

Использование ArcGIS API for Python в Jupyter Notebook

1. Простое использование arcgis

- arcgis.gis** обеспечивает базовую информационную модель для ArcGIS. Этот **модуль** обеспечивает функции управления контентом.
- Класс **GIS** в модуле gis - самый главный в ArcGIS API for Python, это наша входная точка. Он позволяет работать с геоконтентом.

```
In [1]: import arcgis
from arcgis.gis import GIS
```

```
In [2]: # Подключиться к ArcGIS с помощью API-key
gis = GIS(api_key="AAPKdcb853a48b4744fea1099ec94bf300513KrnAv5kCw6pbg-XkH-91doph-i5-Flwn6U6AA7Pj1FY1KI-7dTdmn1L
```

```
In [3]: # Для любого объекта можно вызвать справку
gis?
```

- Чтобы встроить карту нужно вызвать метод `gis.map()`
- На карте работают базовые элементы: перетаскивание мышью и изменение масштаба.

```
In [3]: # Встроить карту
map1 = gis.map()
map1

MapView(layout=Layout(height='400px', width='100%'))
```

```
In [4]: map2 = gis.map("Palm Springs, CA")
map2

MapView(layout=Layout(height='400px', width='100%'))
```

```
In [5]: moscow = gis.map("Москва")
moscow

MapView(layout=Layout(height='400px', width='100%'))
```

- Последующие манипуляции с объектом `moscow` отразятся на **web-карте** во **фрейме** выше

```
In [7]: # Можно управлять зумом карты
moscow.zoom = 13
```

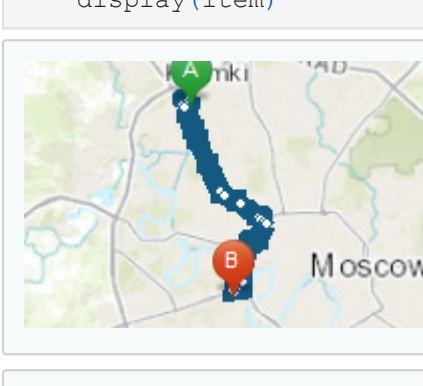
```
In [8]: # Посмотрим список доступных базовых карт
moscow.basemaps
```


```
Out[8]: ['dark-gray',
'dark-gray-vector',
'gray',
'gray-vector',
'hybrid',
'national-geographic',
'oceans',
'osm',
'satellite',
'streets',
'streets-navigation-vector',
'streets-night-vector',
'streets-relief-vector',
'streets-vector',
'terrain',
'topo',
'topo-vector']
```


```
In [9]: # Посмотрим все базовые карты во фрейме
from time import sleep
for basemap in moscow.basemaps:
    moscow.basemap = basemap
    sleep(5)
```


```
In [10]: # Установим конкретную базовую карту
moscow.basemap = 'osm'
```


```
In [11]: # На карту можно добавлять контент из галереи ArcGIS Online
from IPython.display import display
items = gis.content.search('title:Moscow', item_type='Feature Collection')
for item in items:
    display(item)
```




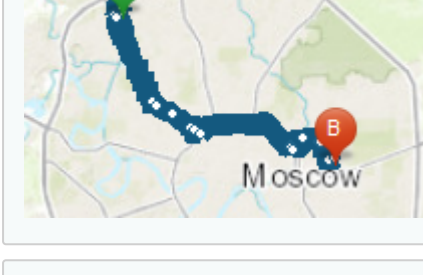
Moscow route rechts
Маршрут и путевой лист для Moscow_route_rechts Feature Collection by OSerebryannaya_i18n
Last Modified: Октябрь 18, 2016
0 comments, 135 views




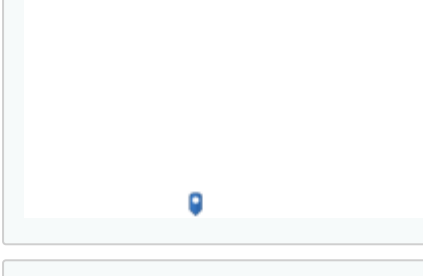
Moscow Folklore Map
Фольклорная карта Москвы, разработчик - Nikita Petrov Feature Collection by nivirov
Last Modified: Май 21, 2015
0 comments, 221 views




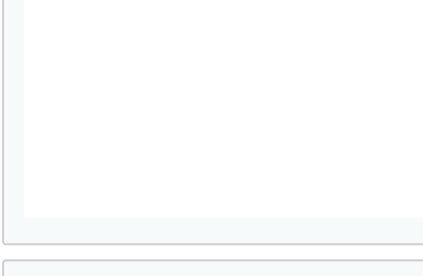
moscow.dtp
 Feature Collection by amaksimov
Last Modified: Ноябрь 16, 2021
0 comments, 2 views




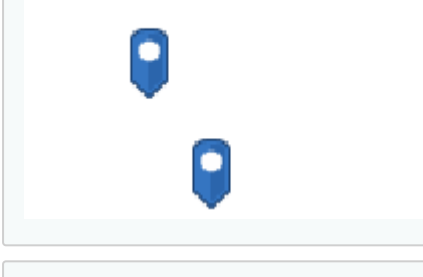
Moscow route
Route en routebeschrijving voor Москва, Смоленная улица 52, RUS — Москва, 8-я Соколиной Горы улица 18, RUS Feature Collection by OSerebryannaya_i18n
Last Modified: Октябрь 18, 2016
0 comments, 113 views





Moscow
 Feature Collection by rokobylas
Last Modified: Март 11, 2022
0 comments, 7 views




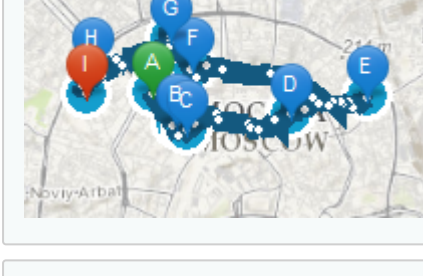
Moscow-Sightseeing
 Feature Collection by FrancoBoni4
Last Modified: Апрель 29, 2018
0 comments, 117 views




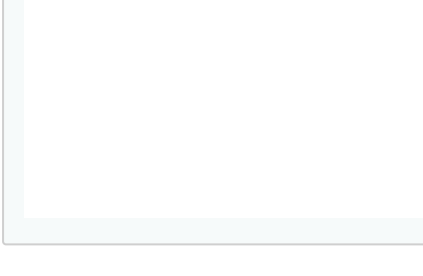
Alphabet of Taste - Moscow
 Feature Collection by richard.trimmer_KumonUK_I
Last Modified: Январь 14, 2020
0 comments, 325 views



Moscow, ID
 Feature Collection by OOHAWilkinsMedia
Last Modified: Март 17, 2021
0 comments, 11 views

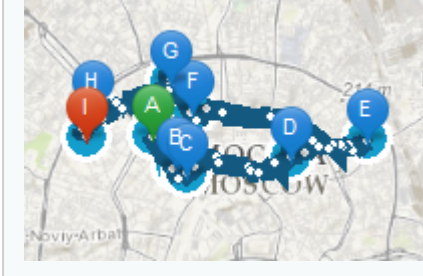



Moscow Walking Tour
Route and directions for Moscow Walking Tour Feature Collection by thomaskoblet
Last Modified: Август 13, 2016
0 comments, 233 views



Moscow-Accommodations
 Feature Collection by FrancoBoni4
Last Modified: Апрель 29, 2018
0 comments, 80 views

```
In [12]: # Moscow Walking Tour - 8-ой элемент
my_layer = items[8]
display(my_layer)
```



Moscow Walking Tour
Route and directions for Moscow Walking Tour Feature Collection by thomaskoblet
Last Modified: Август 13, 2016
0 comments, 233 views

```
In [13]: # Добавляем сервис к нашему объекту moscow и смотрим карту
moscow.add_layer(my_layer)
```

2. Работа с объектами

- Поработаем с **набором данных**, используя **pandas** и **arcgis**
- Сначала импортируем csv в pandas, а потом преобразуем его в **слой карты**
- В качестве примера возьмём слой паркоматов Москвы

```
In [14]: import pandas as pd
parkomats = pd.read_csv(r'.\data\data-417-2017-02-14.csv', sep=';')
```

```
In [15]: # Посмотрим получившийся объект
parkomats
```

```
Out[15]:
```

	ID	NumberOfParkingMeter	global_id	AdmArea	District	StreetName	Location	longitude	latitude	ParkingZoneNumber
0	71	Паркомат № 2071	1774401	Северный административный округ	район Аэропорт	улица Авиаконструктора Яковлева	улица Авиаконструктора Яковлева, дом 27, корпус 2	37.519896	55.806302	
1	72	Паркомат № 2072	1774402	Центральный административный округ	Пресненский район	улица Пресненский Вал	улица Пресненский Вал, дом 42	37.571246	55.771818	
2	74	Паркомат № 2074	1774404	Восточный административный округ	район Перово	Электродная улица	Электродная улица, дом 6Б	37.752553	55.754657	
3	75	Паркомат № 2075	1774405	Восточный административный округ	район Перово	улица Плеханова	улица Плеханова, дом 5	37.758299	55.756431	
4	81	Паркомат № 2081	1774411	Восточный административный округ	район Соколиная Гора	Гаражная улица	Гаражная улица, дом 4	37.740537	55.758795	
...
870	1990	Паркомат № 3287	282752273	Юго-Восточный административный округ	район Люблино	Совхозная улица	Совхозная улица, дом 45	37.761807	55.675818	
871	1991	Паркомат № 3397	282752274	Восточный административный округ	район Сокольники	5-я Сокольническая улица	5-я Сокольническая улица, дом 1	37.688124	55.785373	
872	1972	Паркомат № 2240	282752301	Западный административный округ	район Фили-Давыдовское	улица Герасима Курина	улица Герасима Курина, дом 16	37.472284	55.731046	
873	1980	Паркомат № 2345	282752302	Западный административный округ	район Раменки	Университетский проспект	Университетский проспект, дом 13	37.543410	55.701725	
874	1963	Паркомат № 2031	282920545	Северо-Западный административный округ	район Южное Тушино	Сходненская улица	Сходненская улица, дом 50	37.440234	55.847444	

875 rows × 12 columns

- Теперь импортируем DataFrame из pandas в объект gis
- Для этого просто вызываем функцию `import_data()`
- В результате у нас получится **объект FeatureCollection**

```
In [16]: geoparkomats = gis.content.import_data(parkomats)
geoparkomats
```

```
Out[16]: <FeatureCollection>
```

- Теперь добавим **объекты** на карту
- Для этого инициализируем новый объект map, и в него добавим наш новый **слой**

```
In [17]: parkomat_map = gis.map('Москва')
parkomat_map.zoom = 12
parkomat_map.basemap = 'osm'
parkomat_map.add_layer(geoparkomats)
parkomat_map

MapView(layout=Layout(height='400px', width='100%'))
```

Теперь попробуем эти данные оформить. Мы можем добавить и стилизовать слой. Например:

```
In [18]: # Разукрасим данные по зонам парковки
parkomat_cat_map = gis.map('Москва')
parkomat_cat_map.zoom = 10
parkomat_cat_map.basemap = 'gray'
parkomat_cat_map.add_layer(geoparkomats, {"renderer": "ClassedColorRenderer", "field_name": "ParkingZoneNumber"})
parkomat_cat_map

MapView(layout=Layout(height='400px', width='100%'))
```

```
In [19]: # Теперь попробуем сделать тепловую карту
parkomat_heat_map = gis.map('Москва')
parkomat_heat_map.zoom = 11
parkomat_heat_map.basemap = 'dark-gray'
parkomat_heat_map.add_layer(geoparkomats, {"renderer": "HeatmapRenderer"})
parkomat_heat_map

MapView(layout=Layout(height='400px', width='100%'))
```

```
In [ ]:
```