



山东科技大学——测绘与空间信息学院

Python程序设计

地理信息科学系 刘洪强

J6-557 电话：86081170

2021年3月17日星期三

课程安排:

36个学时，其中授课24个学时，实验12个学时

章节内容

第1章 认识Python

第2章 Python编程基础

第3章 函数、类、包和模块

第4章 文件操作

第5章 地图文档管理与数据访问

第6章 空间数据定位与查询

第7章 空间数据分析

第8章 地图制图与输出

第5章 地图文档管理 与数据访问

主要内容

ArcGIS for Python 编程环境

ArcPy模块

管理地图文档和图层

加载数据

修复数据源

第5章 地图文档管理与数据访问

5.1 ArcGIS for Python 编程环境

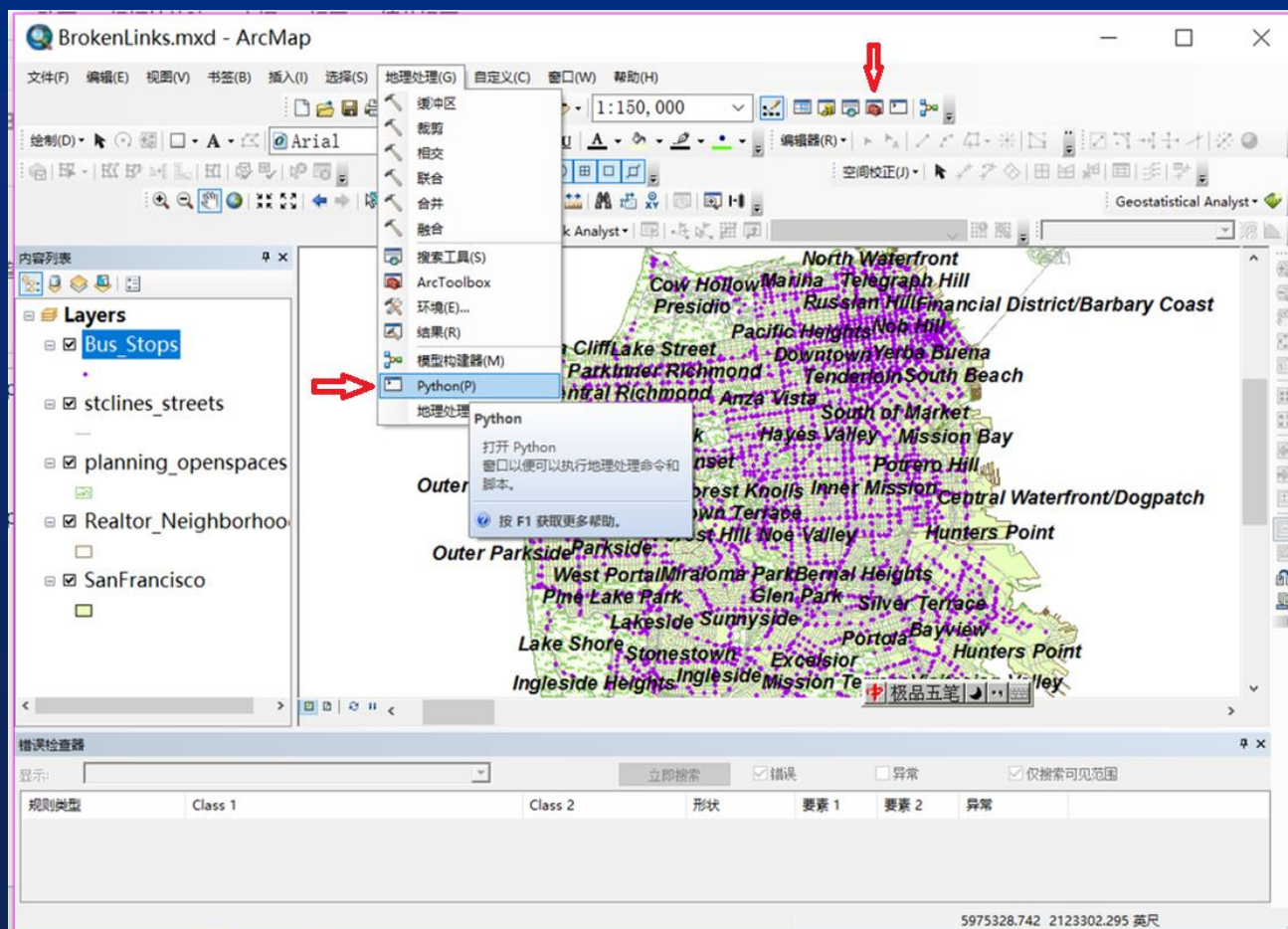
在GIS数据处理过程中，经常需要处理一系列步骤才可以完成的工作。ArcGIS软件提供了多种批处理、建模工具，并且可以结合编程语言完成更为复杂的步骤，这里将要介绍的Python脚本程序模型即为其中一种。

常用的编程环境包括在ArcMap中使用ArcPy；在IDLE中使用ArcPy。

第5章 地图文档管理与数据访问

1.1 ArcGIS for Python 编程环境

(1) ArcMap中集成的环境

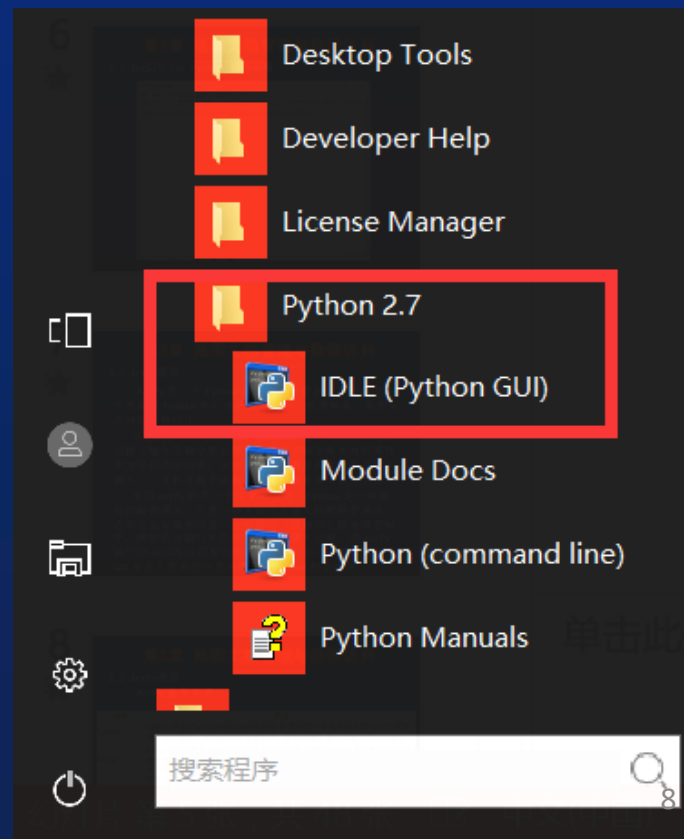


第5章 地图文档管理与数据访问

5.1 ArcGIS for Python 编程环境

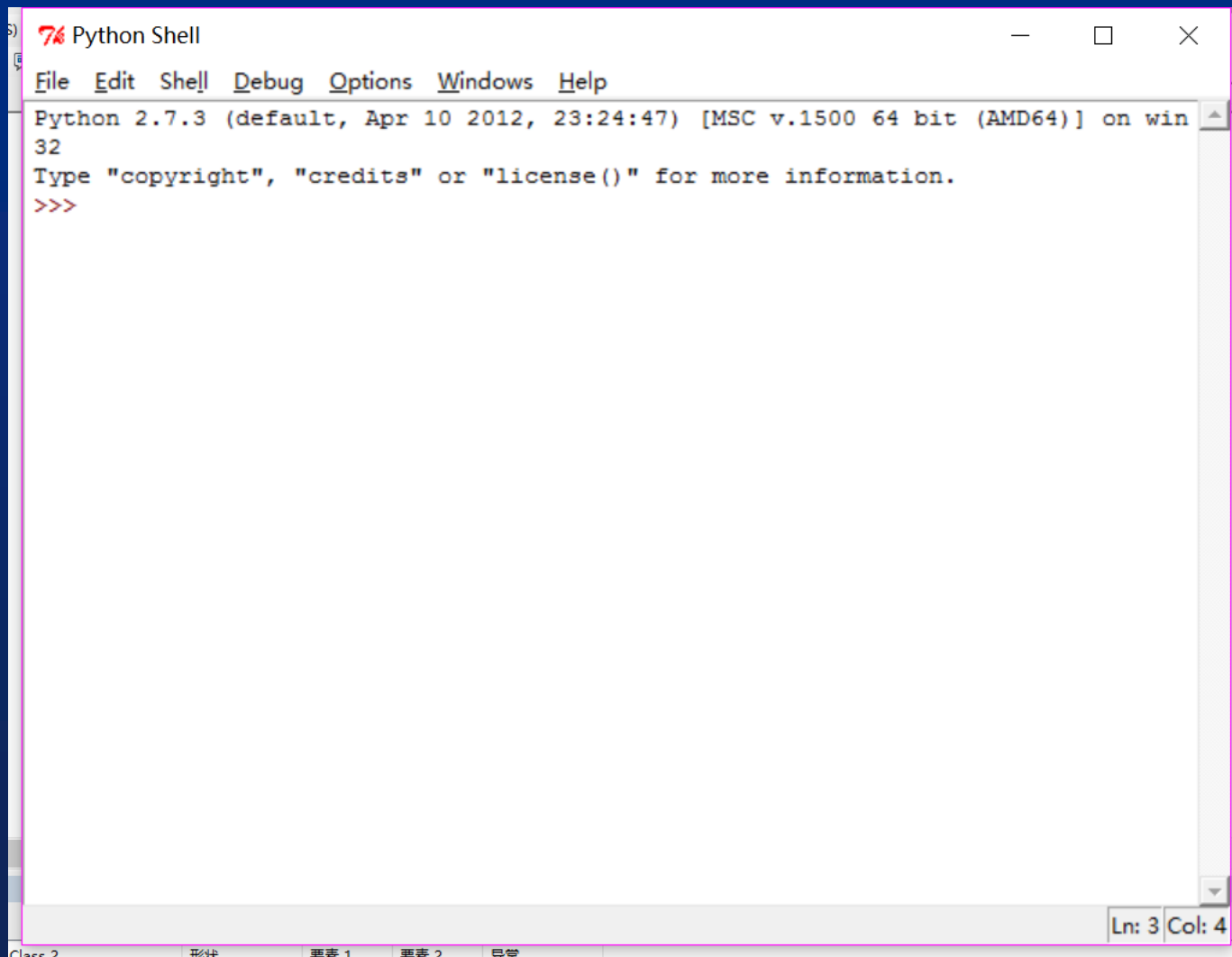
(2) IDLE环境

电脑开始菜单——>ArcGIS——
>Python2.7——>IDLE(Python GUI)
此时打开的是Python Shell窗口，它是用于显示输出结果和脚本的错误信息。在这个窗口中（File——>New file）打开的是Python脚本窗口，正是在这个窗口中进行脚本编写。



第5章 地图文档管理与数据访问

1.1 ArcGIS for Python 编程环境

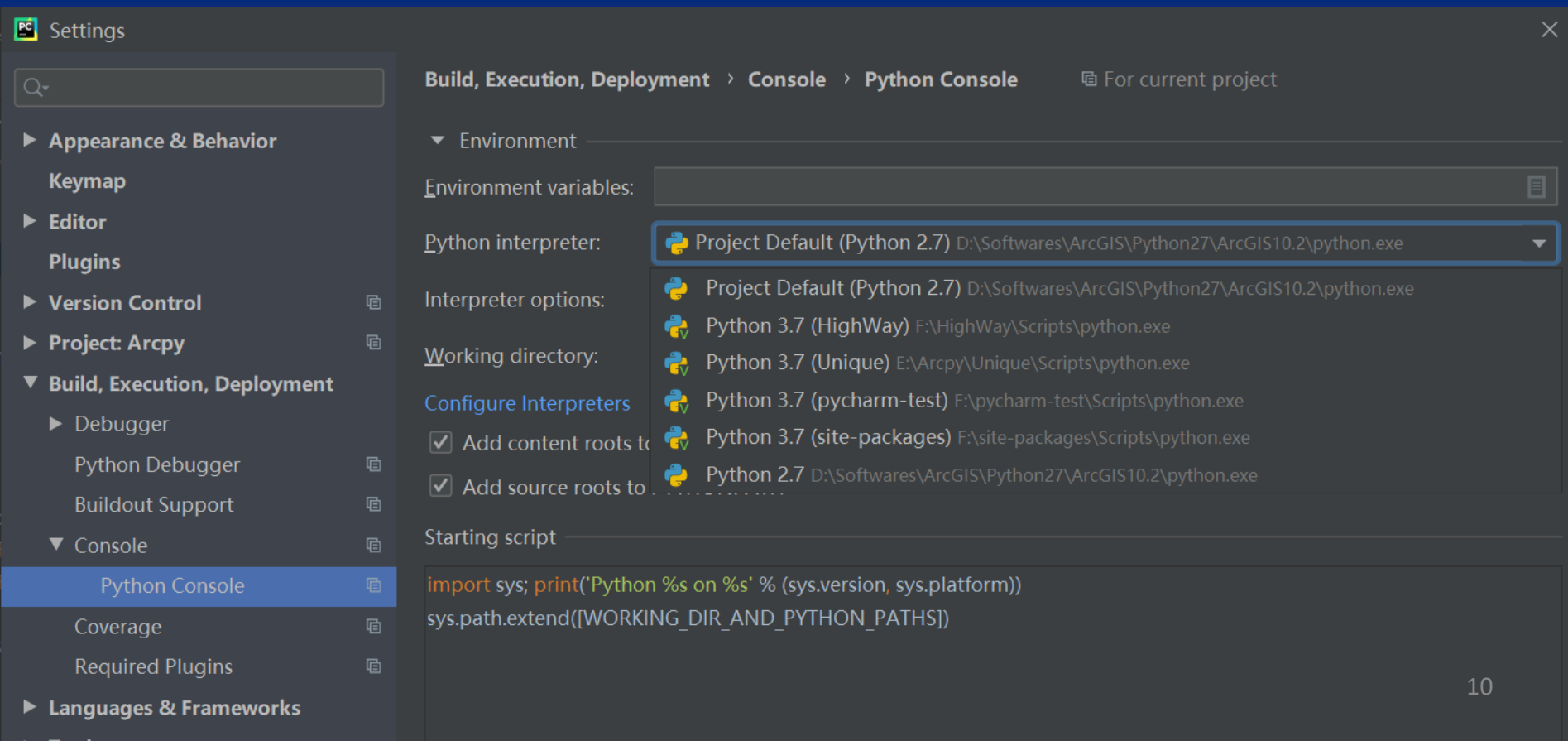


第5章 地图文档管理与数据访问

5.1 ArcGIS for Python 编程环境

(3) 其它编程环境

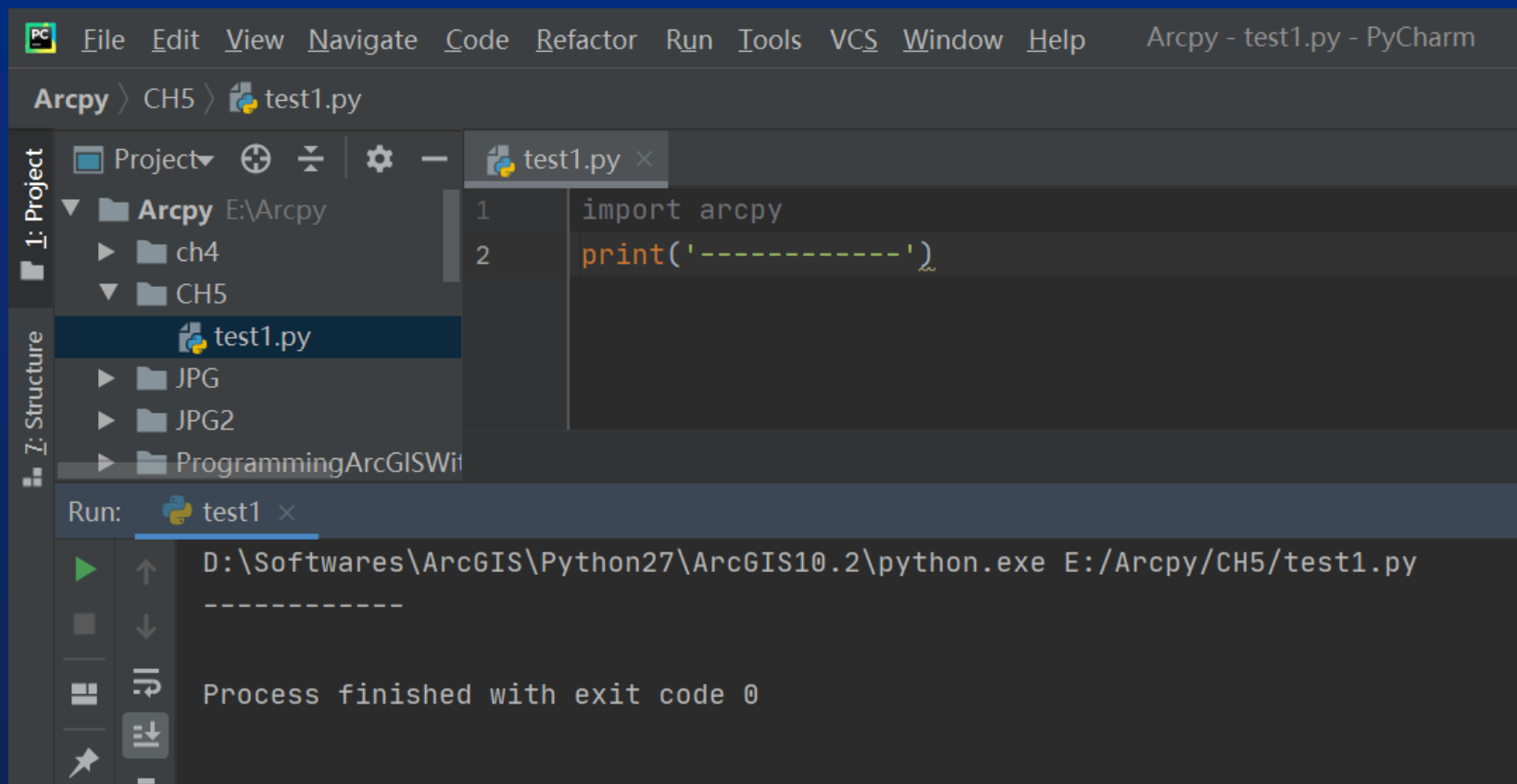
如PyCharm，要设置编译器为ArcGIS平台的 Python.exe



第5章 地图文档管理与数据访问

5.1 ArcGIS for Python 编程环境

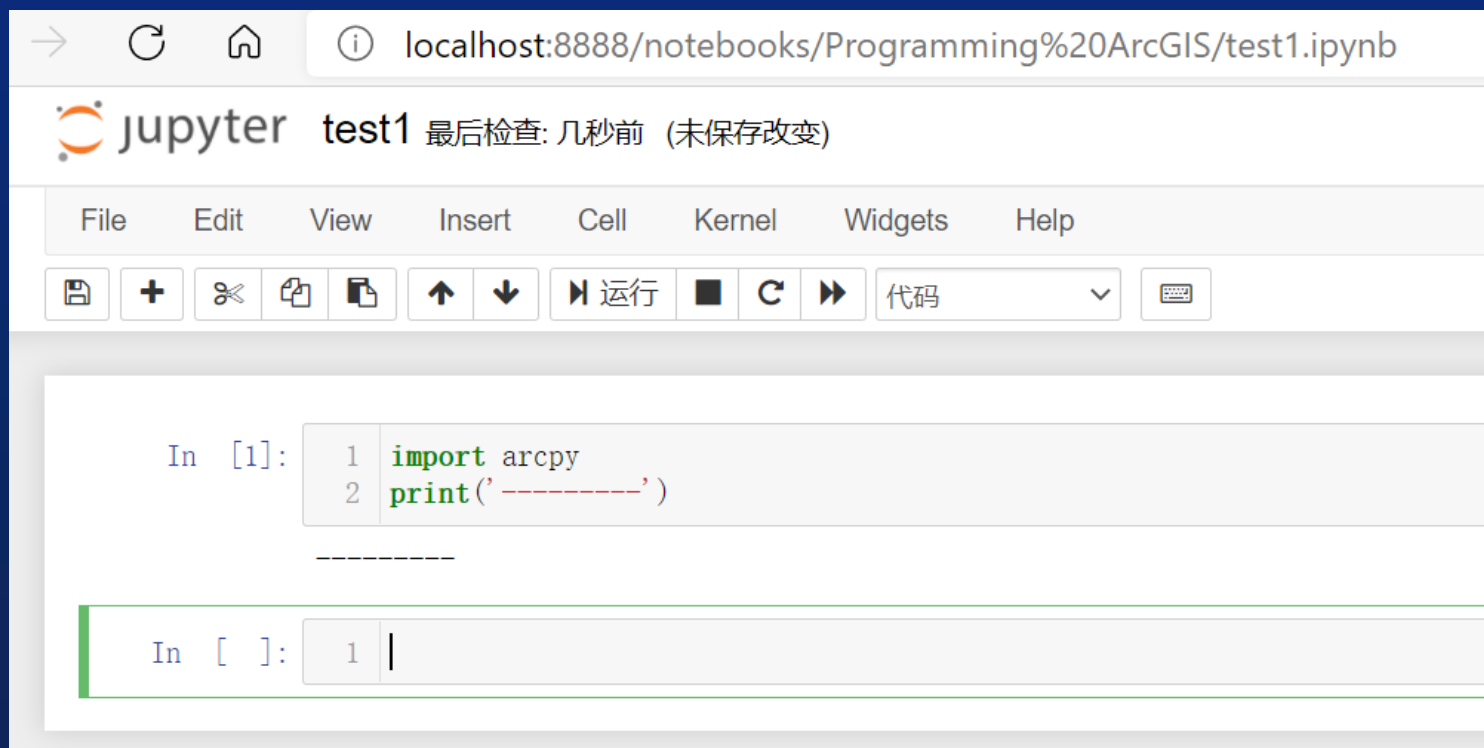
(3) 其它编程环境



第5章 地图文档管理与数据访问

5.1 ArcGIS for Python 编程环境

(3) 其它编程环境



第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy 是一个 Python 站点包，可提供以实用**高效**的方式通过 Python 执行**地理数据分析、数据转换、数据管理和地图自动化**。

该包提供了丰富纯正的 Python 体验，具有代码完成功能（输入关键字和点即可获得该关键字所支持的属性和方法的弹出列表；从中选择一个属性或方法即可将其插入），并针对每个函数、模块和类提供了参考文档。

使用 ArcPy 的另一个主要原因是，Python 是一种通用的编程语言。它是一种支持动态输入的解释型语言，适用于交互操作以及一次性程序（即脚本）快速原型制作，同时具有编写大型应用程序的强大功能。用 ArcPy 编写的 ArcGIS 应用程序的优势在于，不同领域的 GIS 专业人员和程序员可以共同开发 Python 附加模块。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy 基本术语

术语	定义
Python	Python 是由 Guido van Rossum 在1991 年推出的一种开源编程语言，简洁易读、语法清晰、支持动态定型等特点，并且具有大量标准库和第三方库。
ArcPy	ArcPy（通常称为 ArcPy 站点包）为用户提供了使用 Python 语言操作所有地理处理工具（包括扩展模块）的接口，并提供了多种有用的函数和类，以用于处理和查询 GIS 数据。
ArcPy 模块	模块为通常包含函数和类的 Python 文件。包括 数据访问模块 (arcpy.da)、 制图模块 (arcpy.mp)、 ArcGIS Spatial Analyst 空间分析模块 (arcpy.sa) 以及 ArcGIS Network Analyst 网络分析模块 (arcpy.na)。
ArcPy 类	ArcPy 类，如 SpatialReference 和 Extent 类，通常用作地理处理工具的参数，不然这些参数会使用更加复杂的字符串。
ArcPy 函数	函数是用于执行某项特定任务并能够纳入更大的程序的已定义功能。 在 ArcPy 中， 所有地理处理工具均以函数形式提供，但并非所有函数都是地理处理工具 。ArcPy 还提供多种函数来更好地支持 Python 地理处理工作流。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy 基本词汇

术语	定义
Python 脚本	独立 Python 脚本是一种 .py 文件，可通过两种方式执行：通过操作系统提示符，或 Python 集成开发环境 (IDE)。
Python 脚本工具	Python 脚本工具是已添加到地理处理工具箱的 Python 脚本。添加为脚本工具后，便可以像使用其他地理处理工具一样使用此脚本工具 - 可以从工具对话框打开和执行、可以在 Python 窗口和 ModelBuilder 中使用，也可以从其他脚本和脚本工具中调用。
Python 窗口	通过 Python 窗口，用户可以快捷地在 ArcGIS 内部使用 Python，从而以交互方式运行地理处理工具和功能以及充分利用其他 Python 模块和库。 Python 窗口可用于执行单行 Python 代码，并会将由此生成的消息输出到窗口。
Python 工具箱	Python 工具箱是完全使用 Python 语言创建的地理处理工具箱。Python 工具箱及其所包含工具的外观、操作和运行方式与任何以其他方式创建的工具箱和工具类似。 Python 工具箱 (.pyt) 是一个基于 ASCII 的文件，该文件定义了工具箱和一个或多个工具。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy 由一系列模块支持，常用的模块包括：

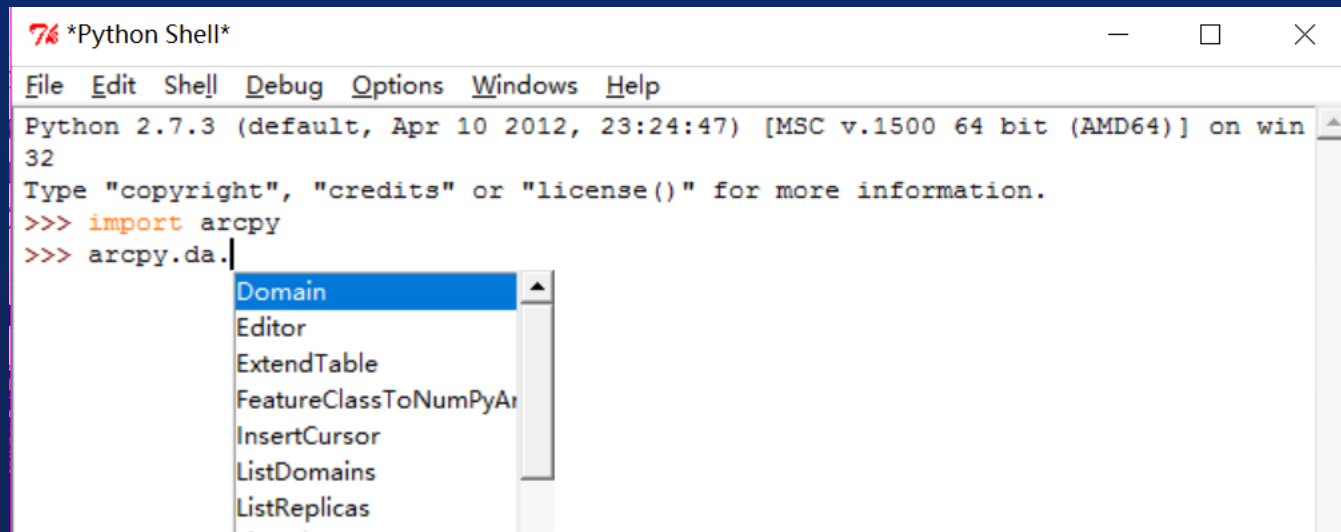
- (1) 数据访问模块 (**arcpy.da**)、
- (2) 制图模块 (**arcpy.mapping**)、
- (3) 空间分析模块(**arcpy.sa**)
- (4) 网络分析模块 (**arcpy.na**)

第5章 地图文档管理与数据访问

5.2 ArcPy模块

(1) ArcPy 数据访问模块

数据访问模块 (**arcpy.da**) 是一个用于处理数据的 Python 模块。通过它可控制编辑会话、编辑操作、改进的游标支持（包括更快的性能）、表和要素类与 NumPy 数组之间相互转换的函数以及对版本化、复本、属性域和子类型工作流的支持。



第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy 数据访问模块中的函数

函数	说明
<u>ExtendTable</u>	基于公共属性字段将 NumPy 结构化数组的内容连接到表。输入表将被更新，从而包含连接表中的字段。
<u>FeatureClassToNumPyArray</u>	将要素类转换为 NumPy 结构数组。
<u>ListDomains</u>	列出地理数据库的属性域。
<u>ListFieldConflictFilters</u>	列出应用字段冲突过滤器的版本化要素类或表中的字段。
<u>ListReplicas</u>	列出工作空间中的复本。
<u>ListSubtypes</u>	返回表或要素类的子类型字典。
<u>ListVersions</u>	列出工作空间中的数据版本。
<u>NumPyArrayToFeatureClass</u>	将 NumPy 结构化数组转换为点要素类。
<u>NumPyArrayToTable</u>	将 NumPy 结构化数组转换为表。
<u>TableToNumPyArray</u>	将表转换为 NumPy 结构化数组。
<u>Walk</u>	通过从上至下或从下至上遍历树，在目录/数据库结构中生成数据名称。各目录/工作空间提供一个三元组：目录路径、目录名称和文件名称。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy 数据访问模块中的类

类名	说明
Domain	属性域对象包括描述属性域的属性。
Editor	通过 Editor 类可以使用编辑会话和编辑操作来管理数据库事务。
InsertCursor	InsertCursor 可在要素类或表上建立写入游标。可以使用 InsertCursor 来添加新行。
Replica	副本对象包含一些用于描述副本的属性。
SearchCursor	SearchCursor 用于建立从要素类或表中返回的记录的单读访问权限。
UpdateCursor	UpdateCursor 用于建立对从要素类或表返回的记录的单写访问权限。
Version	版本对象包含一些用于描述版本的属性。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

(2) ArcPy 制图模块

Arcpy.mapping 是一个作为 ArcPy 站点包一部分的 Python 脚本模块。随 ArcGIS for Desktop 一同安装，并对所有许可均可用。其设计初衷主要是用于操作现有地图文档 (.mxd) 和图层文件 (.lyr) 的内容。此外，还提供自动执行导出和打印的功能。**Arcpy.mapping** 可用于自动执行地图生产；它扩展了数据驱动页面的功能，同时，因其包含导出至 PDF 文档、创建和管理 PDF 文档的函数，而为构建完整地图册所必需。最后，可将 **arcpy.mapping** 脚本发布为地理处理服务，并将脚本功能提供给 Web 应用程序。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

(2) ArcPy 制图模块

函数名	说明
AddLayer	添加图层。
AddLayerToGroup	添加图层组。
AddTableView	添加表视图。
AnalyzeForSD	SearchCursor 用于建立从要素类或表中返回的记录的单读访问权限。
ConvertWebMapToMapDocument	该函数会将完整状态的 web 地图插入过渡的模板地图文档。
CreateGISServerConnectionFile	此函数创建可连接到 ArcGIS for Server 或 Spatial Data Server 的连接文件。该连接文件随后可用于发布 GIS 服务。
CreateMapSDDraft	创建的输出是服务定义草稿 (.sddraft) 文件。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy 制图模块中的函数

函数名	说明
ExportToAI/ExportToBMP/ ExportToEMF/ExportToEPS/ ExportToGIF/ExportToJPEG/ ExportToPDF/ExportToPNG/ ExportToSVG/ExportToTIFF	输出地图到文件。
InsertLayer	将图层插入到数据框或地图文档 (.mxd) 中的图层组内的特定位置处。
Layer	用*.lyr文件创建图层对象。
ListBookmarks	通过返回的元组可访问各个空间书签名称和范围信息。
ListBrokenDataSources	返回已与其最初源数据断开连接的图层 (.lyr) 文件列表。
ListDataFrames	返回单个地图文档 (.mxd) 中存在的 DataFrame 对象列表。
ListLayers	返回单个地图文档 (.mxd) 中存在的图层对象列表。
ListLayoutElements	返回地图文档 (.mxd) 布局中布局元素的 Python 列表。
ListMapServices	列出指定 ArcGIS for Server 的地图服务名称。
ListPrinterNames	返回在本地计算机中的可用打印机的列表。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy制图模块中的函数

函数名	说明
ListStyleItems	返回 StyleItem 对象的 Python 列表。然后可使用样式文件（.style 或 .ServerStyle）中的已引用图例项更新布局中已存在的图例项。
ListTableViews	返回地图文档 (.mxd) 中的 TableView 对象的列表。
MapDocument	加载*.MXD文件，创建MapDocument对象。
MoveLayer	将图层移动到指定的位置。
PDFDocumentCreate	在内存中创建空的 PDFDocument 对象。
PDFDocumentOpen	打开的 PDF 文件，创建PDFDocument对象。
PrintMap	将特定数据框或地图文档 (.mxd) 布局输出到打印机或文件。
RemoveLayer	移除地图文档 (.mxd) 中数据框内的图层。
RemoveTableView	将从地图文档的特定数据框中移除单个表。
TableView	将工作空间中的表作为 TableView 对象引用。
UpdateLayer	通过从源图层提取信息，更新地图文档 (.mxd) 中所有图层属性或仅更新图层的符号系统。
UpdateLayerTime	通过从源图层提取时间属性，更新地图文档 (.mxd) 中图层的时间属性。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy制图模块中的类

类名	说明
DataDrivenPages	用于提供对已启用“数据驱动页面”的地图文档内管理各个页面的方法和属性的访问。
DataFrameTime	用于访问数据框中启用了时间属性的图层管理操作。
DataFrame	用于访问多个数据框属性的 DataFrame 对象。
GraduatedColorsSymbology	访问用于更改图层分级色彩符号系统外观的各种属性。
GraduatedSymbolsSymbology	访问用于更改图层分级符号系统外观的各种属性。
GraphicElement	提供了对属性和方法的访问，属性用于在页面布局上重新定位图形元素，而方法用于复制和删除现有图形元素。
LabelClass	用于访问图层的标注类属性。
Layer	用于对图层的属性和方法进行访问。
LayerTime	用于访问启用了时间属性的图层管理操作。
LegendElement	用于对属性和方法进行访问。
MapDocument	用于访问地图文档 (.mxd) 属性和方法。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

ArcPy制图模块中的类

类名	说明
MapsurroundElement	访问能够在页面布局中重新定位以及识别父数据框的属性。
PDFDocument	可对 PDF 文档进行操作，包括合并页面、删除页面、设置文档打开方式、添加文件附件以及创建或更改文档安全性设置。
PictureElement	用于访问图片属性，从而在页面布局中重新定位图片并获取和设置其数据源。
RasterClassifiedSymbology	访问用于更改图层栅格分类符号系统外观的各种属性。
StyleItem	对 StyleItem 类属性进行访问。
TableView	用于访问基本表属性（如数据源信息等）以及设置表的定义查询。
TextElement	用于对属性进行访问，这些属性可以在页面布局中重新定位文本以及修改文本字符串和字号。
UniqueValuesSymbology	用于更改图层唯一值符号系统外观的各种属性。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

(3) Spatial Analyst 模块

Spatial Analyst 模块(**arcpy.sa**)是用于分析栅格数据的 Python 模块，该模块在进行分析时将使用 ArcGIS Spatial Analyst 扩展模块提供的功能。借助该模块可访问 Spatial Analyst 工具箱中提供的所有地理处理工具以及其他帮助程序函数和类，从而您可以更加轻松地通过 Python 实现工作流的自动化。

该模块具有三个主要组件：**类**、**运算符**和**函数**。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

模糊类

[FuzzyGaussian](#)

通过**高斯分布或基于用户指定中点的正态分布**，定义针对特定值的**模糊隶属度函数**。

[FuzzyLarge](#)

通过一个用户定义的中点和定义的散度来定义函数。

[FuzzyLinear](#)

通过用户指定的最小值（隶属度为**0**）到用户定义的最大值（分配的隶属度为**1**）之间的线性变换来定义模糊隶属度函数。

[FuzzyMSLarge](#)

通过一个基于平均值和标准差的函数，以隶属度更接近于**1**的较大值定义模糊隶属度。

[FuzzyMSSmall](#)

通过一个基于平均值和标准差的函数，以隶属度更接近于**1**的较小值定义模糊隶属度。

[FuzzyNear](#)

通过一个用户定义的中点（分配的隶属度为**1**）及经过定义减至零的散度，定义针对特定值的模糊隶属度函数。

[FuzzySmall](#)

通过一个用户定义的中点和定义的散度来定义函数。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

水平系数类定义了一个从一个像元移动到另一个像元因水平距离等因素造成的困难程度，同时解释可能影响此移动的水平系数。

水平 系数 类	HfBinary	通过二进制函数定义水平成本系数和水平相对移动角度之间的关系。如果水平相对移动角度小于交角，则将水平系数设置为与零系数相关联的值；否则为无穷大。
	HfForward	通过向前函数定义水平成本系数和水平相对移动角度之间的关系。此函数只允许建立向前移动。
	HfLinear	通过线性函数定义水平成本系数和水平相对移动角度之间的关系。该函数规定水平系数与水平相对移动角度具有线性关系。
	HfInverseLinear	通过逆线性函数定义水平成本系数和水平相对移动角度之间的关系。该函数规定水平系数为水平相对移动角度的逆线性函数。
	HfTable	通过指定表文件定义水平成本系数和水平相对移动角度之间的关系。表文件用于标识确定水平系数的水平系数图。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

克里金类定义普通克里金法模型，进行插值。可用模型类型包括“球面”、“圆”、“指数”、“高斯”和“线性”。

克里金类	<u>KrigingModelOrdinary</u>	定义普通克里金法模型。可用模型类型包括“球面”、“圆”、“指数”、“高斯”和“线性”。
	<u>KrigingModelUniversal</u>	定义泛克里金法模型。可用模型类型包括“与一次漂移函数成线性关系”和“与二次漂移函数成线性关系”。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

邻域类是用于定义不同类型的邻域类。

邻域搜索半径工具将基于距离点或线状要素指定距离内的要素执行各种计算。它们分别是点统计和线统计工具。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

邻域类

<u>NbrAnnulus</u>	定义通过指定内、外圆半径（以地图单位或像元数为单位）创建的环形邻域。
<u>NbrCircle</u>	定义通过指定半径（以地图单位或像元数为单位）而创建的圆形邻域。
<u>NbrIrregular</u>	定义由核文件创建的不规则邻域。
<u>NbrRectangle</u>	定义通过指定以地图单位或像元数为单位的高度和宽度而创建的矩形邻域。
<u>NbrWedge</u>	定义通过指定半径和两个角度（以地图单位或像元数为单位）而创建的楔形邻域。
<u>NbrWeight</u>	定义使用核文件创建的权重邻域，该核文件指定用于乘以邻域范围内像元的值。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

加权叠加类定义将用于将各种输入栅格转换为通用比例值的重分类以及要用于进行集成分析的权重。根据重要性对输入栅格重分类或加权，随后一起添加以生成输出栅格。

加权叠加使用 **WOTable**。

加权总和使用 **WSTable**。

叠 加 类	WOTable	定义输入栅格、标识输入值的字段、栅格值的重映射、每个栅格的权重以及评估等级供加权叠加工具使用。
	WSTable	定义将要在 WeightedSum 工具中相加的输入栅格及其权重。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

半径类用于对表面进行插值时。半径对象定义搜索半径的特征以计算合成表面。

半径类	<u>RadiusFixed</u>	通过指定分析所需的距离和最少点数定义固定搜索半径。如果在指定距离内没有找到所需数量的点，则增加搜索半径，直至找到指定最少数量的点。
	<u>RadiusVariable</u>	通过指定分析的最大距离和点数定义可变搜索半径。如果在最大距离内无法满足点数，将使用较小点数。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

重映射类用于定义如何对数据进行重分类。重映射对象为列表中的列表，这些列表用于为新输出像元值分配输入像元值。

重 映 射 类	<u>RemapRange</u>	将输入值重分类为输出栅格所对应的区间列表。
	<u>RemapValue</u>	该列表用于将各输入值重分类为各输出栅格所对应的值。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

变换函数类用于定义要应用于输入栅格值的特定变换函数。共有十三个变换函数类，其中每个变换函数类都定义一个特定的函数。

变换类

<u>TfExponential</u>	定义一个指数变换函数，该函数是根据平移和基础系数这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfGaussian</u>	定义一个高斯变换函数，该函数是根据中点和散度这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfLarge</u>	定义一个大值变换函数，该函数是根据中点和散度这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfLinear</u>	定义一个线性变换函数，该函数是根据最大值和最小值这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfLogarithm</u>	定义一个对数变换函数，该函数是根据平移和系数这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfLogisticDecay</u>	定义一个逻辑衰减变换函数，该函数是根据最大值、最小值和 y 截距百分比这三个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

变换类

<u>TfLogisticGrowth</u>	定义一个逻辑增长变换函数，该函数是根据最大值、最小值和 y 截距百分比这三个形状控制参数，确定函数应用范围的阈值上限和下限确定的。
<u>TfMSLarge</u>	定义一个 MS 大值变换函数，该函数是根据平均值乘数和标准差乘数这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfMSSmall</u>	定义一个 MS 小值变换函数，该函数是根据平均值乘数和标准差乘数这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfNear</u>	定义一个邻近分析变换函数，该函数是根据中点和散度这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfPower</u>	定义一个幂变换函数，该函数是根据平移和指数这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfSmall</u>	定义一个小值变换函数，该函数是根据中点和散度这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。
<u>TfSymmetricLinear</u>	定义一个对称线性变换函数，该函数是根据最大值和最小值这两个形状控制参数，以及确定函数应用范围的阈值上限和下限确定的。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

时间类用于指定时间配置以计算太阳辐射量。共有四种时间类可以指定某天的某段时间、一年中的多天、特定的日子（春分、夏至、秋分和冬至）以及整年。

时间类	<u>TimeMultipleDays</u>	定义特定年份内的多日时间段以进行太阳能量计算。指定年份、起始日和结束日。
	<u>TimeSpecialDays</u>	将夏至/春秋分/冬至日定义为时间段以进行太阳能量计算。
	<u>TimeWholeYear</u>	使用每月间隔定义年份以进行太阳能计算。
	<u>TimeWithinDay</u>	定义一天内的时间段以进行太阳能计算。指定儒略日、起始时间和结束时间。

第5章 地图文档管理与数据访问

5.2 ArcPy模块 Spatial Analyst 模块中的类

地形 (topo) 类用于定义在创建插值栅格表面时要使用的输入数据。

地形类	TopoBoundary	定义要素类列表，这些要素类包含表示输出栅格外缘的面。
	TopoCliff	定义悬崖位置的线要素类列表。
	TopoCoast	定义表示海岸边界的面要素类列表。
	TopoContour	定义线要素类列表，这些线要素类表示等值线以及标识其高程值的字段。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

地形 (topo) 类用于定义在创建插值栅格表面时要使用的输入数据。

地形类

<u>TopoExclusion</u>	定义排除特定数据区域的面要素类列表。
<u>TopoLake</u>	定义面要素类列表，这些面要素类指定湖泊位置。
<u>TopoPointElevation</u>	定义点要素类列表，这些点要素类表示表面高程和标识其高程值的字段。
<u>TopoSink</u>	定义点要素类列表，这些点要素类表示已知地形凹陷的位置和量级以及标识其高程值的字段。
<u>TopoStream</u>	定义表示河流位置的线要素类列表。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

垂直系数 (VF) 类定义从一个像元移至另一个像元的难度，同时解释可能影响此移动的垂直系数。路径距离工具使用垂直系数对象确定垂直系数。

垂直系数类

[VfBinary](#)

通过二元函数定义垂直成本系数和垂直相对移动角度之间的关系。如果垂直相对移动角度大于交角下限且小于交角上限，则将垂直系数设置为与零系数相关联的值；否则为无穷大。

[VfCos](#)

通过余弦函数定义垂直成本系数和垂直相对移动角度之间的关系。

[VfCosSec](#)

通过余弦/正割函数定义垂直成本系数和垂直相对移动角度 (VRMA) 之间的关系。如果 VRMA 为负数，则由余弦函数确定垂直系数，如果 VRMA 为非负数，则由正割函数确定垂直系数。

[VfInveseLinear](#)

通过逆线性函数定义垂直成本系数和垂直相对移动角度之间的关系。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的类

垂直系数 (VF) 类定义从一个像元移至另一个像元的难度，同时解释可能影响此移动的垂直系数。路径距离工具使用垂直系数对象确定垂直系数。

垂直系数类

VfSec	通过正割函数定义垂直成本系数和垂直相对移动角度之间的关系。
VfSecCos	通过正割/余弦函数定义垂直成本系数和垂直相对移动角度 (VRMA) 之间的关系。如果 VRMA 为负数，则由正割函数确定垂直系数，如果 VRMA 为非负数，则由余弦函数确定垂直系数。
VfSymInverseLinear	通过垂直相对移动角度 (VRMA) 负侧或正侧的对称逆线性函数确定垂直成本系数和 VRMA 之间的关系。这两个线性函数关于 VF (y) 轴对称。
VfSymLinear	通过垂直相对移动角度 (VRMA) 负侧或正侧的对称线性函数定义垂直成本系数和 VRMA 之间的关系。这两个线性函数关于 VF (y) 轴对称。
VfTable	使用垂直系数图定义垂直成本系数和垂直相对移动角度之间的关系，此垂直系数图确定表文件指定的垂直系数。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的运算符

地图代数运算符	描述	Spatial Analyst GP 工具	ArcGIS 9.x 地图代数等效运算符
算 术			
+	加	加	+
+	一元加号	N/A	N/A
-	减	减	-
-	一元减号	取反	-
*	乘法	乘	*
**	幂	幂	N/A
/	除	除	/, div
//	整除	N/A	N/A
%	模	求模	求模

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的运算符

地图代数运算符	描述	Spatial Analyst GP 工具	ArcGIS 9.x 地图代数等效运算符
按 位			
<<	按位左移	按位左移	<<
>>	按位右移	按位右移	>>
布尔			
&	布尔与	布尔与	&, and
~	布尔求反	布尔非	^, not
	布尔或	布尔或	, or
^	布尔异或	布尔异或	!, xor

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Spatial Analyst 模块中的运算符

地图代数运算符	描述	Spatial Analyst GP 工具	ArcGIS 9.x 地图代数等效运算符
关 系			
==	等于	等于	==, eq
>	大于	大于	>, gt
>=	大于或等于	大于等于	>=, ge
<	小于	小于	<, lt
<=	小于或等于	小于等于	<=, le
!=	不等于	不等于	^=, <>, ne

第5章 地图文档管理与数据访问

5.2 ArcPy模块

(4) Network Analyst 模块

Network Analyst 模块 (**arcpy.na**) 是用于使用 ArcGIS Network Analyst 扩展模块提供的网络分析功能的 Python 模块。通过它可访问 Network Analyst 工具箱中提供的所有地理处理工具以及允许您通过 Python 使 Network Analyst 工作流自动化的其他帮助程序函数和类。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Network Analyst 模块中的类

类名	说明
<u>ClosestFacilitySolverProperties</u>	用于访问最近设施点网络分析图层中的分析属性。 GetSolverProperties 函数用于从最近设施点网络分析图层中获取 ClosestFacilitySolverProperties 对象。
<u>LocationAllocationSolverProperties</u>	用于访问位置分配网络分析图层中的分析属性。 GetSolverProperties 函数用于从位置分配网络分析图层中获取 LocationAllocationSolverProperties 对象。
<u>NAClassFieldMap</u>	用于映射网络分析图层内网络分析类的属性字段名称，或设置该属性的默认值。执行网络分析时，求解程序会将网络分析类的属性用作输入。
<u>NAClassFieldMappings</u>	提供 NAClassFieldMap 对象的 Python 字典，用于映射网络分析图层内网络分析类的属性字段名称，或设置该属性的默认值。字典关键字是网络分析类属性名称，值为 NAClassFieldMap 对象。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Network Analyst 模块中的函数

类名	说明
<u>ODCostMatrixSolverProperties</u>	用于访问起始 - 目的地 (OD) 成本矩阵网络分析图层中的分析属性。
<u>RouteSolverProperties</u>	用于访问路径网络分析图层的分析属性。 GetSolverProperties 函数用于从路径网络分析图层中获取 RouteSolverProperties 对象。
<u>ServiceAreaSolverProperties</u>	用于访问服务区网络分析图层中的分析属性。 GetSolverProperties 函数用于从服务区网络分析图层中获取 ServiceAreaSolverProperties 对象。
<u>StreetDirectionProperties</u>	为街道方向属性提供读写访问，使您可自定义网络分析图层的方向输出。
<u>VehicleRoutingProblemSolverProperties</u>	用于访问车辆配送 (VRP) 网络分析图层中的分析属性。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Network Analyst 模块中的函数

函数名	说明
<u>AddFieldToAnalysisLayer</u>	向网络分析图层的子图层添加字段。
<u>AddLocations</u>	向网络分析图层添加网络分析对象。
<u>BuildNetwork</u>	重新构建网络数据集的网络连通性和属性信息。
<u>CalculateLocations</u>	向包含要素网络位置的输入要素添加字段。
<u>CheckIntersectingFeatures</u>	用来指示指定的网络数据集内与指定的要素图层中的要素相交的边源要素的数量是否小于或等于指定中断值。
<u>CopyTraversedSourceFeatures</u>	创建两个要素类和一个表，它们组合在一起以包含求解网络分析图层时所遍历的边、交汇点和转弯的信息。
<u>CreateTurnFeatureClass</u>	创建新的转弯要素类，以将对转弯移动进行建模的转弯要素存储在网络数据集中。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Network Analyst 模块中的函数

函数名	说明
Directions	根据包含路径的网络分析图层生成转弯方向。可以将这些方向信息写入到文本、XML 或 HTML 格式的文件中。
DissolveNetwork	创建可最大程度减少正确构建输入网络数据集模型所需线要素数目的网络数据集。
FindClosestFacilities	查找一个或多个离事件点最近的设施点，并输出事件点与设施点间的最佳路径、所选设施点和行驶方向。
FindRoutes	查找路径分析意味着确定来往于各个位置间的最快或最短路径。
GenerateDirectionsFeatures	在通过求解网络分析图层而生成的地图及相应的文本方向上创建路径线的要素类。
GenerateServiceAreas	创建服务区网络分析图层，设置分析属性及求解分析。此工具非常适用于在 web 上设置服务区地理处理服务。
GetNAClassNames	返回指定为参数的网络分析图层中的网络分析类名称字典。 ⁴⁹

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Network Analyst 模块中的函数

函数名	说明
<u>GetSolverProperties</u>	根据指定为参数的网络分析图层类型返回 Network Analyst 求解程序属性对象。
<u>GetTravelModes</u>	返回适用于网络数据集的出行模式对象的字典。字典关键字是出行模式的名称，字典值是出行模式对象。
<u>IncreaseMaximumEdges</u>	增加转弯要素类中每个转弯所允许的最大边数。
<u>ListDirectionsLanguages</u>	返回语言列表，您可以在其中为网络分析期间所创建的路径生成方向。
<u>ListDirectionsStyleNames</u>	返回一个可用方向样式列表，通过这些样式能够根据网络分析过程中创建的路径生成方向。
<u>MakeClosestFacilityLayer</u>	创建最近设施点网络分析图层并设置其分析属性。
<u>MakeLocationAllocationLayer</u>	创建位置分配网络分析图层并设置其分析属性。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Network Analyst 模块中的函数

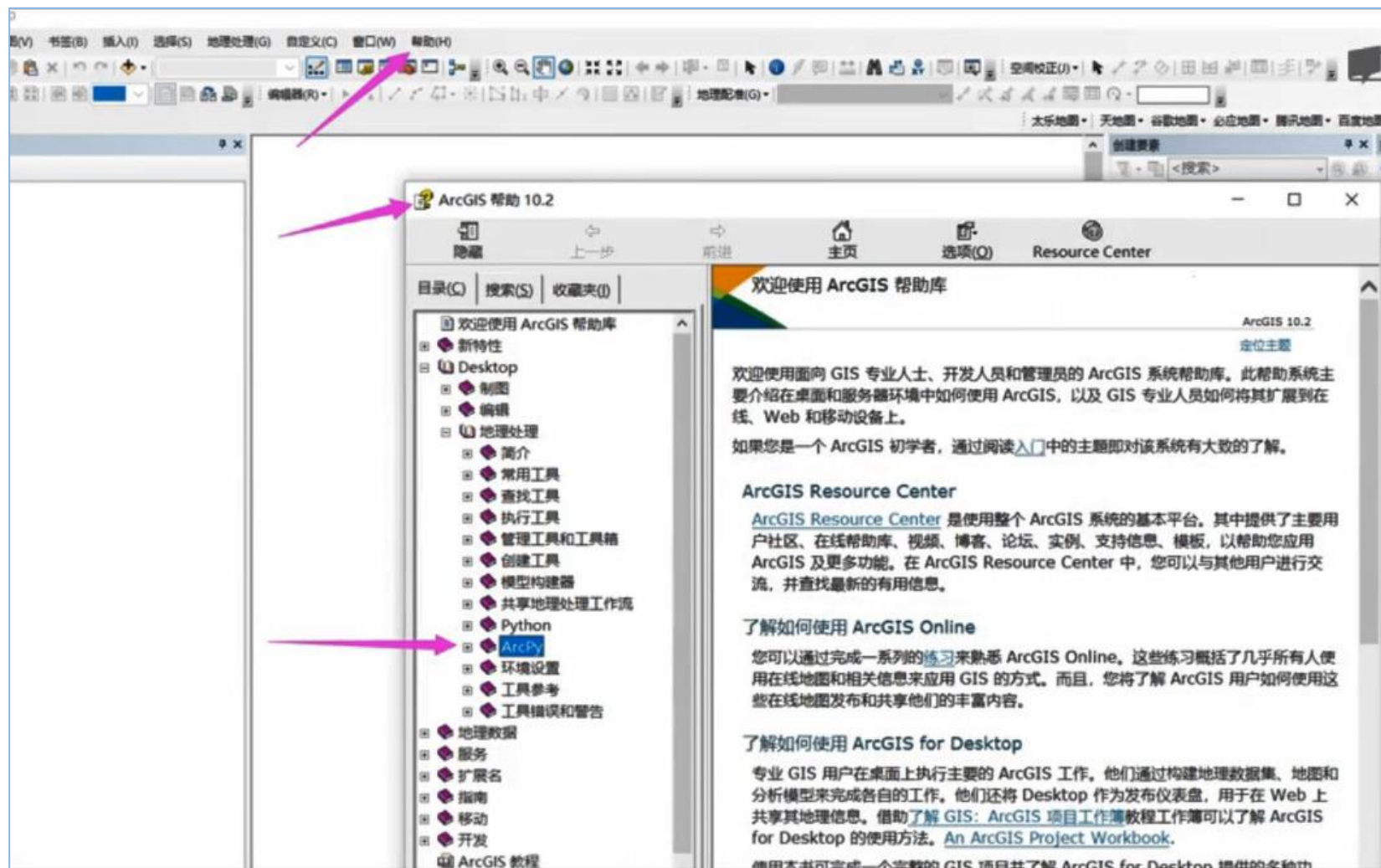
函数名	说明
<u>MakeODCostMatrixLayer</u>	创建起始-目的地 (OD) 成本矩阵网络分析图层并设置其分析属性。
<u>MakeRouteLayer</u>	创建路径网络分析图层并设置其分析属性。
<u>MakeServiceAreaLayer</u>	创建服务区网络分析图层并设置其分析属性。
<u>MakeVehicleRoutingProblemLayer</u>	创建车辆配送 (VRP) 网络分析图层并设置其分析属性。
<u>PopulateAlternateIDFields</u>	为通过备用 ID 来引用边的转弯要素类创建并填充附加字段。
<u>Solve</u>	基于网络位置和属性求解网络分析图层问题。
<u>SolveLocationAllocation</u>	“求解位置分配”工具用于从一组输入位置中选择最佳位置。

第5章 地图文档管理与数据访问

5.2 ArcPy模块

Network Analyst 模块中的函数

函数名	说明
<u>SolveVehicleRoutingProblem</u>	创建车辆配送 (VRP) 网络图层，设置分析属性，求解分析。
<u>TurnTableToTurnFeatureClass</u>	将 ArcView 转弯表或 ArcInfo Workstation coverage 转弯表转换为 ArcGIS 转弯要素类。
<u>UpdateAnalysisLayerAttributeParameter</u>	更新网络分析图层的网络属性参数值。
<u>UpdateByAlternateIDFields</u>	使用备用 ID 字段更新转弯要素类中的所有边引用。
<u>UpdateByGeometry</u>	使用转弯要素的几何更新转弯要素类中的所有边引用。
<u>UpdateTrafficData</u>	下载实时交通数据并用来进行实时交通分析和显示。
<u>UpdateTrafficIncidents</u>	通过 Web 服务创建包含实时交通事件数据的点要素类。
<u>UpgradeNetwork</u>	更新网络数据集。



第5章 地图文档管理与数据访问

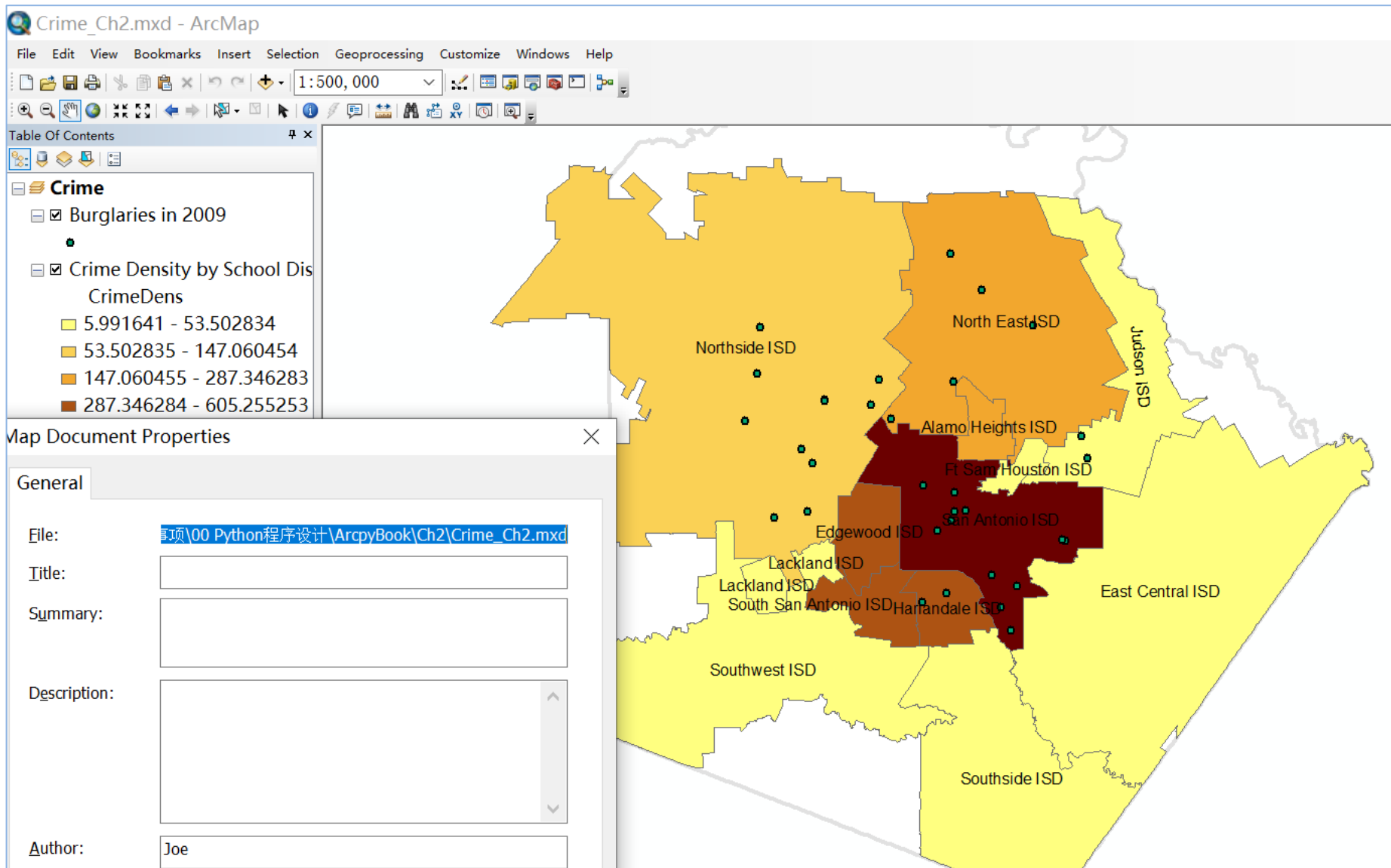
5.3 管理地图文档和图层

(1) 引用当前的地图文档

在对地图文档执行任何操作之前，都需要先在Python脚本中引用地图文档，可以通过调用**arcpy.mapping**模块中的**MapDocument()**函数来实现。引用地图文档的途径有两种：一是引用ArcMap中**当前活动的文档**；二是引用**磁盘中特定位置的文档**。使用CURRENT关键字作为MapDocument()函数的参数，就可以加载ArcMap中当前活动的地图文档。

例：ReferenceCurrentMapDocument.py

```
import arcpy.mapping as mapping
mxd = mapping.MapDocument("CURRENT")
mxd.title = "Crime Project"
mxd.saveACopy("c:/ArcpyBook/Ch2/crime_copy.mxd")
```




```
Python
>>> import arcpy.mapping as mapping
>>> mxd = mapping.MapDocument('CURRENT')
>>> mxd.title = 'Crime project'
>>> mxd.saveACopy(r'C:\ArcpyBook\Ch2\crime_copy.mxd')
>>>
```

crime_copy.mxd - ArcMap

File Edit View Bookmarks Insert Selection Geoprocessing Customize Windows Help

1:500,000

Table Of Contents

- Crime
 - Burglaries in 2009
 - Crime Density by School District
 - 5.991641 - 53.502834
 - 53.502835 - 147.060454
 - 147.060455 - 287.346284
 - 287.346284 - 605.255254
 - 605.255254 - 772.31814
 - Bexar County Boundary
 - Crime2009Table
- Test_Performance
- Inset_Map
- Crime_Inset

Map Document Properties

General

File: C:\ArcpyBook\Ch2\crime_copy.mxd

Title: Crime project

Summary:

Description:

Author: Joe

Credits:

Tags:

Hyperlink base:

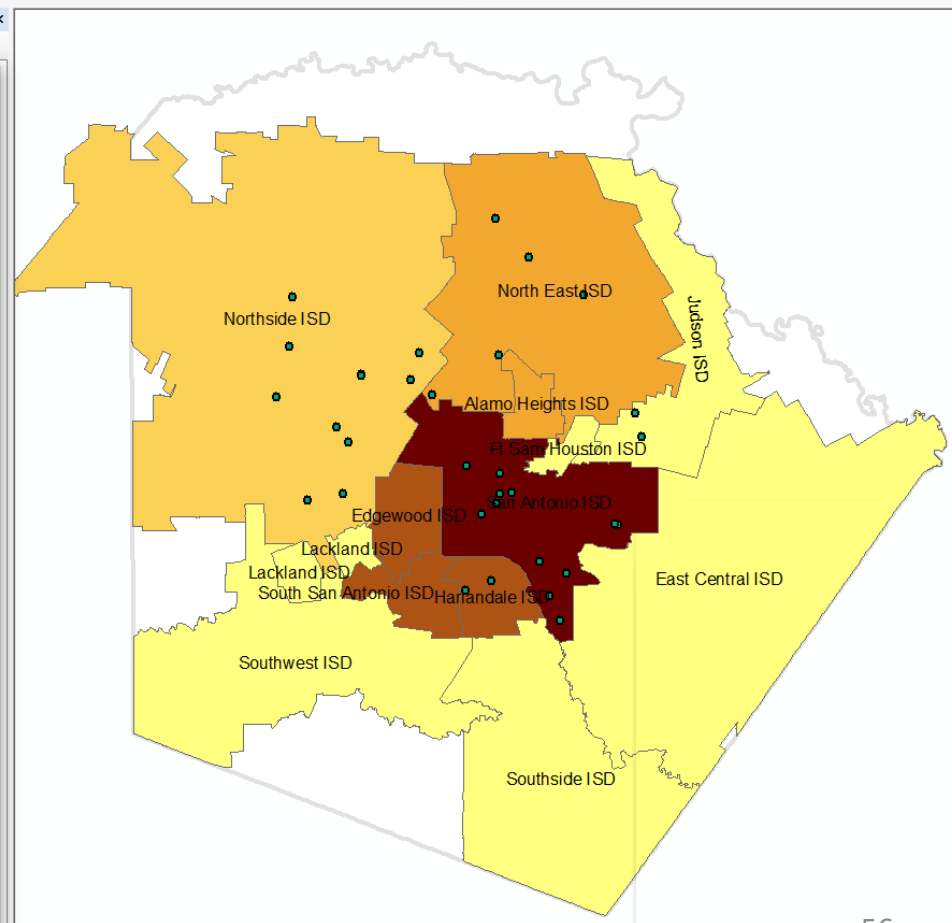
Last Saved: 2021/3/16 9:11:40

Last Printed:

Last Exported: 2016/3/15 17:26:57

Default Geodatabase: C:\Users\Unique\Documents\ArcGIS\Default.gdb

Pathnames: ☐ Store relative pathnames to data sources



第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(1) 引用当前的地图文档

只有在ArcGIS的Python窗口或ArcToolbox的自定义脚本工具中运行脚本时，才可以使用CURRENT关键字。如果在IDLE或其他开发环境中运行脚本时使用CURRENT关键字，则无法访问当前加载在ArcGIS中的地图文档。需要指出的是，CURRENT关键字不区分大小写，所以也可以使用“current”。

第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(2) 引用磁盘上的地图文档

引用存储在本地计算机或共享服务上的地图文档，只需要提供地图文档文件的路径。这种引用地图文档的方法更为通用，因为使用该方法可以在ArcGIS Python窗口以外（IDLE等其他开发环境中）运行脚本。接下来讨论脚本中函数的参数，程序员可以根据需要每次输入一个新的路径作为参数，使用路径参数可使脚本的应用更为广泛。

例：ReferenceMapDocumentOnDisk.py

```
import arcpy.mapping as mapping
mxd = mapping.MapDocument(r"E:\crime_copy.mxd")
print(mxd.title)
```

Python

```
>>> import arcpy.mapping as mapping
>>> mxd = mapping.MapDocument('CURRENT')
>>> mxd.title = 'Crime project'
>>> mxd.saveACopy(r'C:\ArcpyBook\Ch2\crime_copy.mxd')
>>> mxd2 = mapping.MapDocument('C:\ArcpyBook\Ch2
\crime_copy.mxd')
>>> print(mxd2.title)
Crime project
>>>
```

第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(3) 获取地图文档的图层列表

大多数情况下，获取地图文档中的图层列表是地理处理脚本中的首要工作之一。获取图层列表后，脚本可以循环遍历每个图层并执行某些类型的处理。制图模块中的**ListLayers()** 函数提供获取图层列表的功能。本节将介绍如何获取地图文档中的图层列表。

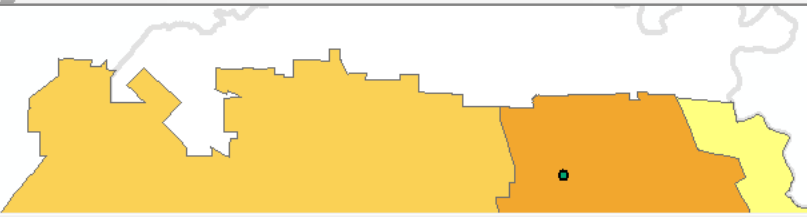
Crime_Ch2.mxd - ArcMap

File Edit View Bookmarks Insert Selection Geoprocessing Customize Windows Help

1:500,000

Table Of Contents

- Crime**
 - ☒ Burglaries in 2009
 - ☒ Crime Density by School District
 - 5.991641 - 53.502834
 - 53.502835 - 147.06045
 - 147.060455 - 287.3462
 - 287.346284 - 605.2552
 - 605.255254 - 772.3181
 - ☒ Bexar County Boundary
- Test_Performance**
 - ☒ Test Performance by School District
 - Test_Perf2
 - Above Average
 - Average
 - Below Average
 - Not in Study
 - ☒ Bexar County Boundary
- Inset_Map**
 - ☒ Bexar County Boundary
 - ☒ Texas Counties



```

Python
>>> mxd.title = 'Crime project'
>>> mxd.saveACopy(r'C:\ArcpyBook\Ch2\crime_copy.mxd')
>>> mxd2 = mapping.MapDocument('C:\ArcpyBook\Ch2\crime_copy.mxd')
>>> print(mxd2.title)
Crime project
>>> layers = mapping.ListLayers(mxd)
>>> for lyr in layers:
...     print(lyr.name)
...
Burglaries in 2009
Crime Density by School District
Bexar County Boundary
Test Performance by School District
Bexar County Boundary
Bexar County Boundary
Texas Counties
School_Districts
Crime_Surface
Bexar County Boundary
>>> |
  
```

61

第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(3) 获取地图文档的图层列表

例：GetListLayers.py

```
import arcpy.mapping as mapping
mxd = mapping.MapDocument(r"E:\crime_copy.mxd")
dfs = arcpy.mapping.ListDataFrames(mxd)
for df in dfs:
    print(df.name)
    layers = mapping.ListLayers(mxd)
    for lyr in layers:
        arcpy.mapping.AddLayer(df, lyr, "AUTO_ARRANGE")
        arcpy.RefreshActiveView()
        arcpy.RefreshTOC()
        print(lyr.name)
```

第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(4) 限制图层列表

有时候并不需要地图文档中全部图层的列表，而仅仅需要图层的子集。`ListLayers()`函数可以限制返回的图层列表。

例：RestrictListLayers.py

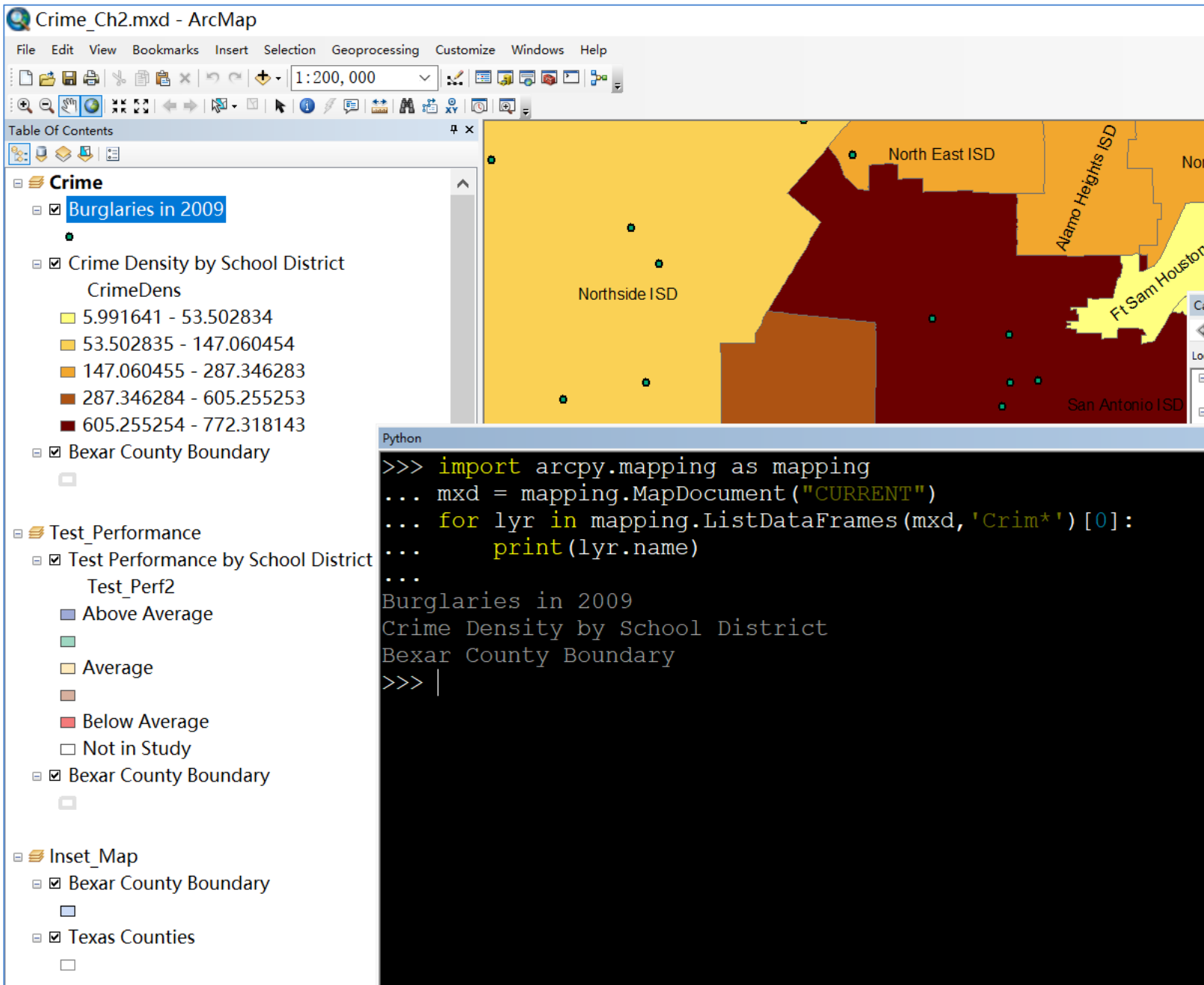
```
import arcpy.mapping as mapping
mxd = mapping.MapDocument("CURRENT")
for df in mapping.ListDataFrames(mxd):
    if (df.name == 'Crime'):
        layers = mapping.ListLayers(mxd,"Burg*",df)
        for layer in layers:
            print(layer.name)
```

第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(4) 限制图层列表

ListDataFrames()函数是arcpy.mapping模块提供的另一种列表函数，该函数返回地图文档中所有数据框的列表。使用该函数循环遍历每个返回的数据框，查找名为Crime的数据框。如果找到这个数据框，就调用ListLayers()函数。该函数的第2个参数是值为“Burg*”的通配符参数，**通配符参数是可选参数**，它的值由任意长度的字符和一个可选字符（*）组成；第3个参数则是对Crime数据框的引用。



Crime_Ch2.mxd - ArcMap

File Edit View Bookmarks Insert Selection Geoprocessing Customize Windows Help

1:500,000

Table Of Contents

- Crime
 - ☒ Burglaries in 2009
 - 5.991641 - 53.502834
 - 53.502835 - 147.06045
 - 147.060455 - 287.3462
 - 287.346284 - 605.2552
 - 605.255254 - 772.3181
 - ☒ Bexar County Boundary
- Test_Performance
 - ☒ Test Performance by School District
 - Above Average
 - Average
 - Below Average
 - Not in Study
 - ☒ Bexar County Boundary
- Inset_Map
 - ☒ Bexar County Boundary
 - ☒ Texas Counties

Python

```
Burglaries in 2009
Crime Density by School District
Bexar County Boundary
Test Performance by School District
Bexar County Boundary
Bexar County Boundary
Texas Counties
School_Districts
Crime_Surface
Bexar County Boundary
>>> import arcpy.mapping as mapping
... mxd = mapping.MapDocument("CURRENT")
... for df in mapping.ListDataFrames(mxd):
...     if (df.name == 'Crime'):
...         layers = mapping.ListLayers(mxd, "Burg*", df)
...         for layer in layers:
...             print(layer.name)
...
...
Burglaries in 2009
>>> |
```

第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(5) 缩放至所选要素

为了更好地显示选择集，用户经常需要将视图缩放至所选要素的范围。Python有几种方法可以程序化地实现这一功能。

缩放至一个数据框的**所有图层**中所选要素的范围，只需要引用当前活动的数据框，并调用**zoomToSelectedFeatures()**方法。

缩放至一个数据框的**特定图层**中所选要素的范围，需要编写的代码更复杂一些。首先，在导入arcpy.mapping模块后，获取对地图文档和Crime数据框的引用。然后，使用ListLayers()函数，传入对数据框的引用参数和通配符参数，搜索名称以Burglaries开头的图层。ListLayers()函数返回一个Python列表。因为本节使用的数据中只有一个图层符合通配符的搜索条件，所以取出列表中的第一个图层，并将其赋值给layer变量。最后，使用**layer.getSelectedExtent()**方法设置数据框的范围。

Crime_Ch2.mxd - ArcMap

File Edit View Bookmarks Insert Selection Geoprocessing Customize Windows Help

1:500,000

Table Of Contents

Crime

☒ Burglaries in 2009

☒ Crime Density by School District

CrimeDens

5.991641 - 53.502834

53.502835 - 147.060454

147.060455 - 287.346283

287.346284 - 605.255253

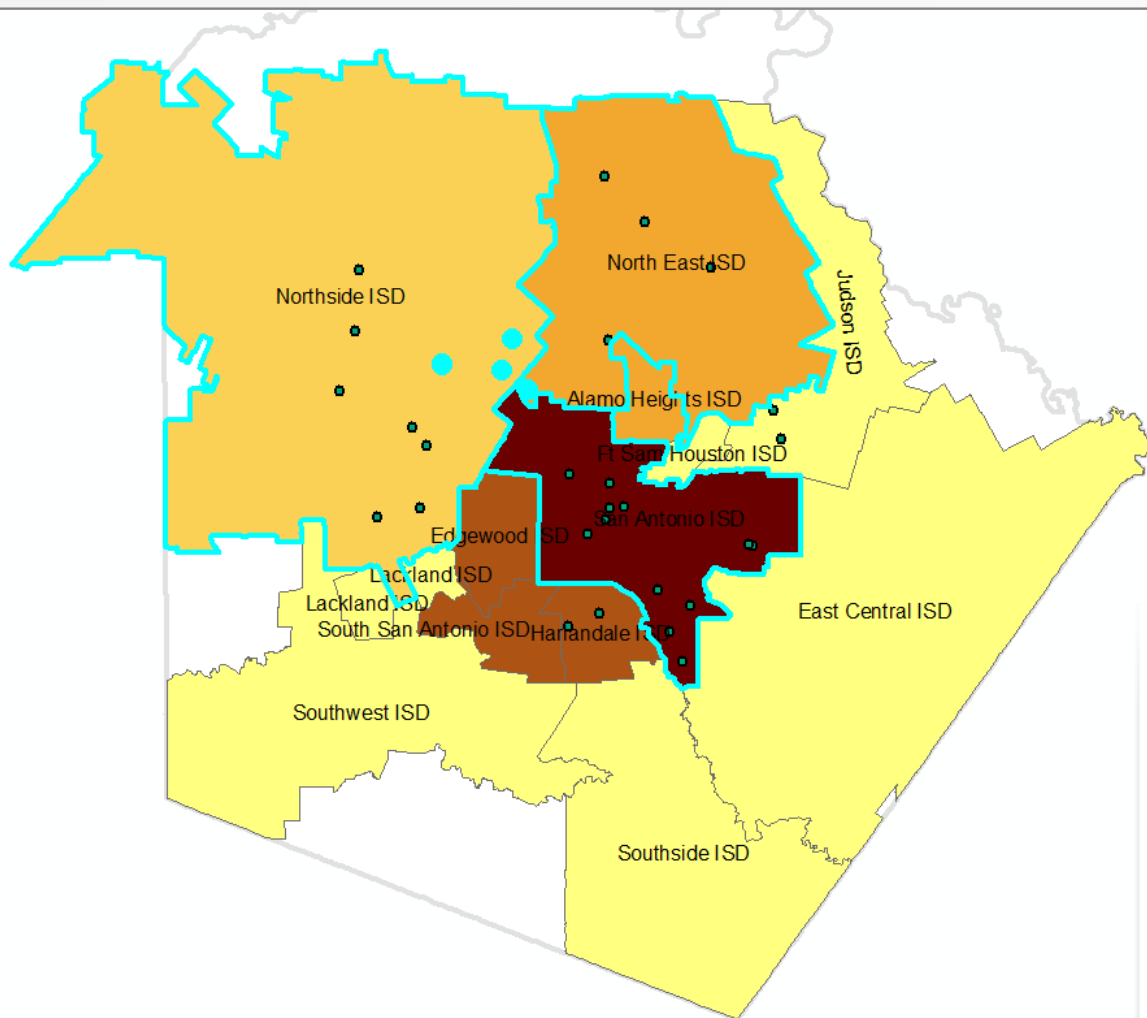
605.255254 - 772.318143

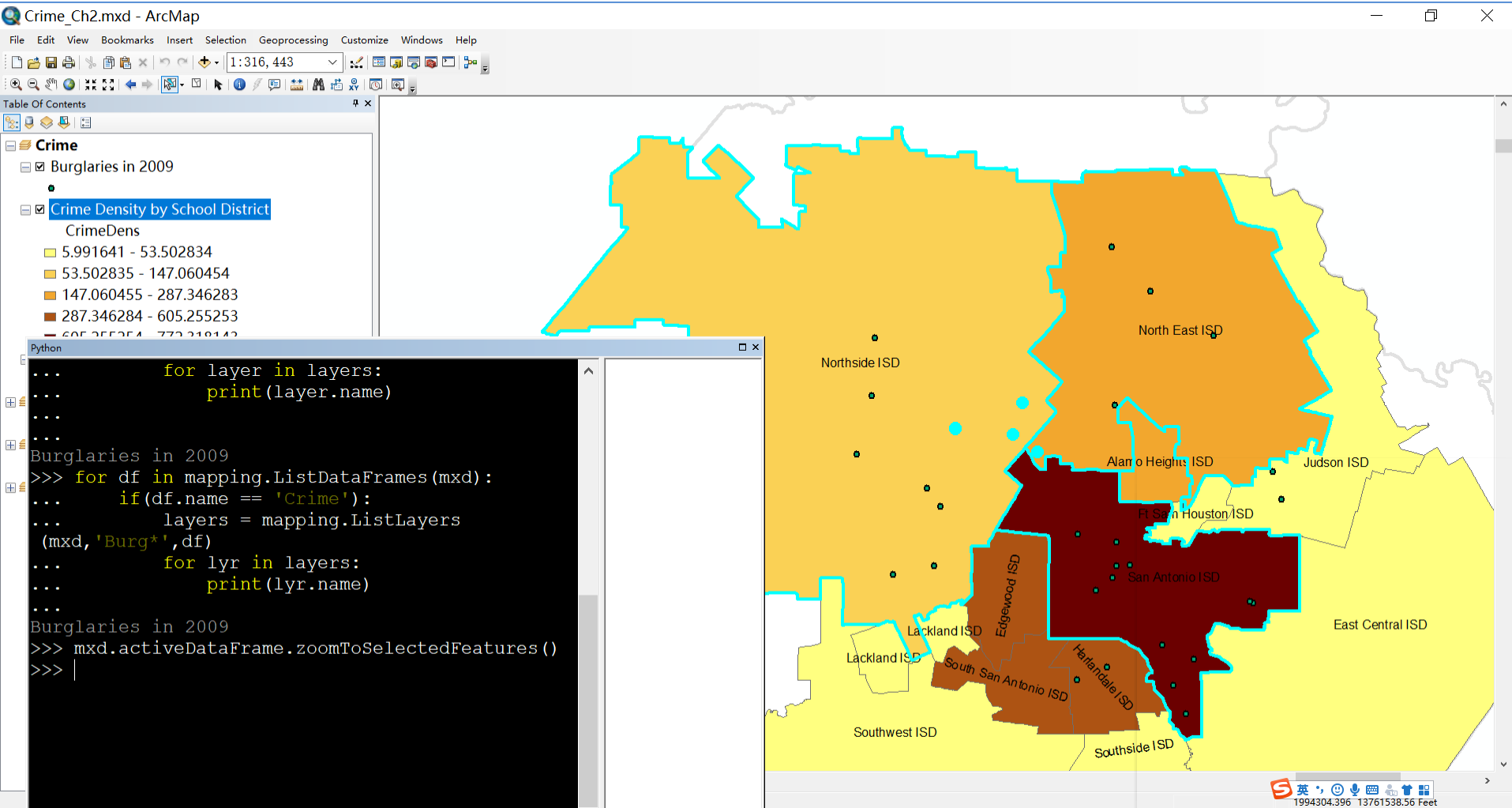
☒ Bexar County Boundary

☒ Test_Performance

☒ Inset_Map

☒ Crime_Inset





Crime_Ch2.mxd - ArcMap

File Edit View Bookmarks Insert Selection Geoprocessing Customize Windows Help

1:600,000

Table Of Contents

Crime

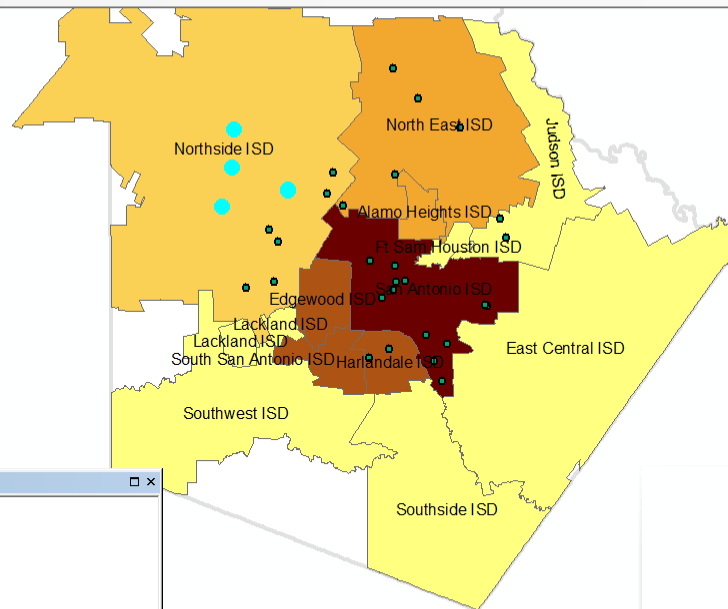
- ☒ Burglaries in 2009
- ☒ Crime Density by School District
 - CrimeDens
 - 5.991641 - 53.502834
 - 53.502835 - 147.060454
 - 147.060455 - 287.346283
 - 287.346284 - 605.255253
 - 605.255254 - 772.318143
- ☒ Bexar County Boundary

Test_Performance

Inset_Map

Python

```
...     for layer in layers:
...         print(layer.name)
...
...
Burglaries in 2009
>>> for df in mapping.ListDataFrames(mxd):
...     if(df.name == 'Crime'):
...         layers = mapping.ListLayers
...         (mxd, 'Burg*', df)
...         for lyr in layers:
...             print(lyr.name)
...
Burglaries in 2009
>>> mxd.activeDataFrame.zoomToSelectedFeatures()
>>>
```



	SPLITDT	SPLITTH	HR	DOW	SHIFT	OFFCODE
2009/11/27	0:00:00	23	Fri	C	220001	
2009/11/28	0:00:00	22	Sat	B	220001	
2009/11/29	0:00:00	0	Sun	C	220001	
2009/11/29	0:00:00	1	Sun	C	220001	
2009/11/29	0:00:00	1	Sun	C	220001	
2009/11/28	0:00:00	19	Sat	B	220001	
2009/11/29	0:00:00	0	Sun	C	220001	
2009/11/28	0:00:00	6	Sat	C	220001	

1887585.184 13746653.25 Feet

第5章 地图文档管理与数据访问

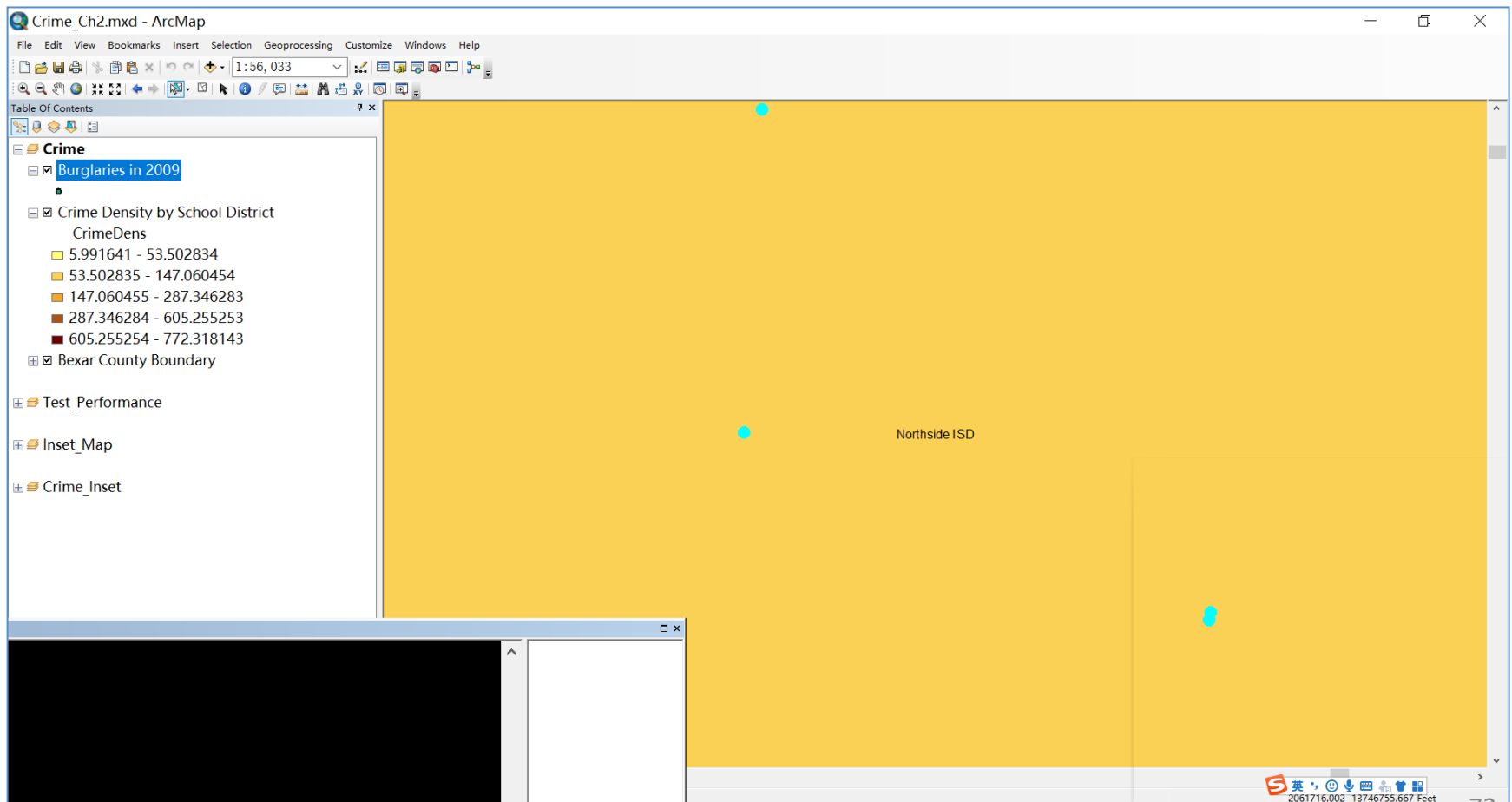
5.3 管理地图文档和图层

(5) 缩放至所选要素

例: ZoomSelectedExtent.py

```
import arcpy.mapping as mapping
mxd = mapping.MapDocument("CURRENT")
df = mapping.ListDataFrames(mxd,"Crime_DataSet")[0]
layer = mapping.ListLayers(mxd,"Burg*",df)[0]
df.extent = layer.getSelectedExtent()
#df.zoomToSelectedFeatures()
arcpy.RefreshActiveView()
```

```
>>> mxd.activeDataFrame.zoomToSelectedFeatures()  
... df = mapping.ListDataFrames(mxd, 'Crime')[0]  
... layer = mapping.ListLayers(mxd, 'Burg*', df)[0]  
... df.extent = layer.getSelectedExtent()  
  
...  
>>> |
```



第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(6) 更改地图范围

常见的情况有以下两种：一是在自动生成地图的过程中，二是在需要创建不同区域或要素的地图时。`arcpy`提供了一些可以改变地图范围的方法，本节将使用定义表达式来改变地图范围。

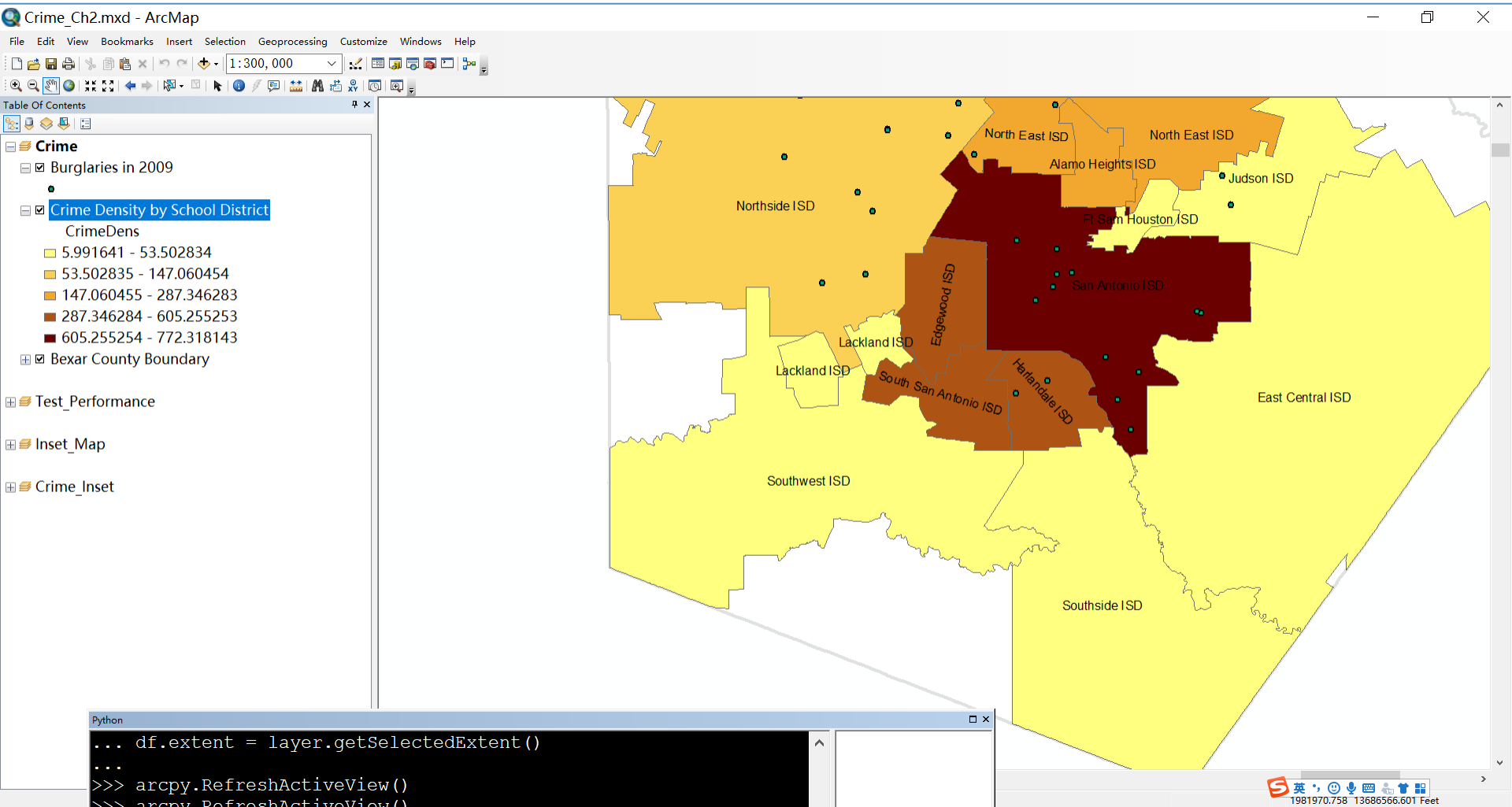
第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(6) 更改地图范围

例: ChangeMapExtent.py

```
import arcpy.mapping as mapping
mxd = mapping.MapDocument("CURRENT")
for df in mapping.ListDataFrames(mxd):
    if (df.name == 'Crime'):
        layers = mapping.ListLayers(mxd,'Crime Density by School
District',df)
        for layer in layers:
            query = ' "NAME" = \'Lackland ISD\''
            layer.definitionQuery = query
            df.extent = layer.getExtent()
```



```
import arcpy.mapping as mapping
mxd = mapping.MapDocument("CURRENT")
for df in mapping.ListDataFrames(mxd):
    if df.name == 'Crime':
        layers = mapping.ListLayers(mxd, 'Crime Density by School District', df)
        for layer in layers:
            query = '"NAME" = \'Lackland ISD\''
            layer.definitionQuery = query
            df.extent = layer.getExtent()
```

Crime_Ch2.mxd - ArcMap

File Edit View Bookmarks Insert Selection Geoprocessing Customize Windows Help

1:60,000

Table Of Contents

- Crime
 - Burglaries in 2009
 - Crime Density by School District
 - CrimeDens
 - 5.991641 - 53.502834
 - 53.502835 - 147.060454
 - 147.060455 - 287.346283
 - 287.346284 - 605.255253
 - 605.255254 - 772.318143
 - Bexar County Boundary
- Test_Performance
- Inset_Map
- Crime_Inset

Layer Properties

Joins & Relates			Time		HTML Popup	
General	Source	Selection	Display	Symbology	Fields	Definition Query
Definition Query: "NAME" = 'Lackland ISD'						
Query Builder...						

Lackland ISD

76

第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(7) 添加图层到地图文档

例：AddLayersMapDocument.py

```
import arcpy.mapping as mapping
mxd = mapping.MapDocument("CURRENT")
df = mapping.ListDataFrames(mxd)[0]
layer =
mapping.Layer(r"C:\ArcpyBook\data\School_Districts.lyr")
mapping.AddLayer(df,layer,"AUTO_ARRANGE")
```

Crime_Ch2.mxd - ArcMap

File Edit View Bookmarks Insert Selection Geoprocessing Customize Windows Help

1:600,000

Table Of Contents



Crime

☒ Burglaries in 2009

☒ Crime Density by School District

CrimeDens

5.991641 - 53.502834

53.502835 - 147.060454

147.060455 - 287.346283

287.346284 - 605.255253

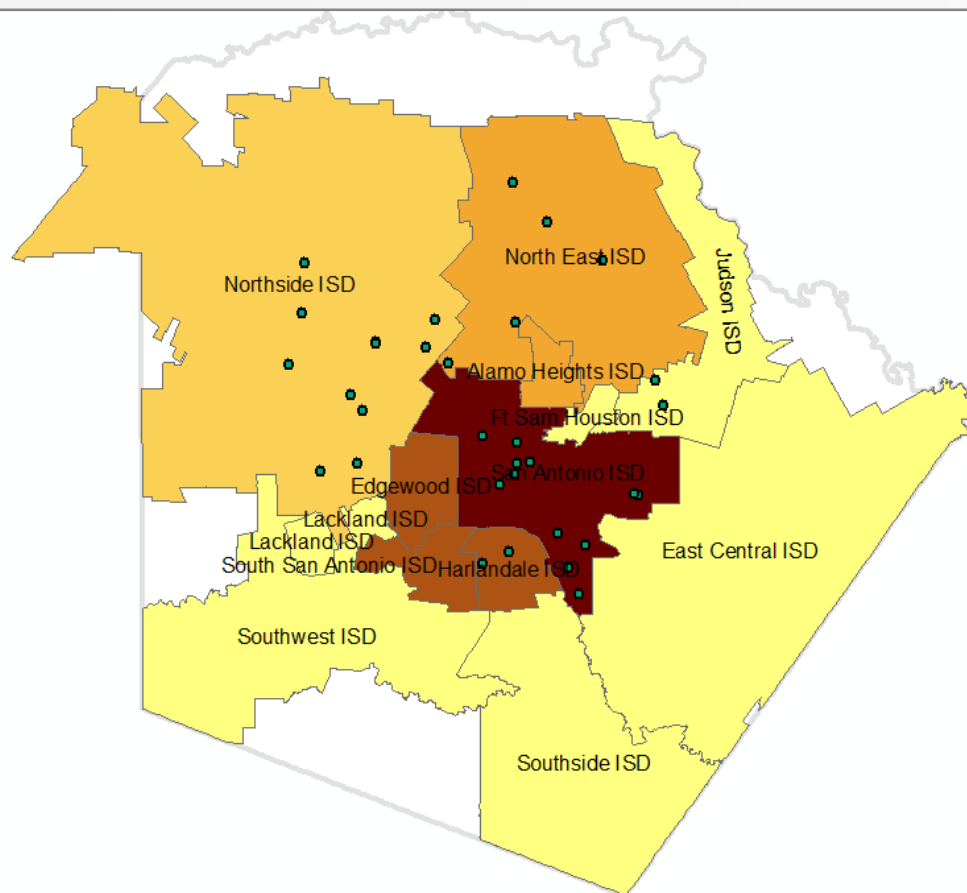
605.255254 - 772.318143

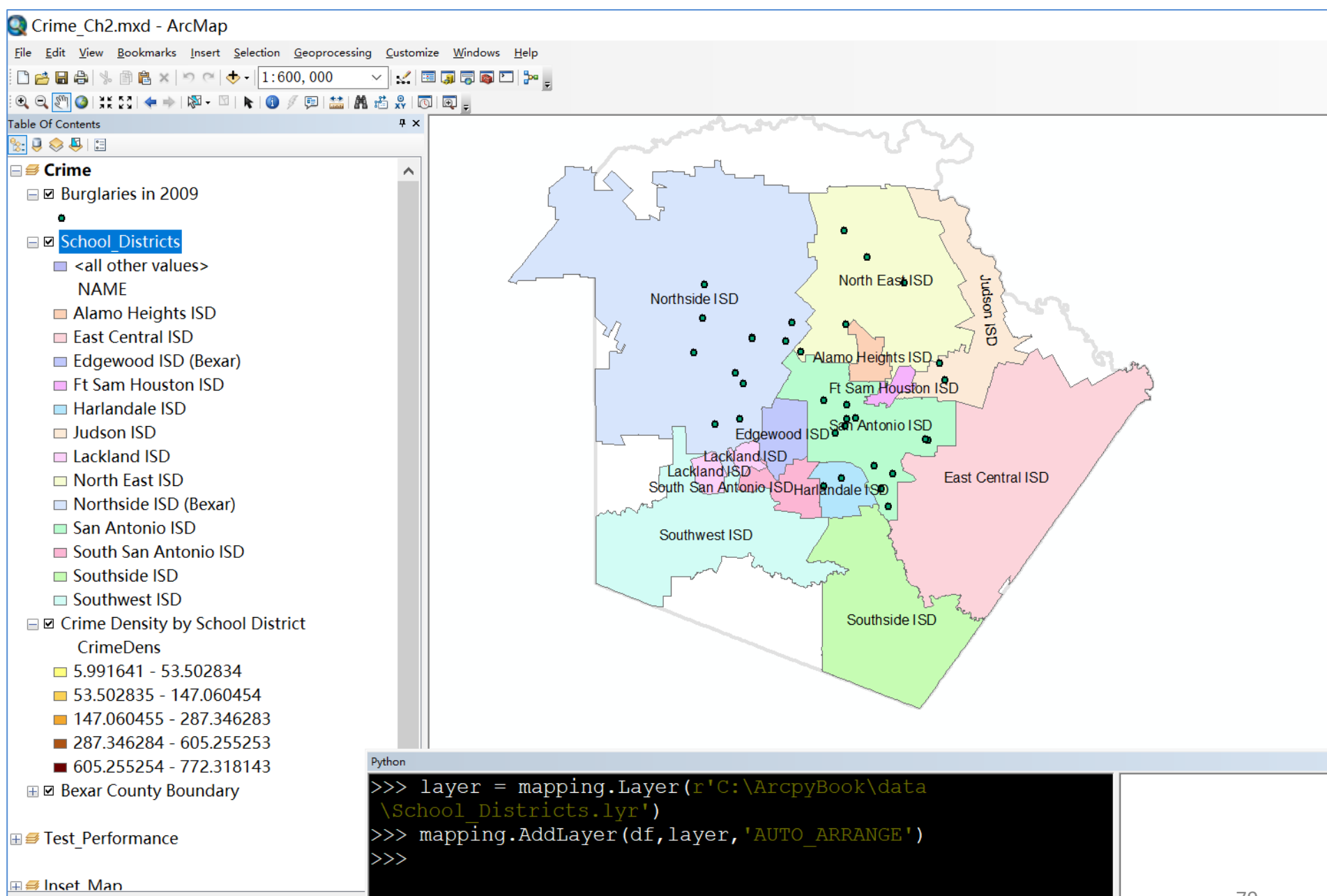
☒ Bexar County Boundary

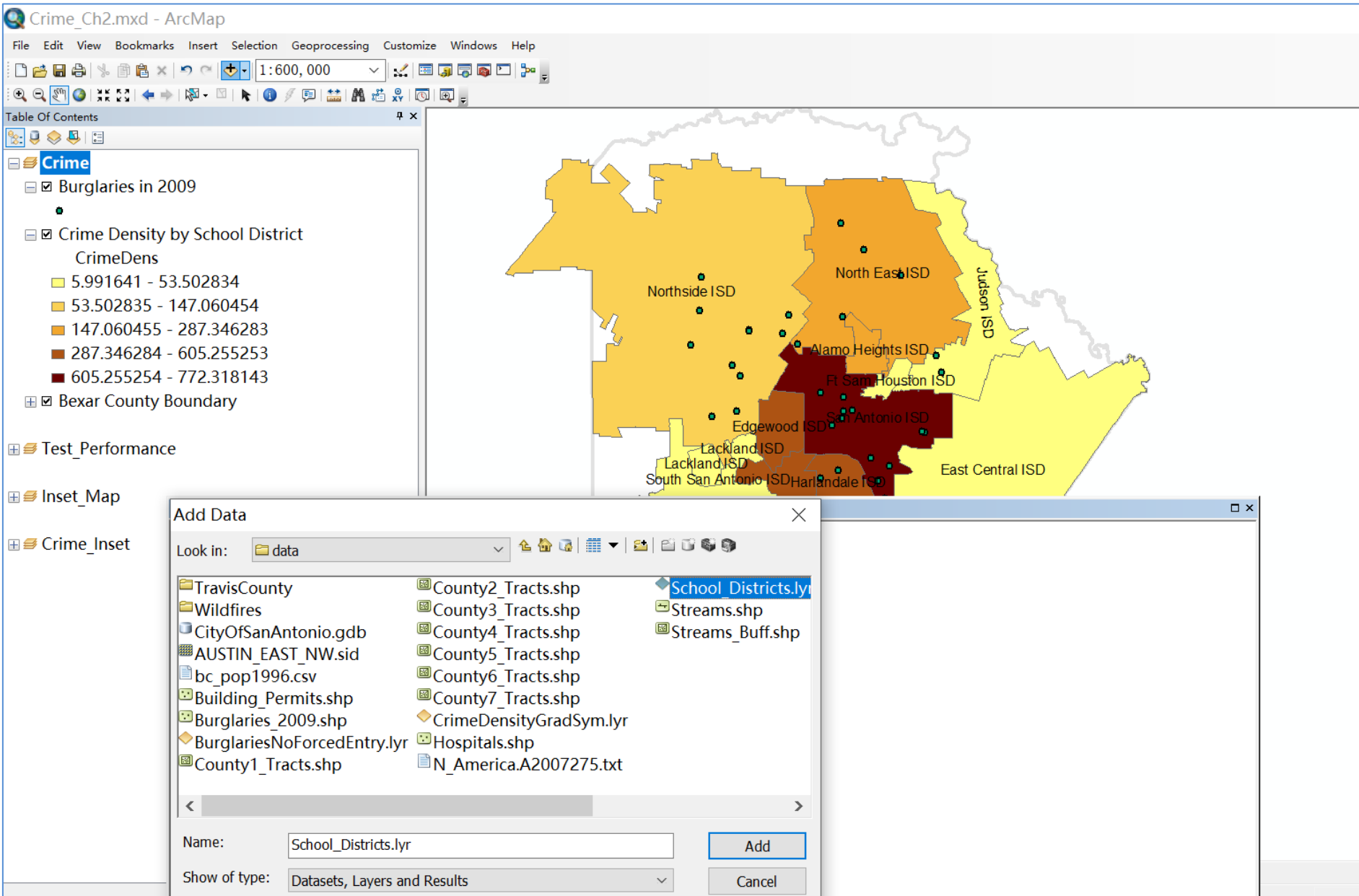
☒ Test_Performance

☒ Inset_Map

☒ Crime_Inset







第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(8) 插入图层到地图文档

AddLayer() 函数仅提供把图层添加到数据框或图层组中的功能，并且只可以使用自动排序方式自动地放置图层的位置，或者选择放置在顶层或底层。然而，**InsertLayer()** 函数可以准确地指定图层添加到数据框或图层组中的位置。**InsertLayer()** 函数使用一个参考图层来指定位置，新图层将会添加在指定参考图层的上方或下方。因为**InsertLayer()**函数需要使用参考图层，所以**不能在空数据框中使用该函数**。

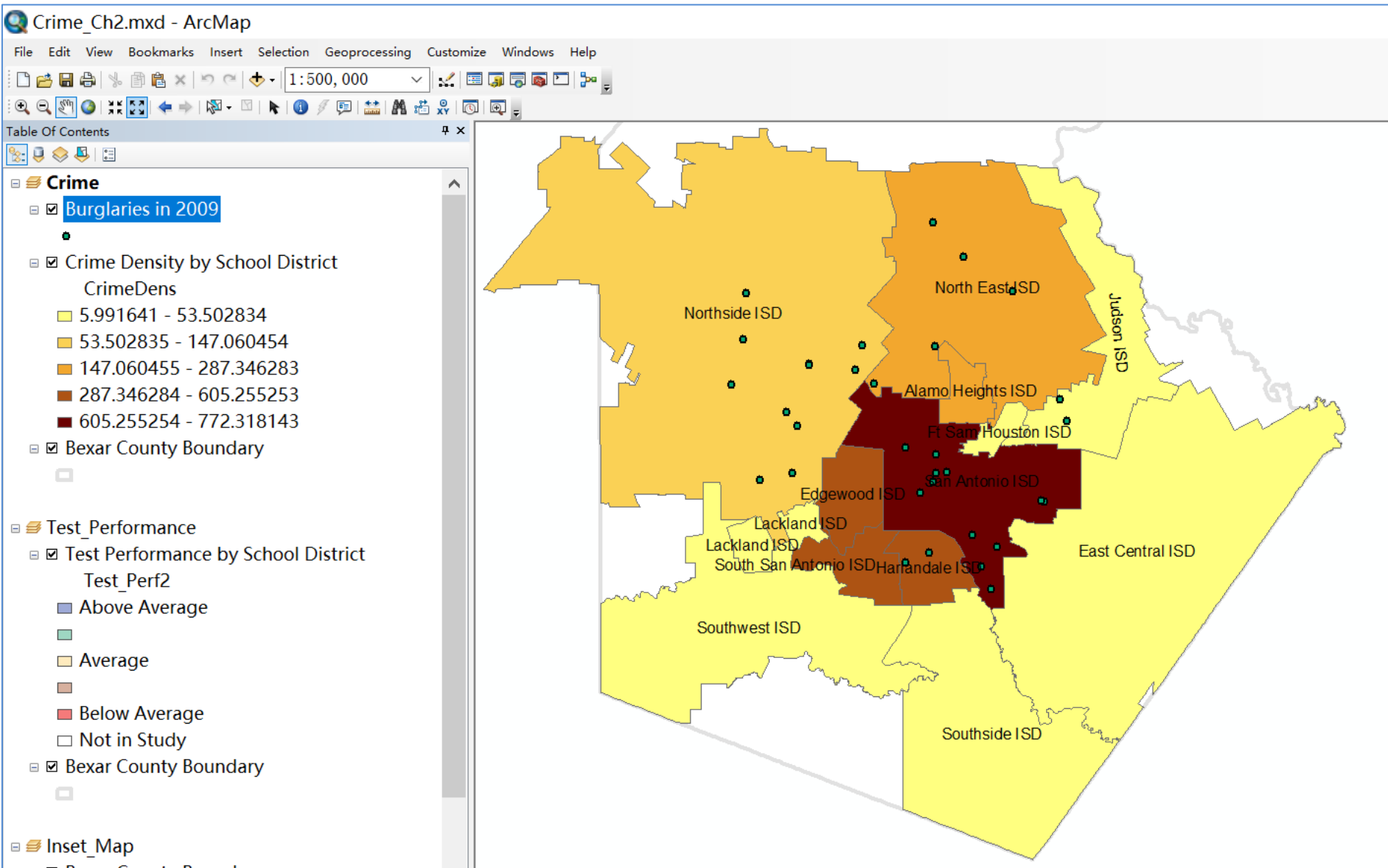
第5章 地图文档管理与数据访问

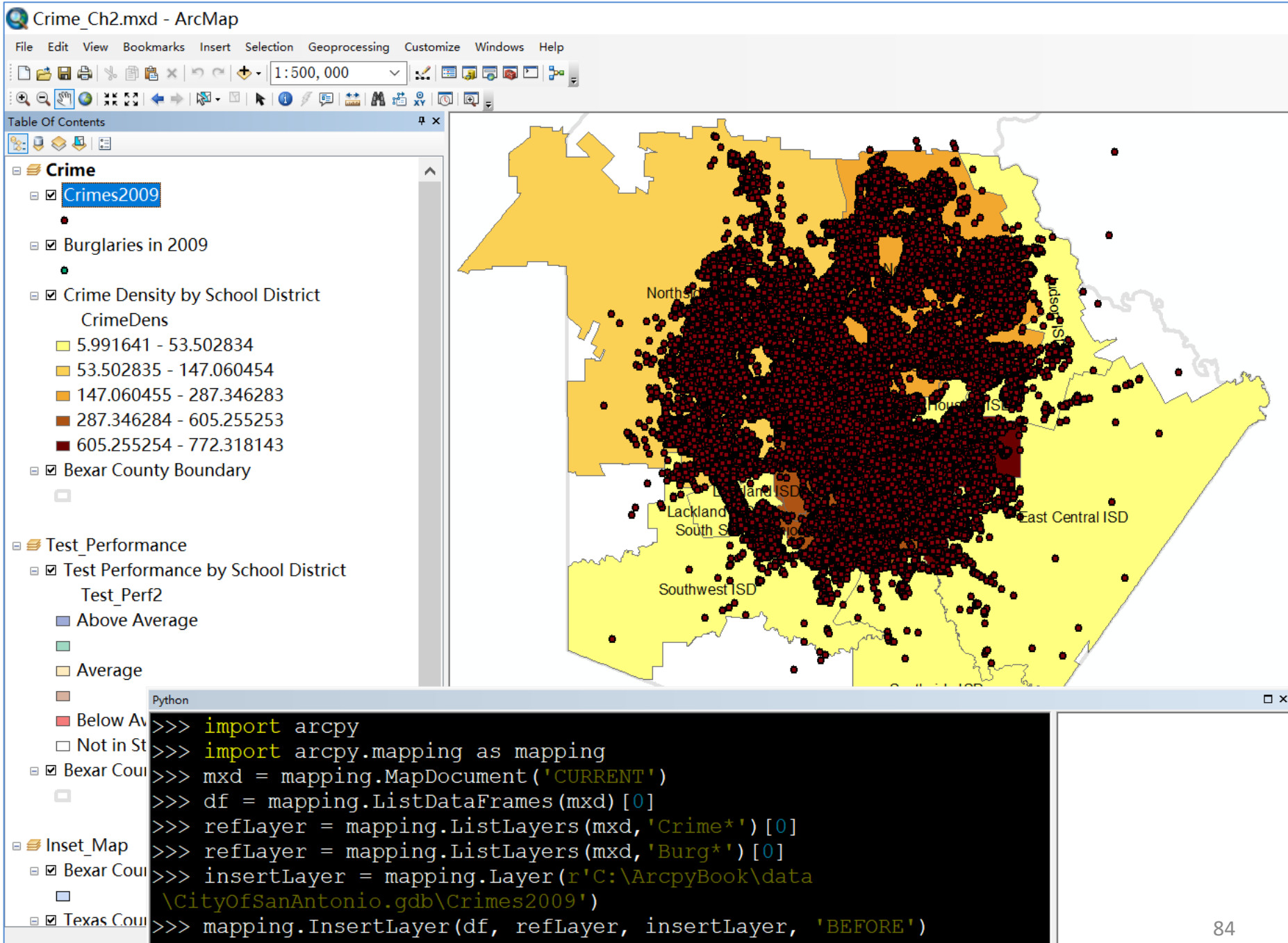
5.3 管理地图文档和图层

(8) 插入图层到地图文档

例: InsertLayerMapDocument.py

```
import arcpy.mapping as mapping
mxd = mapping.MapDocument("CURRENT")
df = mapping.ListDataFrames(mxd, "Crime")[0]
refLayer = mapping.ListLayers(mxd, "Burg*", df)[0]
insertLayer =
mapping.Layer(r"C:\ArcpyBook\data\CityOfSanAntonio.gdb\C
rimes2009")
mapping.InsertLayer(df, refLayer, insertLayer, "BEFORE")
```





第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(9) 更新图层的符号系统

arcpy.mapping 模块的 **UpdateLayer()** 函数具有更新图层符号系统的功能。例如，可以将图层的符号系统由分级颜色更新为分级符号。UpdateLayer() 也可以用来**更新各种图层属性**，但在**默认情况下是更新符号系统**。因为 UpdateLayer() 是一个多功能的函数，既能改变符号系统，也能改变其他图层属性，所以需要掌握 UpdateLayer() 函数的各个参数。

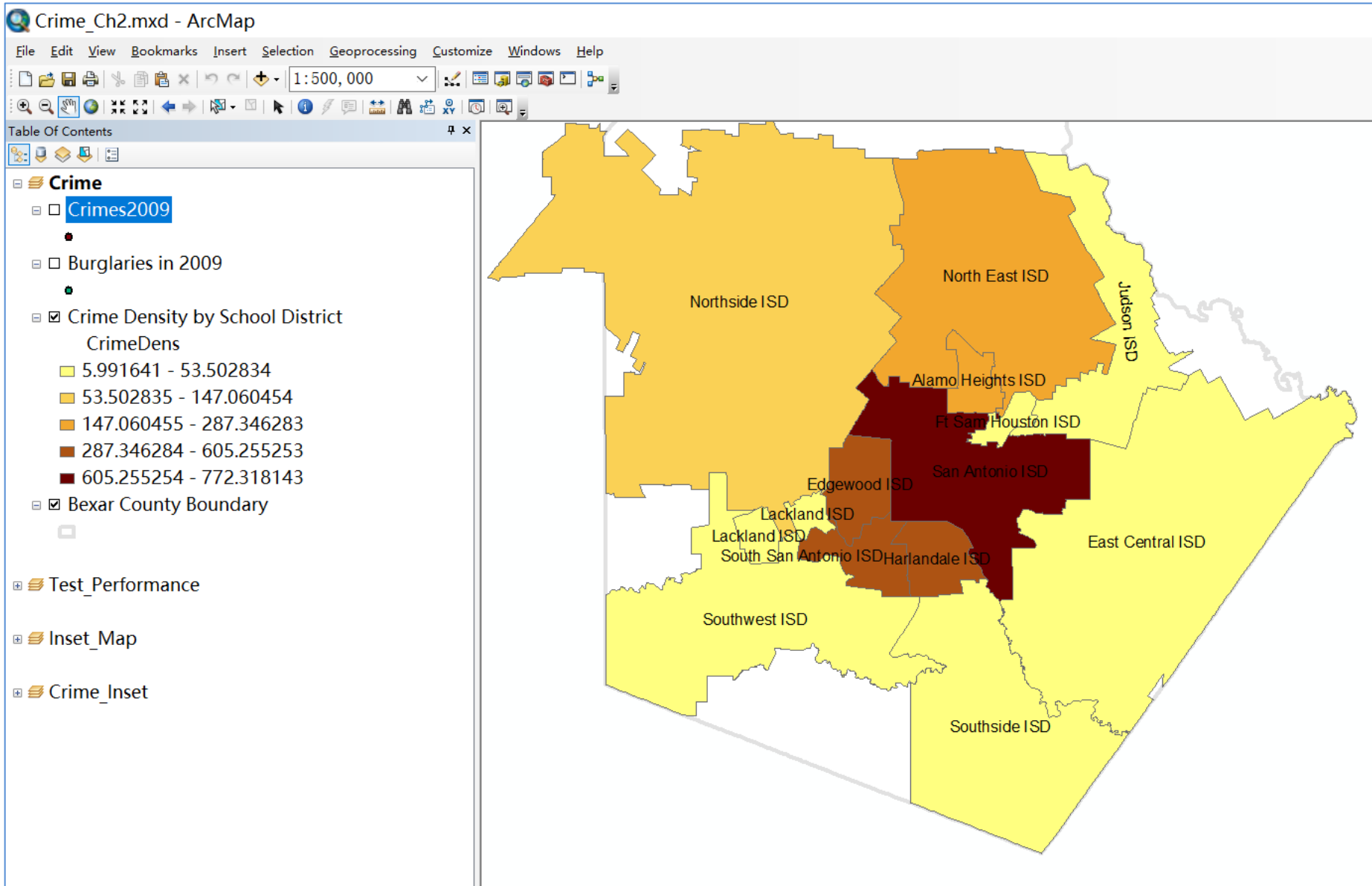
第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(9) 更新图层的符号系统

例: UpdateLayerSymbology.py

```
import arcpy.mapping as mapping
mxd = mapping.MapDocument("CURRENT")
df = mapping.ListDataFrames(mxd, "Crime")[0]
updateLayer = mapping.ListLayers(mxd, "Crime Density by
School District", df)[0]
sourceLayer =
mapping.Layer(r"C:\ArcpyBook\data\CrimeDensityGradSym
.lyr")
mapping.UpdateLayer(df, updateLayer, sourceLayer, True)
```



第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(10) 更新图层属性

UpdateLayer()函数可以用来更新有限数量的图层属性。所有可在“图层属性”对话框中找到的属性都可使用 UpdateLayer 函数进行修改，包括字段别名、符号系统、定义查询和标注字段等。一种常见的情况是，有一个图层被添加到多个地图文档中，而GIS分析人员需要在全部地图文档中改变所有该图层实例的某个特定属性。

第5章 地图文档管理与数据访问

5.3 管理地图文档和图层

(10) 更新图层属性

例: UpdateLayerProperties.py

```
import arcpy.mapping as mapping
mxd = mapping.MapDocument("CURRENT")
df = mapping.ListDataFrames(mxd, "Crime")[0]
updateLayer = mapping.ListLayers(mxd, "Crimes2009", df)[0]
sourceLayer =
mapping.Layer(r"C:\ArcpyBook\data\BurglariesNoForcedEntry.lyr")
mapping.UpdateLayer(df, updateLayer, sourceLayer, False)
```

The End

人生苦短
python 當歌

