



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30日星期二

课程安排:

36个学时，其中授课24个学时，实验12个学时

成绩:

出勤5% + 实验报告25% + 考试70%

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

第6章 数据链接查找与修复

第7章 地图制图与输出

第8章 地理处理工具的执行

第9章 地理处理工具的创建

第10章 数据查询与选择

第11章 数据访问模块

第12章 获取GIS数据的列表和描述

第6章 数据链接查找与修复

查找丢失的数据源

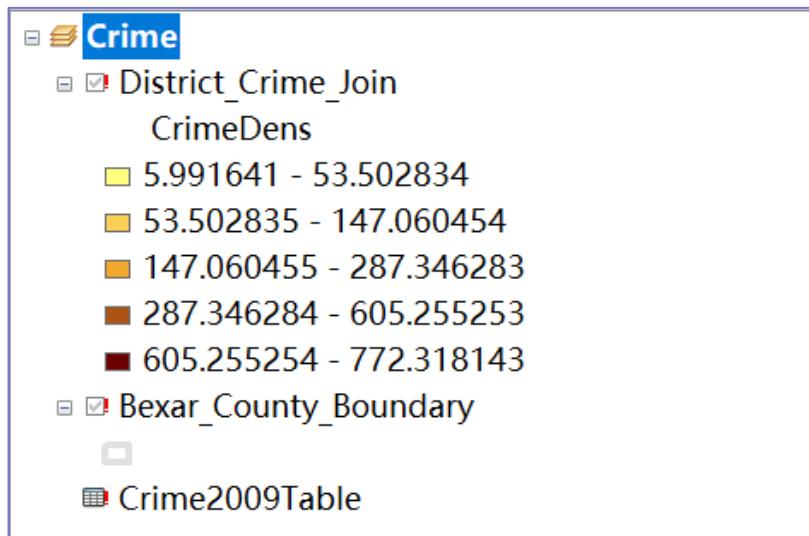
修复丢失的数据源

修复单个图层和
表对象

查找文件夹中所有
地图文档内丢失
的数据源

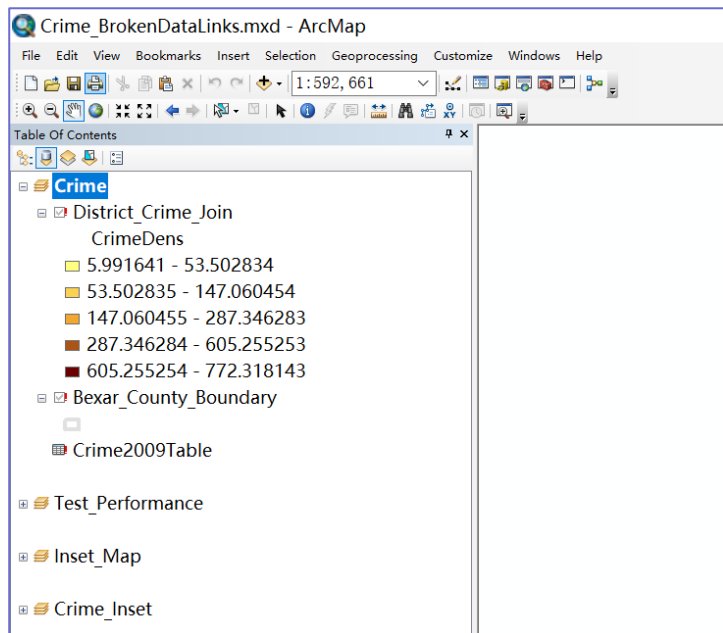
6.1 查找丢失的数据源

- 在ArcMap中，图层名称前的**红色叹号**表示该图层的数据链接已经丢失。
arcpy.mapping模块中的ListBrokenDataSources()函数，可以返回一个图层列表，列表中的对象是地图文档或图层文件中已经丢失了数据链接的图层。



6.1 查找丢失的数据源

▪ ListBrokenDataSources()函数



```
>>> import arcpy.mapping as mapping
... mxd = mapping.MapDocument('CURRENT')
... listBrokenDS = mapping.ListBrokenDataSources(mxd)
... for lyr in listBrokenDS:
...     print(lyr.name)
...
District_Crime_Join
Bexar_County_Boundary
District_Crime_Join
Bexar_County_Boundary
Bexar_County_Boundary
Texas_Counties_LowRes
School_Districts
Crime_surf
Bexar_County_Boundary
Crime2009Table
>>>
```

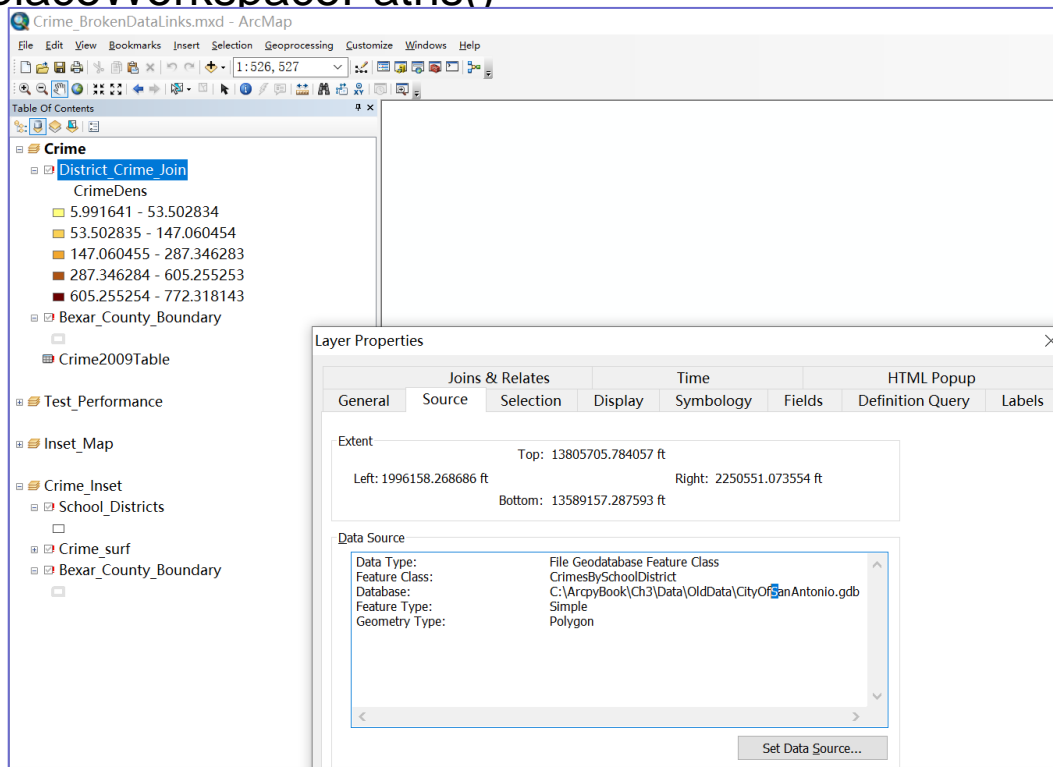
6.2 修复丢失的数据源

- `findAndReplaceWorkspacePaths()`

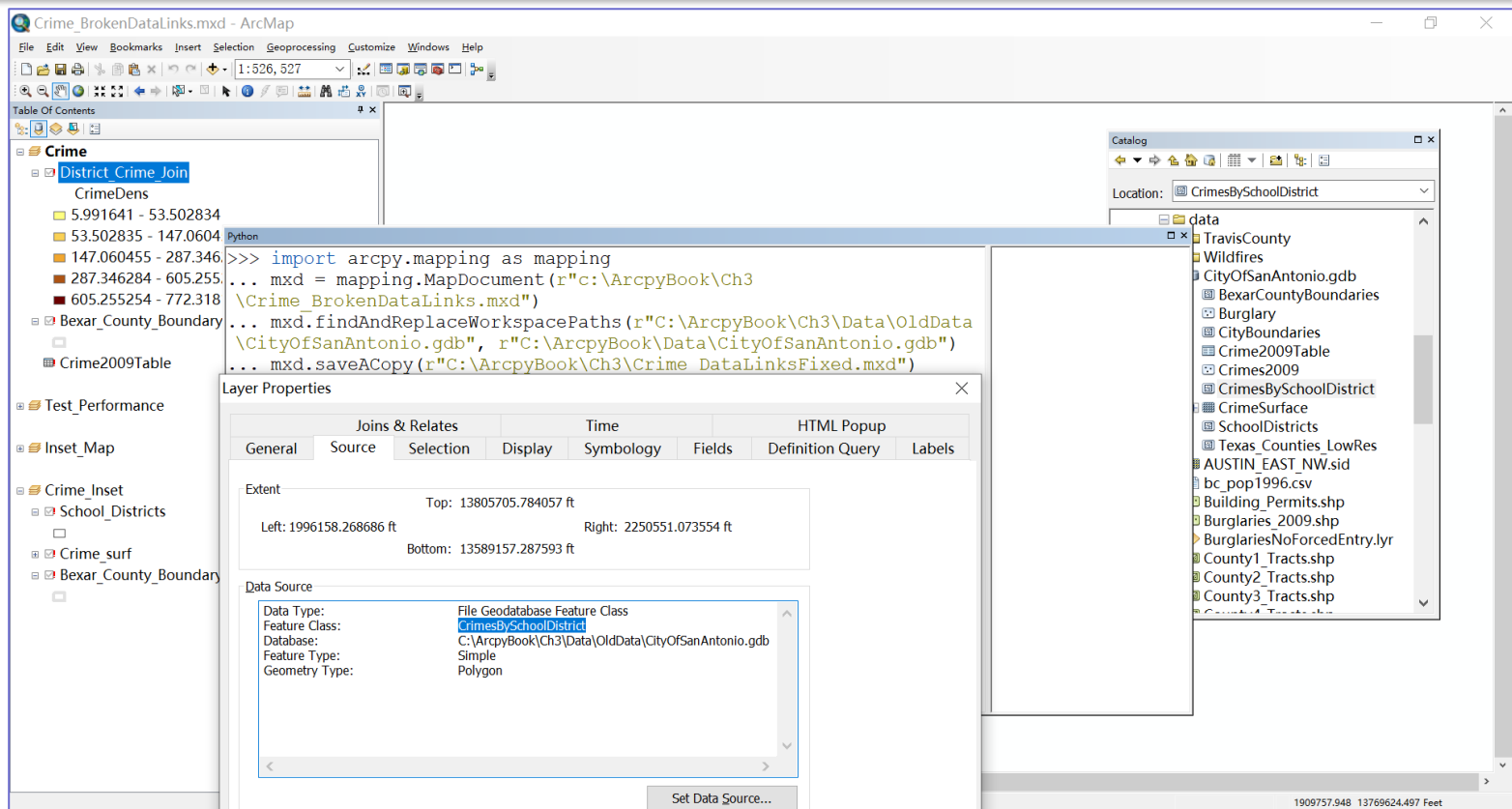
`MapDocument.findAndReplaceWorkspacePaths()`方法用于执行全局查找，并替换地图文档中图层和表的工作空间路径。它还可以一次替换多种工作空间类型的路径，例如可以同时替换个人地理数据库和文件地理数据库两种工作空间类型。

6.2 修复丢失的数据源

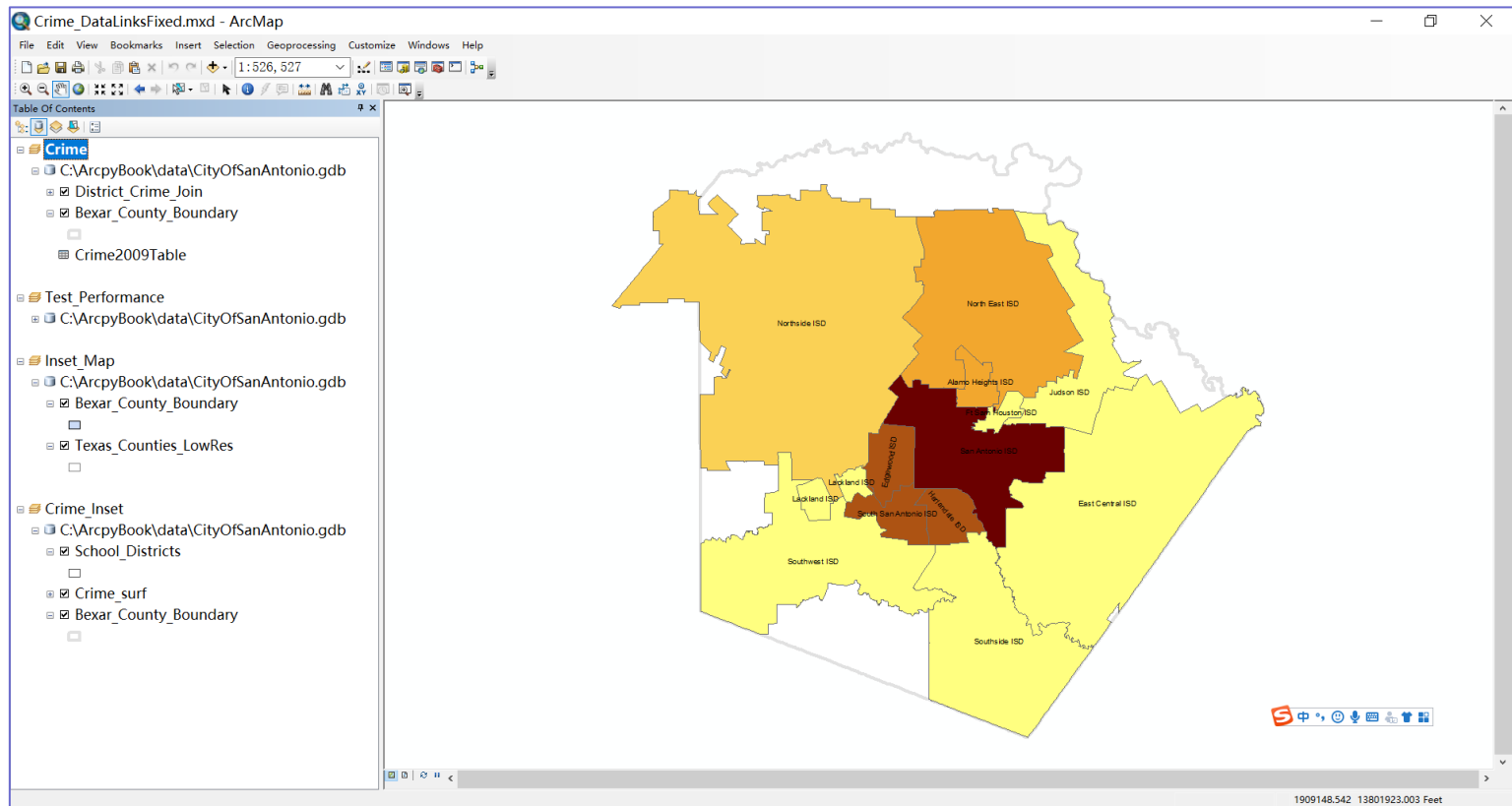
- findAndReplaceWorkspacePaths()



6.2 修复丢失的数据源



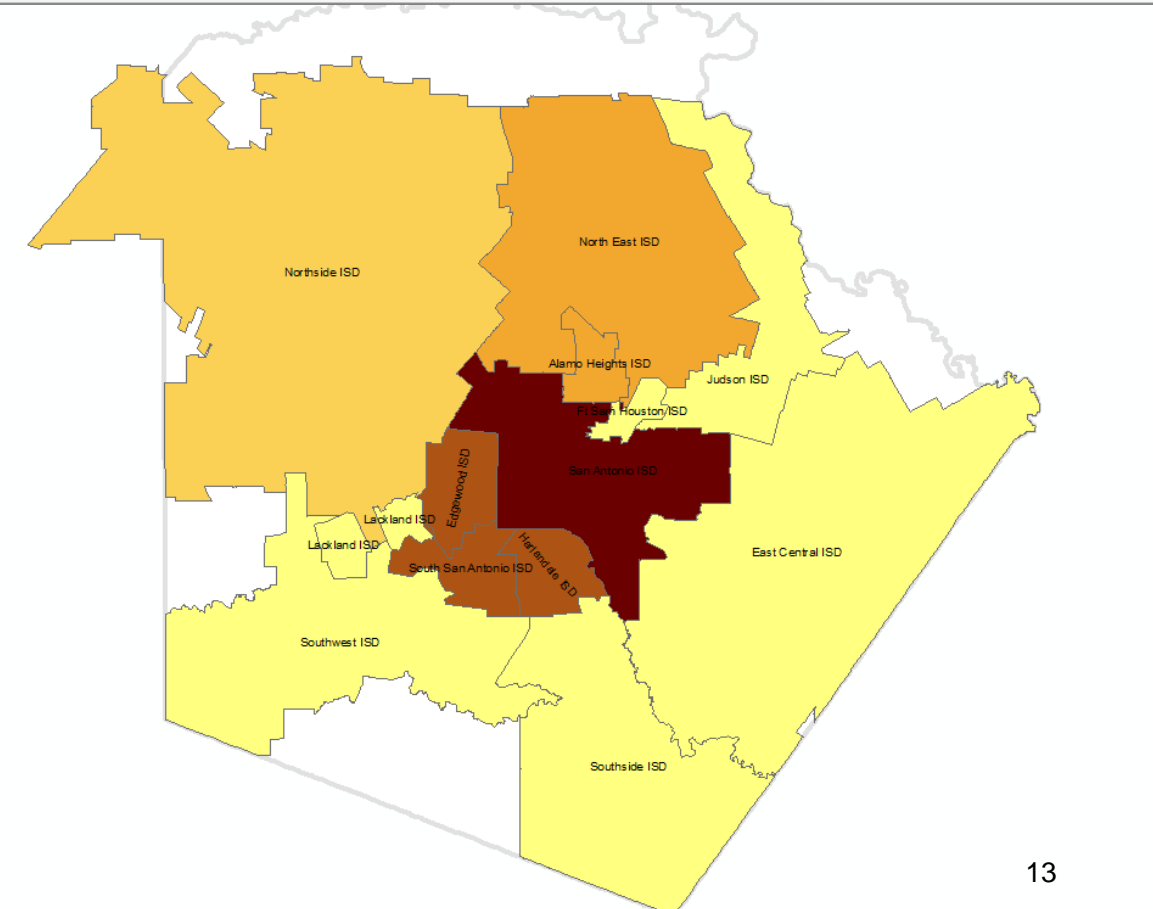
6.2 修复丢失的数据源



6.2 修复丢失的数据源

- `replaceWorkspaces()`

`MapDocument.replaceWorkspaces()` 方法与 `MapDocument.findAndReplaceWorkspacePaths()` 方法类似，但是前者还允许用户从一种工作空间类型转换至另一种工作空间类型。例如，由文件地理数据库转换至个人地理数据库。然而，此方法一次只能替换一种工作空间。



Crime_DataLinksFixed.mxd - ArcMap

File Edit View Bookmarks Insert Selection Geoprocessing Customize Windows Help

1:526,527

Table Of Contents

- Crime
 - C:\ArcpyBook\data\CityOfSanAntonio.gdb
 - District_Crime_Join
 - Bexar_County_Boundary
 - Crime2009Table
- Test_Performance
 - C:\ArcpyBook\data\CityOfSanAntonio.gdb
- Inset_Map
 - C:\ArcpyBook\data\CityOfSanAntonio.gdb
 - Bexar_County_Boundary
 - Texas_Counties
- Crime_Inset
 - C:\ArcpyBook\data\CityOfSanAntonio.gdb
 - School_Districts
 - Crime_surf
 - Bexar_County_Boundary

Map View

Northside ISD North East ISD Alamo Heights ISD Judson ISD P.E. San Antonio ISD

Python

```
>>> import arcpy.mapping as mapping
... mxd = mapping.MapDocument(r"c:\ArcpyBook\Ch3\Crime_DataLinksFixed.mxd")
... mxd.replaceWorkspaces(r"c:\ArcpyBook\data\CityOfSanAntonio.gdb",
... "FILEGDB_WORKSPACE", r"c:\ArcpyBook\new_data\CityOfSanAntonio_Personal.mdb", "ACCESS_WORKSPACE")
... mxd.saveACopy(r"c:\ArcpyBook\Ch3\Crime_DataLinksUpdated.mxd")
...
>>> |
```

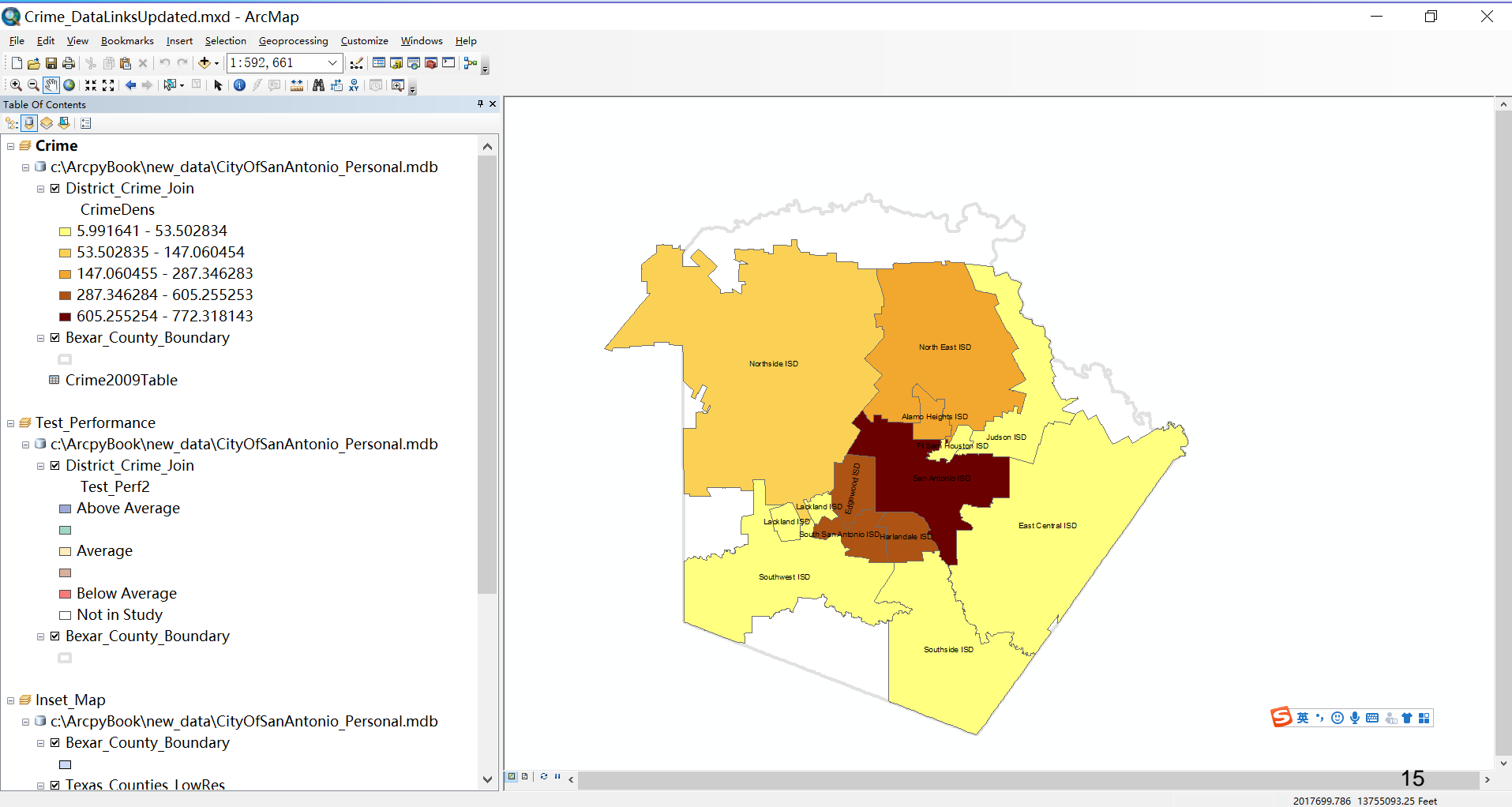
Catalog

Location: CrimesBySchoolDistrict

- Ch5
- Ch6
- Ch7
- Ch8
- Ch9
- code
- data
- GenerateRandomPoints
- New_Data
 - CityOfSanAntonio_Personal.mdb
 - BexarCountyBoundaries
 - CityBoundaries
 - Crime2009Table
 - Crimes2009
 - CrimesBySchoolDistrict
 - SchoolDistricts
 - Texas_Counties_LowRes
- PythonAddInTest
- ch4Crime.sddraft
- Intel

14

1982277.235 13747990.592 Feet



GDB与MDB的区别？

6.2 修复丢失的数据源

■ 工作空间类型

* ACCESS_WORKSPACE: A personal geodatabase or Access workspace

* ARCINFO_WORKSPACE: An ArcInfo coverage workspace

* CAD_WORKSPACE: A CAD file workspace

* EXCEL_WORKSPACE: An Excel file workspace

* FILEGDB_WORKSPACE: A file geodatabase workspace

* OLEDB_WORKSPACE: An OLE database workspace

* PCCOVERAGE_WORKSPACE: A PC ARC/INFO Coverage workspace

* RASTER_WORKSPACE: A raster workspace

* SDE_WORKSPACE: An SDE geodatabase workspace

* SHAPEFILE_WORKSPACE: A shapefile workspace

* TEXT_WORKSPACE: A text file workspace

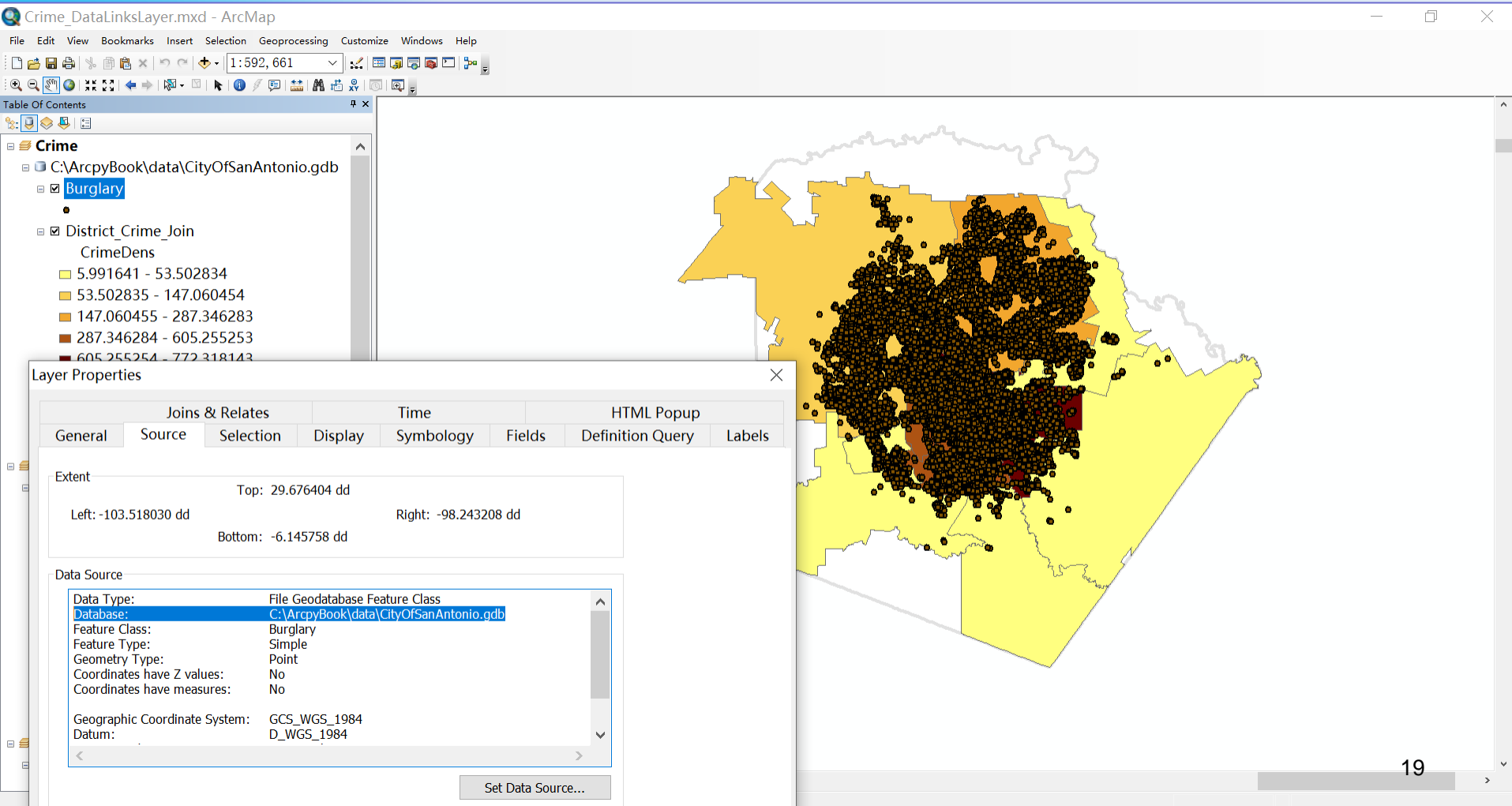
* TIN_WORKSPACE: A TIN workspace

* VPF_WORKSPACE: A VPF workspace

6.3 修复单个图层和表对象

- `replaceDataSource()`

`Layer`和`TableView`类都有`replaceDataSource()`方法，该方法可以改变单个图层或表的工作空间路径、工作空间类型和数据集名称。



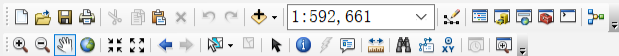
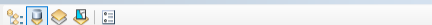


Table Of Contents



Crime

C:\ArcpyBook\data\CityOfSanAntonio.gdb

Burglary

District_Crime_Join

CrimeDens

5.991641 - 53.502834

53.502835 - 147.060454

147.060455 - 287.346283

287.346284 - 685.355253

Python

>>> import arcpy.mapping as mapping

... mxd = mapping.MapDocument(r"c:\ArcpyBook\Ch3\Crime_DataLinksLayer.mxd")

... df = mapping.ListDataFrames(mxd, "Crime") [0]

... lyr = mapping.ListLayers(mxd, "Burglary", df) [0]

... lyr.replaceDataSource(r"c:\ArcpyBook\data", "SHAPEFILE_WORKSPACE", "Burglaries_2009")

... mxd.saveACopy(r"c:\ArcpyBook\Ch3\Crime_DataLinksNewLayer.mxd")

...

Test_P

C:\A

Di

Be

Be

Be

Be

Be

Be

Be

Be

Be

Be

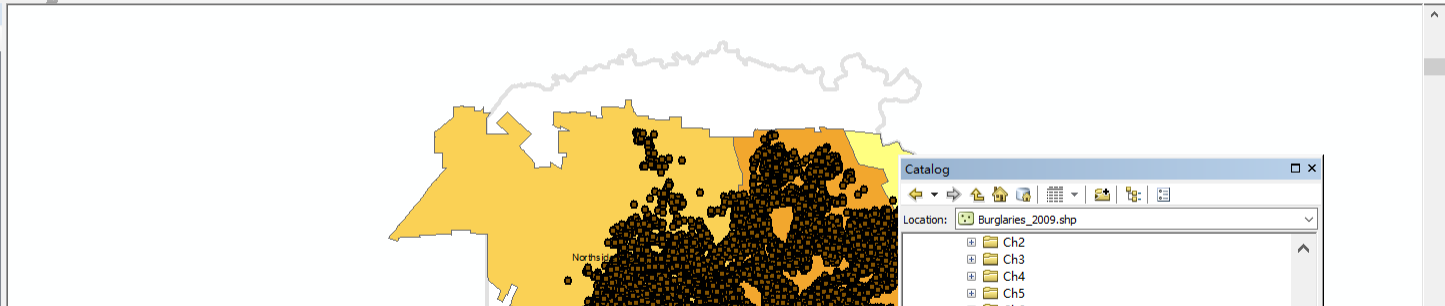
Be

Be

Be

Be

Be



Catalog

Location: Burglaries_2009.shp

Ch2

Ch3

Ch4

Ch5

Ch6

Ch7

Ch8

Ch9

code

data

TravisCounty

Wildfires

CityOfSanAntonio.gdb

AUSTIN_EAST_NW.sid

bc_pop1996.csv

Building_Permits.shp

Burglaries_2009.shp

BurglariesNoForcedEntry.lyr

County1_Tracts.shp

County2_Tracts.shp

County3_Tracts.shp

County4_Tracts.shp

County5_Tracts.shp

County6_Tracts.shp

County7_Tracts.shp

CrimeDensityGradSym.lyr

Hospitals.shp

N_America.A2007275.txt

School_Districts.lyr

Streams.shp

Streams_Buff.shp

GenerateRandomPoints

_layer, {w

st of Layer

ocument (

in a map do

.lyr) fil

ject):

a MapDocum

(*) and ch

the results

a DataFram

Catalog

Location: Home - ArcpyBook\Ch3

- Home - ArcpyBook\Ch3
 - Folder Connections
 - C\
 - ArcpyBook
 - Appendix1
 - Appendix2
 - Ch1
 - Ch10
 - Ch11
 - Ch12
 - Ch13
 - Ch2
 - Ch3
 - Ch4
 - Ch5
 - Ch6
 - Ch7
 - Ch8
 - Ch9
 - code
 - data
 - TravisCounty
 - Wildfires
 - CityOfSanAntonio.gdb
 - AUSTIN_EAST_NW.sid
 - bc_pop1996.csv
 - Building_Permits.shp
 - Burglaries_2009.shp
 - BurglariesNoForcedEntry.lyr
 - County1_Tracts.shp
 - County2_Tracts.shp
 - County3_Tracts.shp

6.4 查找文件夹中所有地图文档内丢失的数据源

在许多部门的工作中经常会遇到这样的情况，即需要将数据从一个工作空间移动到另一个工作空间，或从一种工作空间类型转换至另一种工作空间类型。当移动数据或转换工作空间类型时，任何地图文档或图层中引用的数据源都会丢失链接。手动查找每个丢失的数据源是一项非常繁琐的工作，比较方便的方法是通过创建地理处理脚本，自动查找文件夹或文件夹列表中所有丢失的数据源。

6.4 查找文件夹中所有地图文档内丢失的数据源

```
import arcpy.mapping as mapping, os
f = open('BrokenDataList.txt', 'w')
for root, dirs, files in os.walk("c:\\ArcpyBook"):
    for name in files:
        filename = os.path.join(root, name)
        if ".mxd" in filename:
            mxd = mapping.MapDocument(filename)
            f.write("MXD: " + filename + "\n")
            brknList = mapping.ListBrokenDataSources(mxd)
            for brknItem in brknList:
                print("Broken data item: " + brknItem.name + " in " + filename)
                f.write("\t" + brknItem.name + "\n")
print("All done")
f.close()
```

6.4

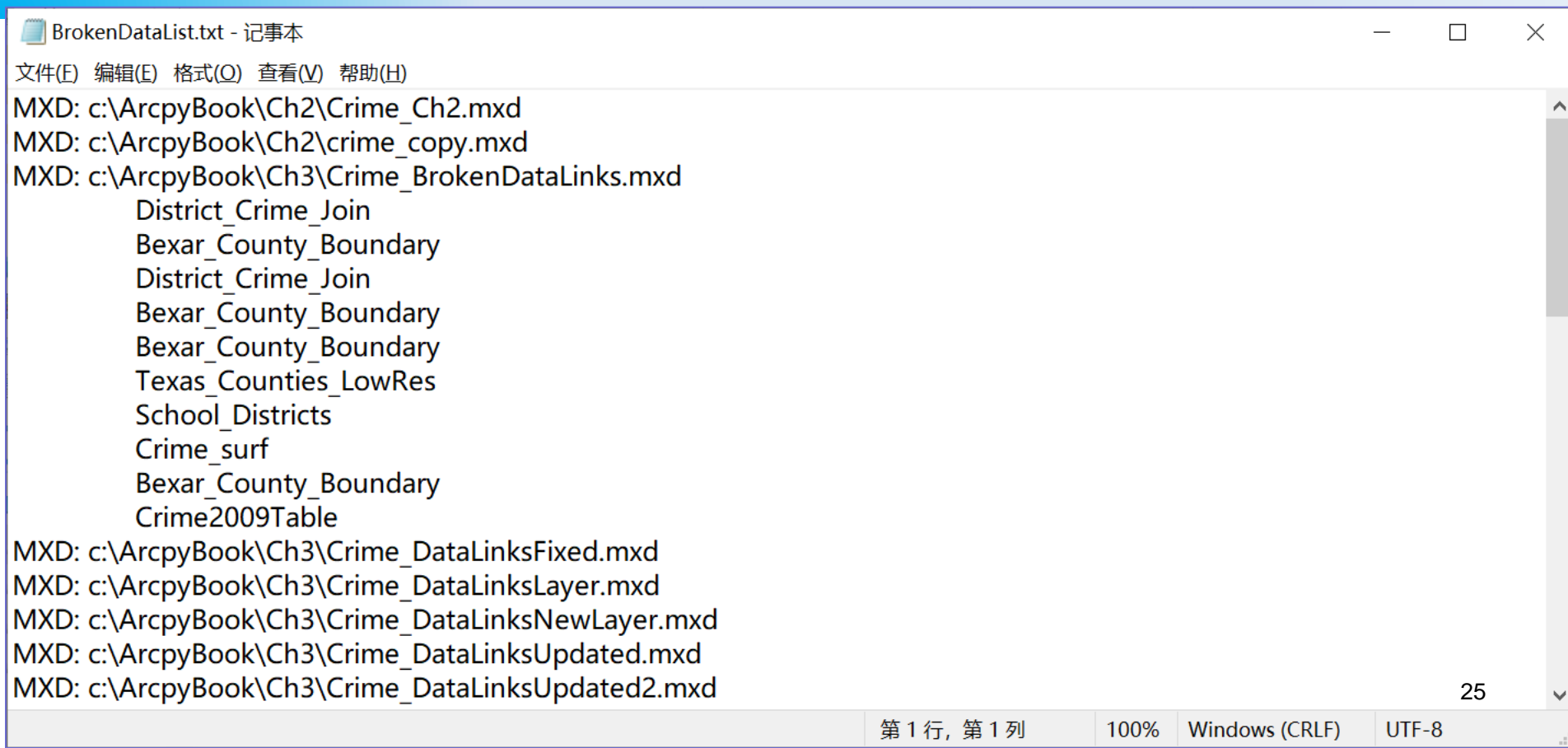
Python

```

Broken data item: District_Crime_Join in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: Bexar_County_Boundary in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: District_Crime_Join in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: Bexar_County_Boundary in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: Bexar_County_Boundary in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: Texas_Counties_LowRes in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: School_Districts in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: Crime_surf in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: Bexar_County_Boundary in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: Crime2009Table in c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
Broken data item: GridIndexFeatures in c:\ArcpyBook\Ch4\Topographic.mxd
Broken data item: Locator_Mask Current Page in c:\ArcpyBook\Ch4\Topographic.mxd
Broken data item: Locator_Mask in c:\ArcpyBook\Ch4\Topographic.mxd
Broken data item: Page Labels in c:\ArcpyBook\Ch4\Topographic.mxd
Broken data item: Burglaries in 2009 in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: Crime Density by School District in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: Bexar County Boundary in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: Test Performance by School District in c:\ArcpyBook\data\TravisCounty
\Crime.mxd
Broken data item: Bexar County Boundary in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: Bexar County Boundary in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: Texas Counties in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: School_Districts in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: Crime_Surface in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: Bexar County Boundary in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: Crime2009Table in c:\ArcpyBook\data\TravisCounty\Crime.mxd
Broken data item: District_Crime_Join in c:\ArcpyBook\data\TravisCounty
\Crime_BrokenDataLinks.mxd
Broken data item: Bexar_County_Boundary in c:\ArcpyBook\data\TravisCounty
\Crime_BrokenDataLinks.mxd
Broken data item: District_Crime_Join in c:\ArcpyBook\data\TravisCounty
\Crime_BrokenDataLinks.mxd
Broken data item: Bexar County Boundary in c:\ArcpyBook\data\TravisCounty

```


6.4 查找文件夹中所有地图文档内丢失的数据源



```
BrokenDataList.txt - 记事本
文件(E) 编辑(E) 格式(O) 查看(V) 帮助(H)
MXD: c:\ArcpyBook\Ch2\Crime_Ch2.mxd
MXD: c:\ArcpyBook\Ch2\crime_copy.mxd
MXD: c:\ArcpyBook\Ch3\Crime_BrokenDataLinks.mxd
    District_Crime_Join
    Bexar_County_Boundary
    District_Crime_Join
    Bexar_County_Boundary
    Bexar_County_Boundary
    Texas_Counties_LowRes
    School_Districts
    Crime_surf
    Bexar_County_Boundary
    Crime2009Table
MXD: c:\ArcpyBook\Ch3\Crime_DataLinksFixed.mxd
MXD: c:\ArcpyBook\Ch3\Crime_DataLinksLayer.mxd
MXD: c:\ArcpyBook\Ch3\Crime_DataLinksNewLayer.mxd
MXD: c:\ArcpyBook\Ch3\Crime_DataLinksUpdated.mxd
MXD: c:\ArcpyBook\Ch3\Crime_DataLinksUpdated2.mxd
```

第 1 行, 第 1 列 100% Windows (CRLF) UTF-8

教材P74

6.4 查找文件夹中所有地图文档内丢失的数据源

- `os.walk()`

`os.walk()` 方法用于通过在目录树中行走输出在目录中的文件名。

`os.walk()` 方法是一个简单易用的文件、目录遍历器，可以帮助我们高效的处理文件、目录方面的事情。

6.4 查找文件夹中所有地图文档内丢失的数据源

- `os.walk()`

返回的是一个三元组(**root,dirs,files**):

- ◆ **root** 所指的是当前正在遍历的这个文件夹的本身的地址
- ◆ **dirs** 是一个 **list** , 内容是该文件夹中所有的目录的名字(不包括子目录)
- ◆ **files** 同样是 **list** , 内容是该文件夹中所有的文件(不包括子目录)

6.4 查找文件夹中所有地图文档内丢失的数据源

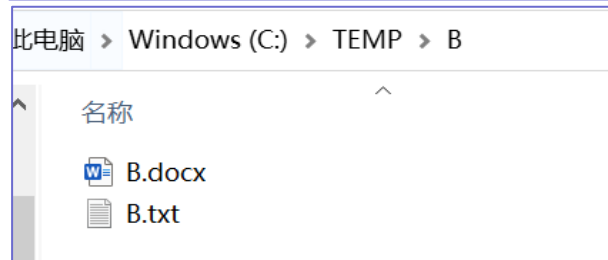
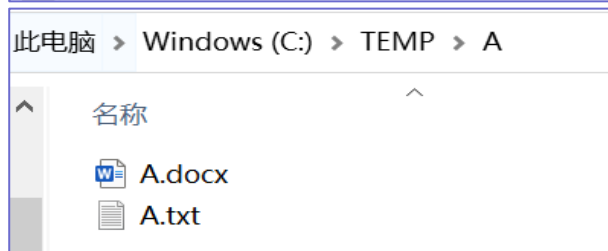
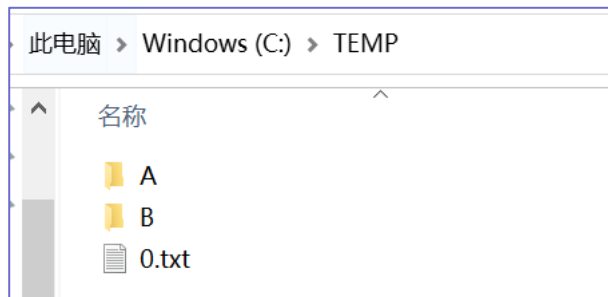
- os.walk()

```
1 import os
2 def walkFile(file):
3     for root, dirs, files in os.walk(file):
4
5         # root 表示当前正在访问的文件夹路径
6         # dirs 表示该文件夹下的子目录名list
7         # files 表示该文件夹下的文件list
8
9         # 遍历文件
10        for f in files:
11            print(os.path.join(root, f))
12
13        # 遍历所有的文件夹
14        for d in dirs:
15            print(os.path.join(root, d))
```

6.4 查找文件夹中所有地图文档内丢失的数据源

▪ os.walk()

```
1 import os
2 def walkFile(file):
3     for root, dirs, files in os.walk(file):
4
5         # root 表示当前正在访问的文件夹路径
6         # dirs 表示该文件夹下的子目录名list
7         # files 表示该文件夹下的文件list
8
9         # 遍历文件
10        for f in files:
11            print(os.path.join(root, f))
12
13        # 遍历所有的文件夹
14        for d in dirs:
15            print(os.path.join(root, d))
```



6.4 查找文件夹中所有地图文档内丢失的数据源

- `os.walk()`

```
1 import os
2 def walkFile(file):
3     for root, dirs, files in os.walk(file):
4
5         # root 表示当前正在访问的文件夹路径
6         # dirs 表示该文件夹下的子目录名list
7         # files 表示该文件夹下的文件list
8
9         # 遍历文件
10        for f in files:
11            print(os.path.join(root, f))
12
13        # 遍历所有的文件夹
14        for d in dirs:
15            print(os.path.join(root, d))
```

```
1 walkFile(r'C:\TEMP')
```

```
C:\TEMP\0.txt
C:\TEMP\A
C:\TEMP\B
C:\TEMP\A\A.docx
C:\TEMP\A\A.txt
C:\TEMP\B\B.docx
C:\TEMP\B\B.txt
```

The End

