



Arcpy und python 2.6

Tipps und Tricks und Neuigkeiten

Bundesamt für Kartographie und Geodäsie



■ Umgebungseinstellung

ArcGIS 9.3

```
import arcgisscripting
gp = arcgisscripting.create(9.3)
gp.workspace = r"C:\myData\daten.gdb"
```

ArcGIS 10

```
import arcpy
arcpy.env.workspace = r"C:\myData\daten.gdb"
arcpy.env.outputCoordinateSystem = ...
arcpy.env.extent = ...
arcpy.env.overwriteOutput = True
```



■ Cursor

```
# ArcGIS 9.3
cursor = gp.SearchCursor(table)
rows = cursor.next()
while row:
    print row.ObjectID
    row = rows.Next()
del cursor, row
```

```
# ArcGIS 10
cursor = gp.SearchCursor(table)
for row in cursor:
    print row.ObjectID
del cursor, row
```



■ Geometry List

- Einfache python-Liste mit Geometry-Objekten
- In einigen Tools wie eine Featureclass verwendbar
- Schnellere geoprocessing-Verarbeitung möglich, weil nur im Hauptspeicher ausgeführt

```
list = arcpy.CopyFeatures_management(inputFC,arcpy.Geometry())  
print len(list)  
arcpy.Buffer_analysis(list, "C:/buff.shp", "100 Meters")
```

```
laenge = 0  
for geometry in list:  
    laenge += geometry.length  
print "Gesamtlaenge: %f " % laenge
```



■ Mapping Modul

- Abfragen und Verändern von Layern

```
lyrs = arcpy.mapping.ListLayers(mxd)
```

- Drucken und Exportieren der Karte

```
mxd = arcpy.mapping.MapDocument(r'C:\MyData\MyMap.mxd')  
arcpy.mapping.ExportToPDF(mxd, r'C:\MyData\MyMap.pdf')
```

- Repair Broken Datasources

```
for lyr in arcpy.mapping.ListBrokenDataSources(mxd):  
    name = lyr.name  
    lyr.replaceDataSource(r'C:\MyData\Data.gdb',  
        'FILEGDB_WORKSPACE', name)
```

- Wechsel des Koordinatensystems des Datenrahmens

```
df = arcpy.mapping.ListDataFrames(mxd)[0]  
sr = arcpy.SpatialReference(prjFile)  
df.spatialReference = sr
```