

Модуль ArcPy mapping



ArcPy mapping

- доступен для всех уровней лицензий
- Содержит функциональные возможности для
 - n обработки существующих документов карты (**.mxd**), фреймов данных и файлов слоев (**.lyr**);
 - n для автоматизации печати и экспорта
- Необходим для создания атласов

ArcPy mapping

Классы	Функции
DataFrame/Time	AddLayer
GraduatedColorSym	AddLayerToGroup
GraphicElement	CreateMapSDDraft
LabelClass	InsertLayer
Layer/Time	Layer
LegendElement	ListBrokenDataSource
MapDocument	ListdataFrames
MapSurroundElement	ListLayers
PictureElement	ListLayoutElements
RasterClassified	ListTableViews
StyleItems	MapDocument
TableView	MoveLayer
TextElement...	RemoveLayer....

Класс MapDocument

Синтаксис

MapDocument (mxd_path)

mxd_path

Строка, включающая полный путь и имя файла существующего документа карты (.mxd) или строка, содержащая ключевое слово CURRENT.

```
mxd = arcpy.mapping.MapDocument("D:/Python/Project.mxd")
```

```
mxd = arcpy.mapping.MapDocument("CURRENT")
```

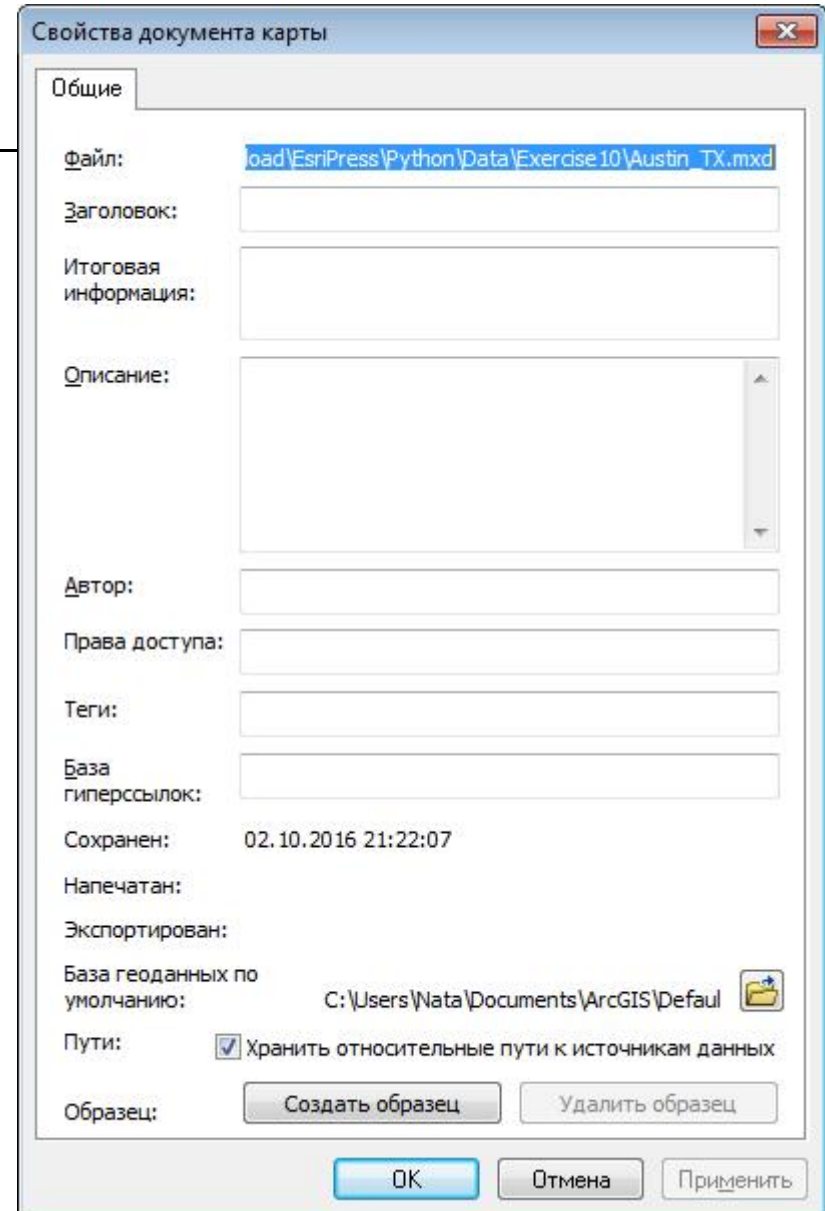
Чтобы использовать ключевое слово **CURRENT** в инструменте-скрипте, необходимо отключить фоновую обработку.

*Переменная, которая ссылается на объект MapDocument, включает блокировку файла документа карты. Рекомендуется удалять ссылку на объект MapDocument, используя команду Python **del** в конце скрипта или в выражении Python try/except.*

Класс MapDocument

Свойства

- n **activeDataFrame** – чт.; объект DataFrame
- n **activeView** – чт./запись; строка
- n **author** – чт./запись; строка
- n **credits** – чт.; строка
- n **dataDrivenPages** – чт.; объект DataDrivenPages
- n **dateExported** – чт.; объект Python datetime
- n **datePrinted** – чт.; объект Python datetime
- n **dateSaved** – чт.; объект Python datetime
- n **description** – чт./запись; строка
- n **filePath** – чт.; строка
- n **hyperlinkBase** – чт./запись; строка
- n **isDDPEnabled** – чт.; логический
- n **pageSize** – чт.; именованный кортеж Python
- n **relativePaths** – чт./запись; логический
- n **summary** – чт./запись; строка
- n **tags** – чт./запись; строка
- n **title** – чт./запись; строка





Класс MapDocument

Методы

n deleteThumbnail()

n findAndReplaceWorkspacePaths
(find_workspace_path,
replace_workspace_path, {validate})

n makeThumbnail()

n replaceWorkspaces(old_workspace_path,
old_workspace_type,
new_workspace_path,
new_workspace_type, {validate})

n save()

n saveACopy(file_name, {version})

Класс DataFrame

СИНТАКСИС

```
ListDataFrames (map_document, {wildcard})
```

ListDataFrames всегда возвращает объект списка Python, даже если возвращается только один фрейм данных.

Чтобы вернуть объект **DataFrame**

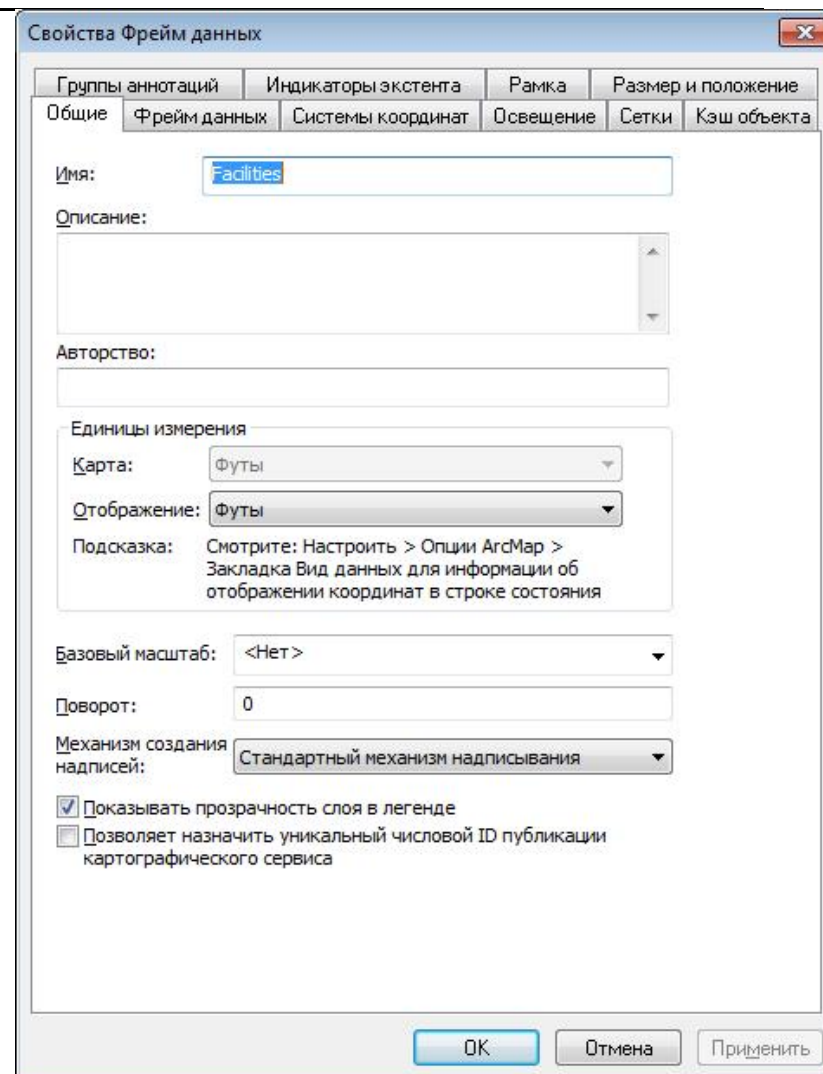
```
mxd = arcpy.mapping.MapDocument("D:/Upload/Project.mxd")  
df = arcpy.mapping.ListDataFrames(mxd)[0].
```

```
import arcpy  
mapdoc = arcpy.mapping.MapDocument("D:/Upload/EsriPress/Python/Data/Exercise10/Austin_TX.mxd")  
listdf = arcpy.mapping.ListDataFrames(mapdoc)  
for df in listdf:  
    print df.name  
del mapdoc
```

print listdf[0].name

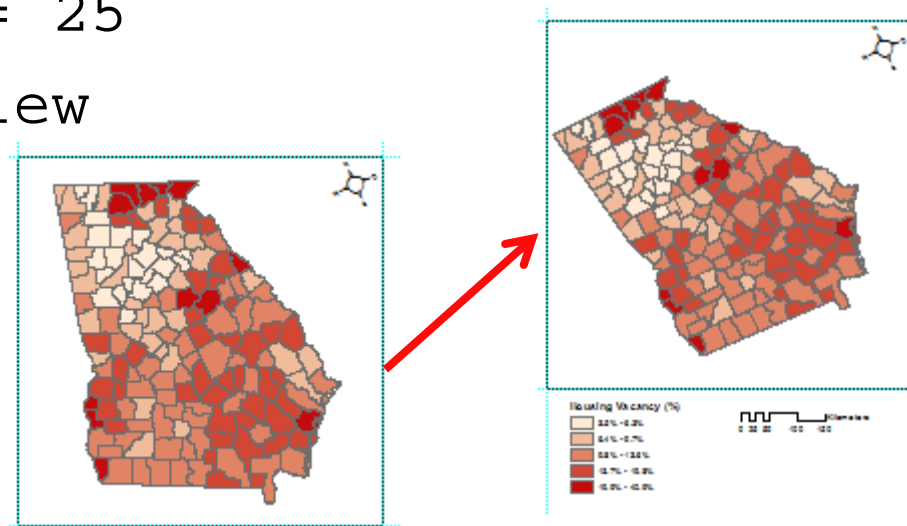
Класс DataFrame. Свойства

- ∅ credits (rw)
 - ∅ description (rw)
 - ∅ displayUnits (rw)
 - ∅ elementHeight (rw)
 - ∅ elementPositionX (rw)
 - ∅ elementPositionY (rw)
 - ∅ elementWidth (rw)
 - ∅ extent (rw)
 - ∅ geographicTransformations (rw)
 - ∅ mapUnits (ro)
 - ∅ name (rw)
 - ∅ referenceScale (rw)
 - ∅ rotation (rw)
 - ∅ scale (rw)
 - ∅ spatialReference (rw)
 - ∅ time (ro)
 - ∅ type (ro)
- ro – чтение
rw – чтение и запись



Класс DataFrame. Свойства

```
import arcpy
mxd = arcpy.mapping.MapDocument("Current")
for df in arcpy.mapping.ListDataFrames(mxd):
    df.scale = 4000000
    df.rotation = 25
arcpy.RefreshActiveView
```



Data Frame. Свойства и методы

СВОЙСТВА

§ credits (rw)
§ description (rw)
§ displayUnits (rw)
§ elementHeight (rw)
§ elementPositionX (rw)
§ elementPositionY (rw)
§ elementWidth (rw)
§ extent (rw)
§ geographicTransformations (rw)
§ mapUnits (ro)
§ name (rw)
§ referenceScale (rw)
§ rotation (rw)
§ scale (rw)
§ spatialReference (rw)
§ time (ro)
§ type (ro)

Методы

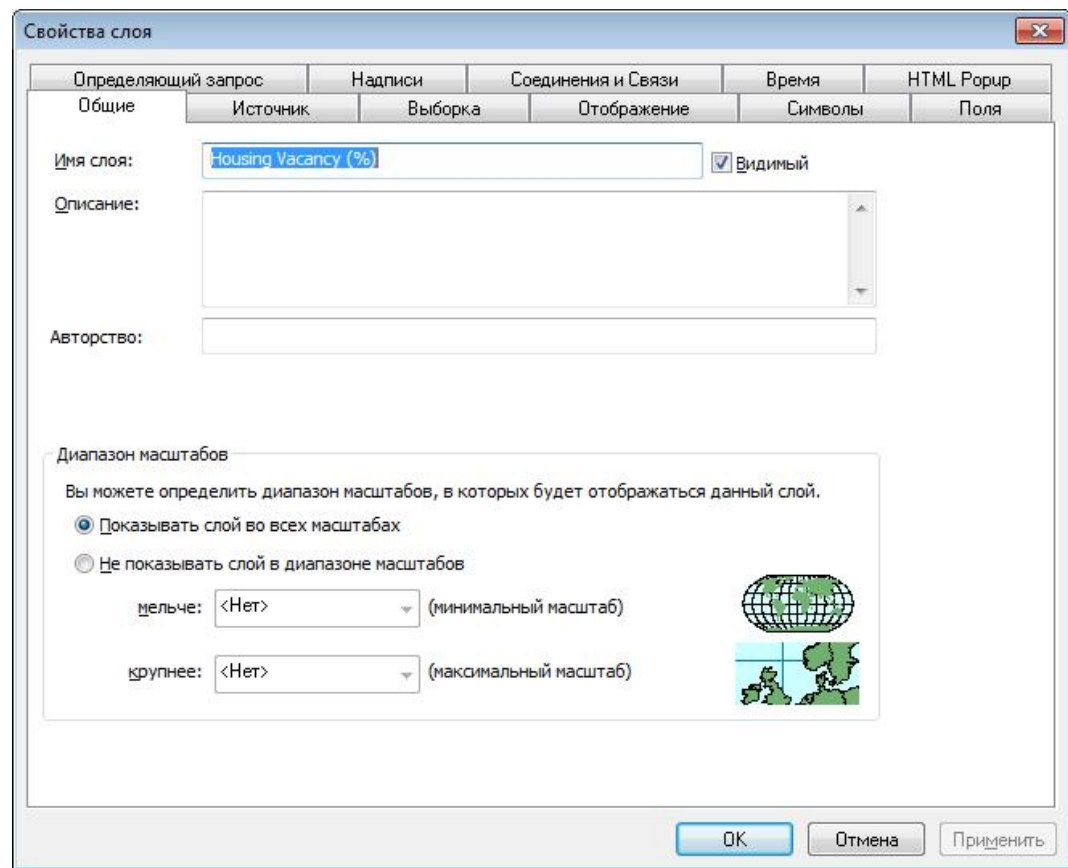
panToExtent (extent)
zoomToSelectedFeatures ()

```
df = arcpy.mapping.ListDataFrames(mxd) [0]
newExtent = df.extent
newExtent.XMin, newExtent.YMin = -180.0, -90.0
newExtent.XMax, newExtent.YMax = 180.0, 90.0
df.extent = newExtent
```

- DATAFRAME_ELEMENT —Элемент Dataframe
- GRAPHIC_ELEMENT —Графический элемент
- LEGEND_ELEMENT —Легенда
- MAPSURROUND_ELEMENT —Элемент Mapsurround
- PICTURE_ELEMENT —Рисунок
- TEXT_ELEMENT —Элемент Текст

Класс Layer

Объект **layer** обеспечивает доступ ко многим свойствам слоя, которые находятся в диалоговом окне **Свойства слоя** в ArcMap, а также предлагает методы для сохранения файлов слоев.





Объект layer

- Функция **Layer**

```
Layer (lyr_file_path)
```

```
lyrFile = arcpy.mapping.Layer(r"D:\Project\Data\Streets.lyr")
```

- Функция **ListLayers**

```
ListLayers (map_document_or_layer,  
            {wildcard}, {data_frame})
```

```
import arcpy  
mxd = arcpy.mapping.MapDocument(r"D:\Project\Project.mxd")  
lyrlist = arcpy.mapping.ListLayers(mxd)[0]  
for lyr in lyrlist:  
    print lyr.name
```

Layer. Свойства и методы

Свойства

- brightness (rw)
- contrast (rw)
- credits (rw)
- datasetName (rw)
- dataSource (rw)
- definitionQuery (rw)
- description (rw)
- isFeatureLayer (ro)
- isGroupLayer (ro)
- isRasterizingLayer (ro)
- isRasterLayer (ro)
- labelClasses (rw)
- longName (ro)
- name (rw)
- serviceProperties (ro)
- showLabels (rw)
- transparency (rw)
- visible (rw)
- workspacePath (ro)


Методы

```
findAndReplaceWorkspacePath(  
    find_workspace_path,  
    replace_workspace_path,  
    {validate})  
  
getExtent(  
    {symbolized_extent})  
  
getSelectedExtent(  
    {symbolized_extent})  
  
replaceDataSource(  
    workspace_path,  
    workspace_type,  
    dataset_name,  
    {validate})  
  
save ()  
  
saveACopy(  
    file_name,  
    {version})  
  
supports (  
    layer_property)
```

Layer =

- .lyr file
- Layer in DataFrame

Selected properties





Метод supports

- Синтаксис

`supports (layer_property)`

```
mxd=arcpy.mapping.MapDocument("CURRENT")
dflist=arcpy.mapping.ListDataFrames(mxd)
lyrlist=arcpy.mapping.ListLayers(mxd, "", dflist[0])
for lyr in lyrlist:
    if lyr.supports("SHOWLABELS")==True:
        lyr.showLabels=True

del lyrlist
```

Управление слоями

Функция	Описание
AddLayer	Добавляет слой к фрейму данных документа карты (.mxd)
AddLayerToGroup	Добавляет слой в составной слой документа карты (.mxd)
InsertLayer	Обеспечивает возможность вставлять слой в нужное местоположение фрейма данных или в составной слой документа карты (.mxd)
MoveLayer	Позволяет передвинуть слой в нужное местоположение во фрейме данных или групповом слое документа карты (.mxd)
RemoveLayer	Обеспечивает возможность удаления слоя фрейма данных в документе карты (.mxd)
UpdateLayer	Позволяет обновлять все свойства слоя или только символы слоя в документе карты (.mxd), извлекая информацию из исходного слоя.



Работа со слоями

- `arcru.mapping` не может
 - Создать новый слой `.lyr`
 - Изменить все свойства слоя в скрипте
- `arcru.mapping` может
 - Добавить слой к фрейму данных и изменить выбранные свойства слоя
 - n `dataSource`, `definitionQuery` и др.
 - Применить символизацию существующих `lyr` файлов к слоям `DataFrame`

Обновление и установка ИСТОЧНИКОВ ДАННЫХ

MapDocument.findAndReplaceWorkspacePaths (find_workspace_path,
replace_workspace_path,
{validate})

MapDocument.replaceWorkspaces (old_workspace_path, old_workspace_type,
new_workspace_path, new_workspace_type,
{validate})

Layer.findAndReplaceWorkspacePath (find_workspace_path,
replace_workspace_path,
{validate})

Layer.replaceDataSource (workspace_path, workspace_type, dataset_name,
{validate})

TableView.findAndReplaceWorkspacePath (find_workspace_path,
replace_workspace_path,
{validate})

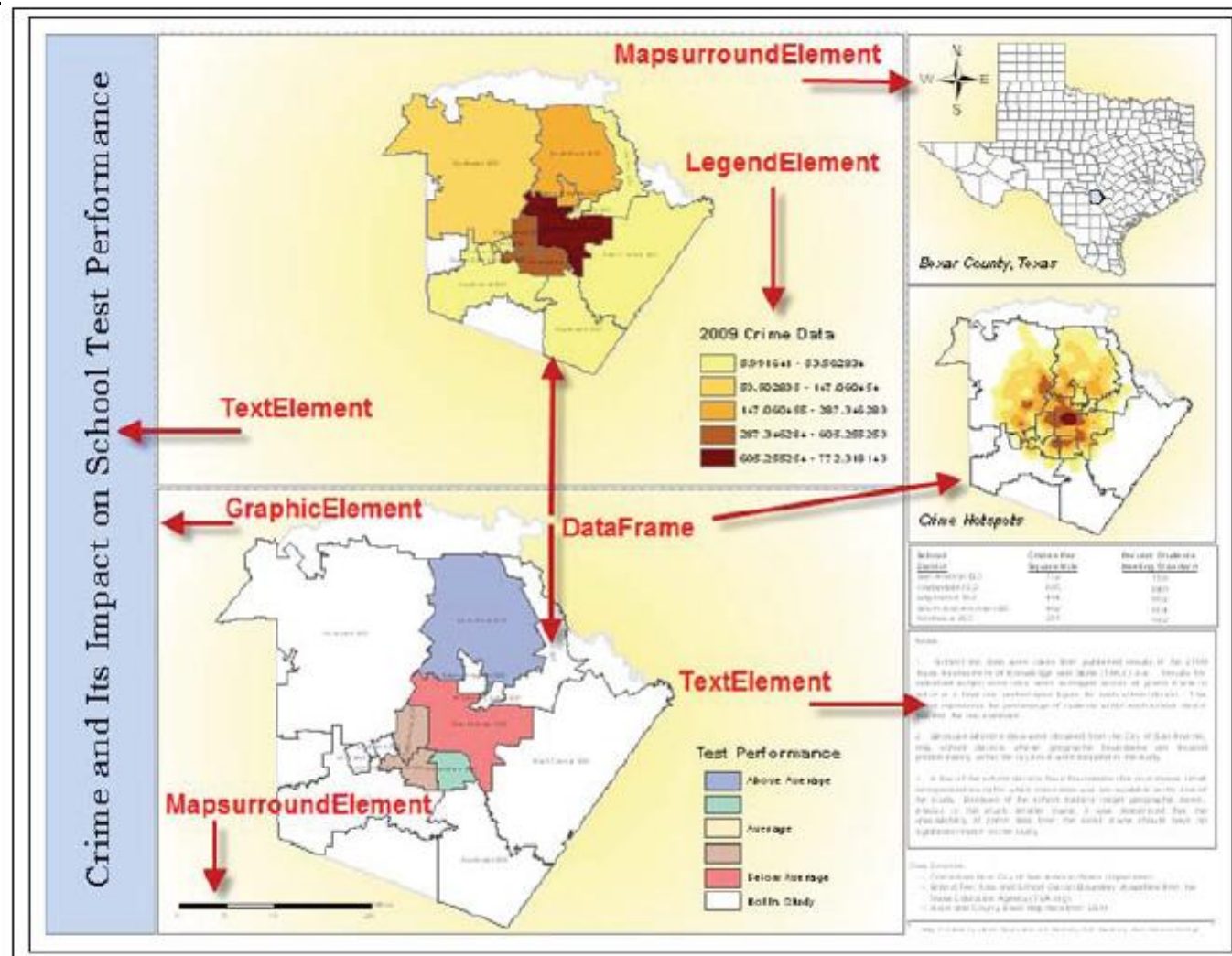
TableView.replaceDataSource (workspace_path, workspace_type, dataset_name,
{validate})

Обновление и установление ИСТОЧНИКОВ ДАННЫХ

```
MapDocument.findAndReplaceWorkspacePaths(find_workspace_path, replace_workspace_path, {validate})
```

```
import arcpy
mapdoc = arcpy.mapping.MapDocument("CURRENT")
mapdoc.findAndReplaceWorkspacePaths("D:\\Upload\\EsriPress\\Python\\Data\\Exercise10\\Austin",
    "D:\\Upload\\Austin")
mapdoc.save()
del mapdoc
```

Работа с элементами компоновки





Работа с элементами компоновки

○ Синтаксис

```
ListLayoutElements (map_document, {element_type},  
{wildcard})
```

element_type

- DATAFRAME_ELEMENT —Элемент Dataframe
- GRAPHIC_ELEMENT —Графический элемент
- LEGEND_ELEMENT —Легенда
- MAPSURROUND_ELEMENT —Элемент Mapsurround
- PICTURE_ELEMENT —Рисунок
- TEXT_ELEMENT —Элемент Текст

ListLayoutElements

```
>>> mxd = arcpy.mapping.MapDocument("CURRENT")
>>> elemlist = arcpy.mapping.ListLayoutElements(mxd)
>>> for elem in elemlist:
...     print elem.name + " " +elem.type
...
Title TEXT_ELEMENT
Stepped Scale Line MAPSURROUND_ELEMENT
North Arrow MAPSURROUND_ELEMENT
Legend LEGEND_ELEMENT
Vacancy DATAFRAME_ELEMENT
```

```
>>> title=arcpy.mapping.ListLayoutElements(mxd)[0]
```

```
>>> title=arcpy.mapping.ListLayoutElements(mxd, "TEXT_ELEMENT")[0]
```

```
>>> title=arcpy.mapping.ListLayoutElements(mxd, "", "Title")[0]
```

Работа с элементами компоновки

LegendElement. Свойства

autoAdd (rw)
elementHeight (rw)
elementPositionX (rw)
elementPositionY (rw)
elementWidth (rw)
isOverflowing (ro)
items (ro)
name (rw)
parentDataFrameName (ro)
title (rw)
type(ro)

TextElement . Свойства

angle (rw)
elementHeight (rw)
elementPositionX (rw)
elementPositionY (rw)
elementWidth (rw)
fontSize (rw)
name (rw)
text(rw)
title (rw)

```
import arcpy
mxd = "D:/Upload/EsriPress/Python/Data/Exercise10/Georgia.mxd"
mapdoc = arcpy.mapping.MapDocument(mxd)
elemlist = arcpy.mapping.ListLayoutElements(mapdoc)
title = elemlist[0]
title.text = "Housing Vacancy for Georgia Counties (2000)"
mapdoc.save()
del mapdoc
```

Экспорт и печать карт

○ ФУНКЦИИ

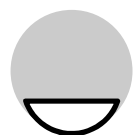
ExportReport
ExportToAI
ExportToBMP
ExportToEMF
ExportToEPS
ExportToGIF
ExportToJPEG
ExportToPDF
ExportToPNG
ExportToSVG
ExportToTIFF
ListPrinterNames
PrintMap

Синтаксис

```
ExportToJPEG (map_document, out_jpeg,  
{data_frame}, {df_export_width},  
{df_export_height}, {resolution},  
{world_file}, {color_mode},  
{jpeg_quality}, {progressive})
```

```
import arcpy  
mxd =  
arcpy.mapping.MapDocument(r"D:\Project\Project.mxd")  
arcpy.mapping.ExportToJPEG(mxd,  
r"D:\Project\Output\Project.jpg", resolution=300)  
del mxd
```

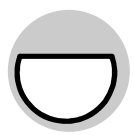
Работа с pdf



ФУНКЦИИ

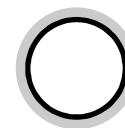
PDFDocumentOpen

PDFDocumentCreate



Класс

PDFDocument (pdf_path)



Методы

appendPages

attachFile

deletePages

insertPages

saveAndClose

updateDocProperties

updateDocSecurity

```
import arcpy
pdfPath = r"D:\Upload\AtlasMapBook.pdf"
pdfDoc = arcpy.mapping.PDFDocumentCreate(pdfPath)
pdfDoc.appendPages(r"D:\Upload\Title.pdf")
pdfDoc.appendPages(r"C:\Upload\Atlas.pdf")
pdfDoc.appendPages(r"C:\Upload>ContactInfo.pdf")
pdfDoc.saveAndClose()
del pdfDoc
```