command 3, 13
- SplitLine command 26
- SQR. *See* Square command
- SQRT. *See* SquareRoot command
- Square command 56
- SquareRoot command 56
- Standard deviational ellipse 66
- StandardDistance command 66
- StandardizeAddresses command 37
- Standardizing
 - addresses 37
- Statistics
 - calculating 31
 - calculating for a block 61
 - calculating summary 5
 - command 5
 - frequency 5
 - of a cell 53
 - toolset 5
- StreamLink command 52
- StreamOrder command 52
- StreamShape. *See* StreamToFeature command
- StreamToFeature command 52
- Subtract. *See* Minus command
- Subtypes toolset 32
- Sum. *See* CellStatistics command
- Summary Statistics. *See* Statistics: command
- Summing
 - two raster datasets 56

Surface toolset	63
SurfaceLength command	70
SurfaceSpot command	70
SurfaceVolume command	70
SymDiff command	4
Symmetrical difference	4
Syntax example	2

T

Table toolset	33
Tables	
calculating frequency	5
converting or copying to tables	11
converting to dBASE	10
copying	27
creating	33
creating domains	24
creating pivot tables	33
deleting	27
importing to geodatabases	11
making table views	29
renaming	27
selecting	3
toolset	19
TableSelect command	3
TableToDBASE command	10
TableToDomain command	24
TableToGeodatabase command	11
TableToRelationshipClass command	32
TableToTable command	11
TabulateArea command	64
Tan command	60
TanH command	60
Temporary layers	
making. <i>See</i> Layers and Table Views toolset	
Test command	59
Thiessen command	14
Thin command	51
Tiger conversion	
advanced	15
basic	15
TigerArc command	15
TigerTool command	15
Times command	56, 71
TIN	
calculating aspect	73
calculating contours	73

TIN (continued)	
calculating slope	73
converting to raster datasets	69
creating	73
editing	73
extracting edges	69
extracting interpolation zones	69
extracting nodes	69
extracting polygon tag information	69
extracting triangle polygons	69
TIN Creation toolset	73
TIN Surface toolset	73
TINArc. <i>See</i> FeatureClassToCoverage command	
TINAspect command	73
TINContour command	73
TINDomain command	69
TINEdge command	69
TINEdit command	73
TINHull. <i>See</i> TINDomain command	
TINLattice. <i>See</i> TINRaster command	
TINLines. <i>See</i> TINEdge command	
TINNode command	69
TINPolygonTag command	69
TINRaster command	69
TINSlope command	73
TINSpot. <i>See</i> SurfaceSpot command	
TINTriangle command	69
To CAD toolset	9
To Coverage toolset	10, 15
To dBASE toolset	10
To Geodatabase toolset	10
To Raster toolset	11
To Shapefile toolset	12
Tolerance	
command	19
setting cluster tolerances	34
Tolerances toolset	19
Tool	
described	1
Toolbox	
3D Analyst	69
Analysis	3
Cartography	7
Conversion	9
Coverage	13
Data Interoperability	21
Data Management	21, 23
described	1

Toolbox (continued)		Toolset (continued)	
Geocoding	37	Multivariate	60
Geostatistical Analyst	39	Neighborhood	61
Linear Referencing	39, 41	Network Dataset	44
Network Analyst	43	Overlay	3, 13, 62
Spatial Analyst	47	Projections	18
Spatial Statistics	65	Projections and Transformations	29
Toolset		Proximity	4, 14
Aggregate	16	Raster	30
Analysis	13, 43	Raster Creation	62
Analyzing Patterns	65	Raster Interpolation	70
Bitwise	57	Raster Math	71
Composite Features	16	Raster Reclass	71
Conditional	47	Raster Surface	72
Conversion	14, 69	Reclass	62
Data Management	16	Relationship Classes	32
Database	23	Statistics	5
Density	47	Subtypes	32
described	1	Surface	63
Disconnected Editing	23	Table	33
Distance	48	Tables	19
Domains	23	TIN Creation	73
Extract	3, 13	TIN Surface	73
Extraction	49	To CAD	9
Feature	29	To Coverage	10, 15
Feature Class	24	To dBASE	10
Features	25	To Geodatabase	10
Fields	26	To Raster	11
From Coverage	14	To Shapefile	12
From Raster	9	Tolerances	19
Functional Surface	69	Topology	19, 33
General	27	Trigonometric	59
Generalization	17, 27, 50	Turn Feature Class	44
Groundwater	51	Utilities	67
Hydrology	51	Versions	34
Indexes	18, 28	Workspace	35
Interpolation	53	Workspace Management	19
items	18	Zonal	64
Joins	18, 28	Topology	
Layers and Table Views	28	adding rules	34
Local	53	creating	34
Logical	57	removing feature classes	34
Map Algebra	55	removing rules	34
Mapping Clusters	65	toolset	19, 33
Masking	7	validating	34
Math	55	VPF tiles	19
Measuring Geographic Distributions	66	TopoToRaster command	53, 70

TopoToRasterByFile command	53, 70
Transform command	18
Transforming	
raster datasets	30, 31
route events	42
TransformRouteEvents command	42
Trend command	53, 71
Trigonometric toolset	59
Turn Feature Class toolset	44
TurnTableToTurnFeatureClass command	45

U

Ungenerate command	15
Union command	4, 14
UnregisterAsVersioned command	35
Unregistering	
as versioned	35
Update command	4, 14
UpdateAnnotation command	25
UpdateByAlternateIDFields command	45
UpdateByGeometry command	45
Updating	
IDs	19
Utilities toolset	67

V

ValidateTopology command	34
Variety. <i>See</i> CellStatistics command	
Velocity. <i>See</i> DarcyVelocity command	
Versions	
altering	34
changing edit session	35
creating	34
deleting	35
reconciling	35
registering	35
toolset	34
unregistering	35
Viewshed command	63, 72
VisdeCode. <i>See</i> ObserverPoints command	
Visibility. <i>See</i> Viewshed command; LineOfSight command.	
Volume. <i>See</i> SurfaceVolume command	
VPF	
creating tile topology	19
VPFExport command	15

VPFImport command	16
VPFFile command	19

W

Warp command	30
Watershed command	52
WeightedOverlay command	62
Workspace	
creating	35
toolset	35
Workspace Management toolset	19

Z

Zonal toolset	64
ZonalArea. <i>See</i> TabulateArea command	
ZonalCentroid. <i>See</i> ZonalGeometry command	
ZonalFill command	64
ZonalGeometry command	64. <i>See also</i>
ZonalGeometryAsTable command	
ZonalGeometryAsTable command	64
ZonalMajority. <i>See</i> ZonalStatistics command	
ZonalMax. <i>See</i> ZonalStatistics command	
ZonalMean. <i>See</i> ZonalStatistics command	
ZonalMedian. <i>See</i> ZonalStatistics command	
ZonalMin. <i>See</i> ZonalStatistics command	
ZonalMinority. <i>See</i> ZonalStatistics command	
ZonalPerimeter. <i>See</i> ZonalGeometry command	
ZonalRange. <i>See</i> ZonalStatistics command	
ZonalStatistics command	64
ZonalStatisticsAsTable command	64
ZonalStats. <i>See</i> ZonalStatisticsAsTable command	
ZonalStd. <i>See</i> ZonalStatistics command	
ZonalSum. <i>See</i> ZonalStatistics command	
ZonalThickness. <i>See</i> ZonalGeometry command	
ZonalVariety. <i>See</i> ZonalStatistics command	
ZRenderer command	67

