



جزوه شماره ۱۷



# FOLIUM

Visualizing Geospatial Data with Python



ناهید نعمتی کوتنائی (تیسا)  
دکتری جغرافیا و برنامه‌ریزی شهری  
مدرس دانشگاه

 **Dr.nemati.K**  
 **@Nemati\_k**  
 ۰۹۱۱۲۲۳۰۷۹۸



محمدطاهر طاهرپور  
دانشجوی ارشد مدیریت شهری  
دانشگاه تهران

 **mttahrpoor**  
 **@mtahrpoor**  
 ۰۹۳۳۶۱۴۴۹۴۷



زمستان ۱۴۰۲

## فهرست مطالب:

۳	..... folium
۳	..... Colab چیه؟
۴	..... نحوه دسترسی به Colab
۵	..... folium
۵	..... ایجاد یه نقشه ساده
۶	..... اضافه کردن Control Layer به نقشه
۸	..... اضافه کردن Shapefile به نقشه
۹	..... ذخیره notebook به فرمت ipynb و py
۱۰	..... ذخیره نقشه در یک فایل HTML
۱۰	..... اضافه کردن دایره روی نقشه
۱۲	..... نحوه مارکر گذاشتن روی نقشه
۱۵	..... ساخت نقشه از اکسل
۱۶	..... ساخت نقشه heatmap با folium
۱۷	..... پلاگینها در فولیوم
۱۸	..... نکات تکمیلی در مورد Colab

## folium

تو این جزوه یکی از کتابخانه‌های مهم پایتون به اسم Folium که واسه تحلیل و مصورسازی داده‌های جغرافیایی استفاده میشه رو با هم یاد میگیریم. اینبار برای نوشتن کدها از سرویس ابری و رایگان گوگل به اسم Colab استفاده کردیم که ببینیم چطوری میشه کدهای پایتون رو روی محیطی مبتنی بر مرورگر نوشت. واسه نوشتن جزوه از منابع زیر استفاده کردیم:

۱. مجموعه آموزشی Folium Interactive Maps in python موجود در یوتیوب با لینک

<https://www.youtube.com/playlist?list=PLCgehTvigkDNzlGnvmLqPxUjDyhMGwm-A>

۲. سایت <https://nagasudhir.blogspot.com/introduction-to-folium-for-interactive.html>

۳. سایت <https://python-visualization.github.io/folium/> برای آشنایی بیشتر با قابلیت‌ها و استفاده از

folium که مستندات شامل نمونه‌ها، مثال‌ها و راهنمایی‌های کاملی در مورد استفاده از این کتابخانه برای تولید نقشه‌های تعاملی داره.

ابهاماتی که تو این سایتها داشتیم رو هم از هوش مصنوعی poe پرسیدیم و برامون رفعش کرد:

<https://poe.com/>

اول سرویس ابری کولب رو معرفی میکنیم و بعد میریم سراغ کار با داده‌های جغرافیایی با کتابخانه folium.

## Colab چیه؟

Google Colab (کولب) یک سرویس ابری رایگان هست که گوگل ارائه داده و بهمون کمک میکنه که کدهای پایتون رو در محیطی مبتنی بر مرورگر اجرا کنیم. این سرویس برای توسعه و اجرای پروژه‌های یادگیری ماشین، دیتا ساینس و تحقیقات علمی خیلی کاربردی هست. امکانات Google Colab شامل موارد زیر میشه:

✚ **محیط اجرایی پایتون:** Google Colab یک محیط اجرایی پایتون کامل رو فراهم میکنه که شامل بسته‌های محبوب مثل numpy، pandas و tensorflow هست. میتونیم کدهای پایتون خودمون رو در سلولهای نوتبوک اجرا کنیم و نتایج رو ببینیم.

✚ **پشتیبانی از GPU و TPU:** یکی از مزیت‌های بزرگ Google Colab، ارائه منابع پردازشی قوی هست. با استفاده از Colab، میتونیم از GPU و TPU رایگان برای اجرای الگوریتم‌های یادگیری ماشینی و پردازش‌های سنگین استفاده کنیم.

✚ **اشتراک گذاری آسون:** میتونیم نوتبوک‌های خودمون رو با دیگران به راحتی به اشتراک بگذاریم. از طریق ارسال لینک نوتبوک، بقیه میتونن کدهای ما رو مشاهده و اجرا کنن.

✚ **نصب آسون پکیج‌ها:** با استفاده از دستورات pip و apt، میتونیم به آسونی پکیج‌های مورد نیاز خودمون رو نصب کنیم و از اونها در کدهای پایتون خودمون استفاده کنیم.

✚ **انعطاف‌پذیری در ذخیره و بازیابی داده:** میتونیم داده‌هامون رو در Google Drive ذخیره کنیم و به اونها از طریق Colab دسترسی پیدا کنیم. علاوه بر اون میتونیم از GitHub برای ذخیره و بازیابی نوتبوک‌ها استفاده کنیم.

✚ **پشتیبانی از Markdown در Colab:** Markdown بهمون این امکان رو میده که متن‌های توضیحی و توضیحات مربوط به کدهای خودمون رو با استفاده از Markdown فرمت کنیم و نوتبوک‌های خودمون رو توسط متن‌های توضیحی توصیف کنیم.

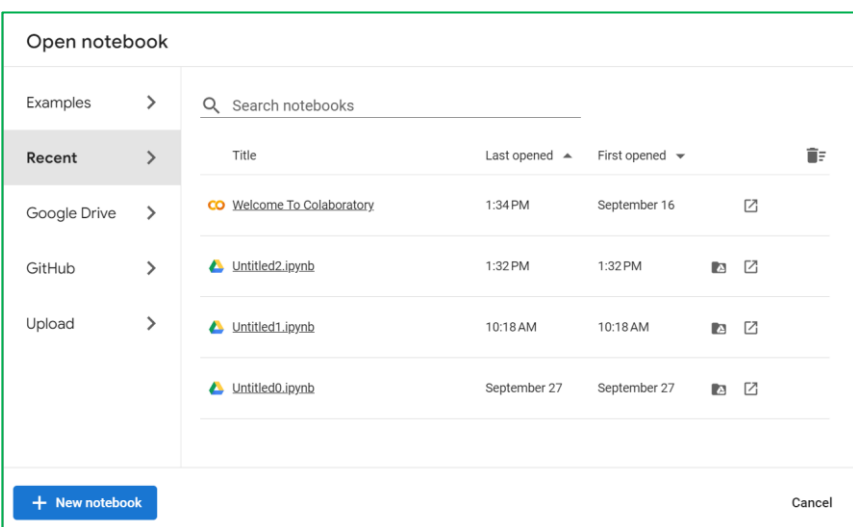
🚩 **Ctrl + Space**: در Jupyter همیشه پیشنهادهای پایتون رو با زدن گزینه Tab دید، ولی در Colab به جای Tab باید از Ctrl + Space استفاده کنی.

## نحوه دسترسی به Colab

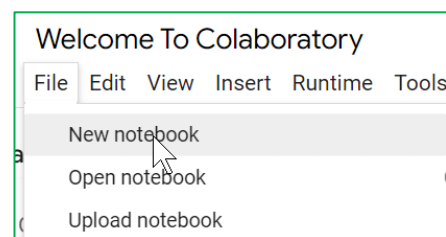
برای دسترسی به Google Colab، مراحل زیر را دنبال کن:  
۱. باز کردن مرورگر و ورود به سایت Google Colab با آدرس زیر:

<https://colab.research.google.com>

۲. وارد شدن به حساب Google خودت با استفاده از اطلاعات ورودی مربوطه. اگر حساب Google نداری، می‌توانی یک حساب جدید بسازی.



۳. بعد از ورود به Google Colab، یه صفحه خالی با یک نوت‌بوک جدید میبینی. این نوت‌بوک محیطی هست که می‌تونی در اون کدهای پایتون رو اجرا کنی. برای شروع استفاده از Colab، همیشه یک نوت‌بوک جدید ساخت یا از نوت‌بوک‌های موجود در بخش "Examples" استفاده کرد.



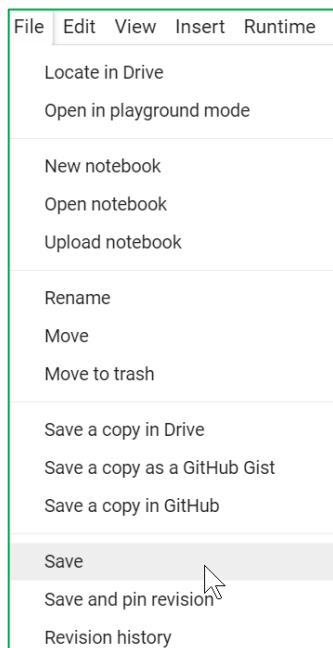
۴- اگه این صفحه بسته شد میتونی از file یه notebook جدید بیاری.

۵. در صورتی که می‌خواهی کدی را اجرا کنی که نیاز به پکیج‌های خاصی داره، می‌تونی اونها رو با استفاده از دستورات pip یا apt نصب کنی. به طور کلی، Colab پکیج‌های معمول رو پیش نصب داره. عبارت `import folium` رو وارد کن ببین نصب داره یا نه. اگه نصب نبود از دستور زیر واسه نصبش استفاده کن و بعد با دستور `import folium` واردش کن.

`!pip install folium`

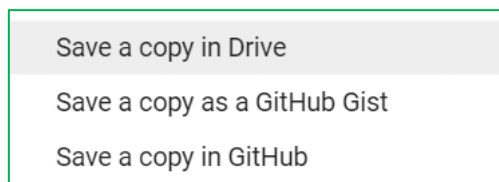
۶. برای اجرای کد، می‌تونی هر سلول رو به صورت جداگانه اجرا کنی. سلولها می‌تونن شامل کد پایتون، متن به صورت Markdown و یا دستورات سیستمی باشن.





۷. برای ذخیره نوت‌بوک، می‌توننی از گزینه "File" در نوار منوی بالا استفاده کنی و بعدش "Save" رو انتخاب کنی.

۸. می‌توننی نوت‌بوک رو به GitHub یا Google Drive خودت ارسال کنی و یا روی سیستم خودت اون رو ذخیره کنی.



## folium

folium یک کتابخانه متن‌باز یا open source در زبان پایتون هست که برای تولید نقشه‌های تعاملی و قابل تعامل استفاده می‌شه. این کتابخانه بر پایه کتابخانه Leaflet جاوا اسکریپت توسعه پیدا کرده و امکانات پیشرفته‌تری رو در کار با نقشه‌ها در پایتون فراهم می‌کنه. استفاده از folium خیلی ساده و قابل فهم هست. با استفاده از این کتابخانه، می‌توننی نقشه‌های تعاملی با امکانات مختلفی مثل نمایش نقاط، خطوط، چندضلعی‌ها و عوارض دیگه رو ایجاد کنی. علاوه بر اون می‌توننی نقشه‌ها رو با لایه‌های مختلفی مثل تصاویر ماهواره‌ای، ترسیمات و برجسب‌ها تزئین کنی.

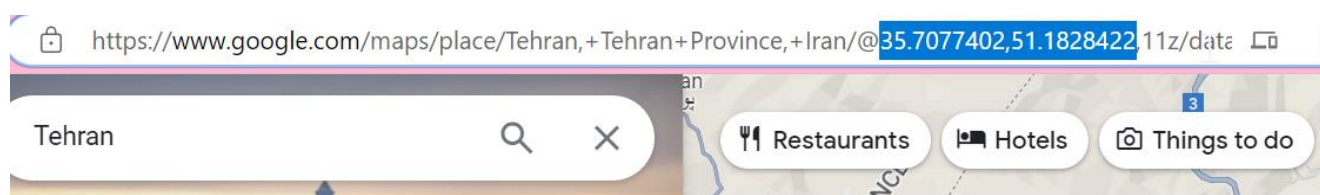
## ایجاد یه نقشه ساده

برای ایجاد یک نقشه ساده، نیاز به تعیین مرکز نقشه و سطح بزرگنمایی اون داری. بعدش می‌توننی عناصر مختلفی مثل نقاط یا خطوط رو به نقشه اضافه کنی. ولی حواست باشه که در Colab، نقشه‌هایی که با استفاده از folium ایجاد می‌کنی، به صورت تعاملی در خروجی نمایش داده نمی‌شه. به جای اون، برای مشاهده نقشه‌ها در Colab، باید از تابع display از کتابخانه IPython.display استفاده کنی تا نقشه را درون یک فریم باز کنی.

```
import folium
```

```
from IPython.display import display
```

حالا با folium.Map() یه نقشه درست کن و مرکز و بزرگنمایی رو بهش بده. برای اینکه موقعیت یه نقطه رو بهش بدی میتونی از Google Maps استفاده کنی. تو قسمت سرچ Google Maps بنویس Tehran و بعد از لینک بالای صفحه X و Y که داده رو بعد از @ کپی کن.



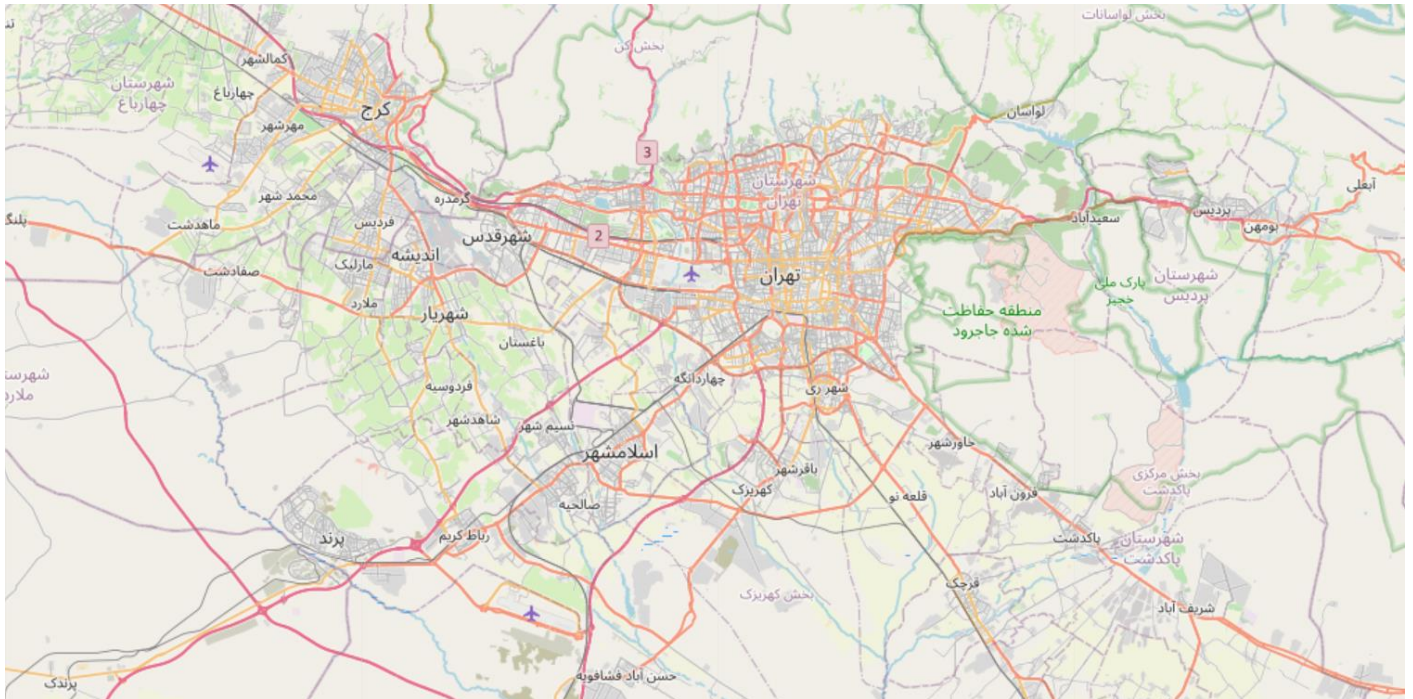
۳۵,۷۰۷۷۴۰۲, ۵۱,۱۸۲۸۴۱

این اعداد طول و عرض جغرافیایی رو باید بذاری توی براکت [] = location. اعداد رو تو کد زیر وارد کن. کل کد رو به به متغیر به اسم mapObj اختصاص بده که بعدا بتونی ازش استفاده کنی. zoom\_start هم تعیین میکنه که بزرگنمایی نقشه چقدر باشه.

```
mapObj = folium.Map(location=[۳۵,۷۰۷۷۴۰۲,۵۱,۱۸۲۸۴۱], zoom_start=۱۰)
```

بعد با دستور display نمایشش بده.

```
display(mapObj)
```



## اضافه کردن Control Layer به نقشه

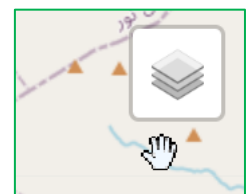
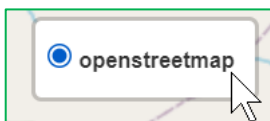
میخوایم یه بخش کنترل لایه روی نقشه بندازیم و بعد یه سری لایه به نقشه اضافه کنیم. دستور زیر رو بنویس و اجرا کن:

```
folium.LayerControl().add_to(mapObj)
```

```
display(mapObj)
```

میبینی که control layer به نقشه اضافه میشه. وقتی موس رو روش

ببری گزینه openstreetmap دیده میشه.



میتونی به جای openstreetmap تصاویر ماهواره دیگه‌ای زیر کارت بندازی. برای اینکار میتونی از

دستور folium.TileLayer() استفاده کنی. مثل:

- ✚ Mapbox Bright: 'Mapbox Bright'
- ✚ Mapbox Control Room: 'Mapbox Control Room'
- ✚ Stamen Terrain: 'Stamen Terrain'
- ✚ Stamen Toner: 'Stamen Toner'
- ✚ Stamen Watercolor: 'Stamen Watercolor'
- ✚ CartoDB Positron: 'CartoDB Positron'
- ✚ CartoDB Dark Matter: 'CartoDB Dark Matter'



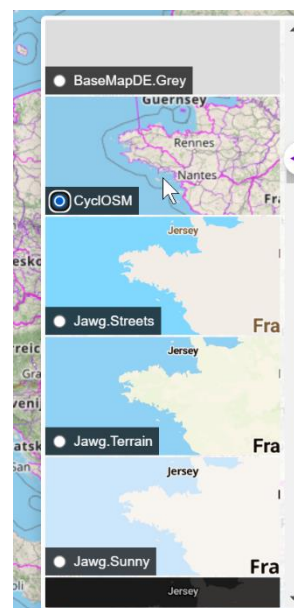
```
tile_layer = folium.TileLayer('CartoDB Positron', attr='CartoDB Positron').add_to(mapObj)
```



میتونی به کمک url هم به نقشه‌ها تایل اضافه کنی.

<https://leaflet-extras.github.io/leaflet-providers/preview/>

از سمت راست روی هر tile که دوست داری کلیک و آدرسی که بهت  
میده رو کپی کن.

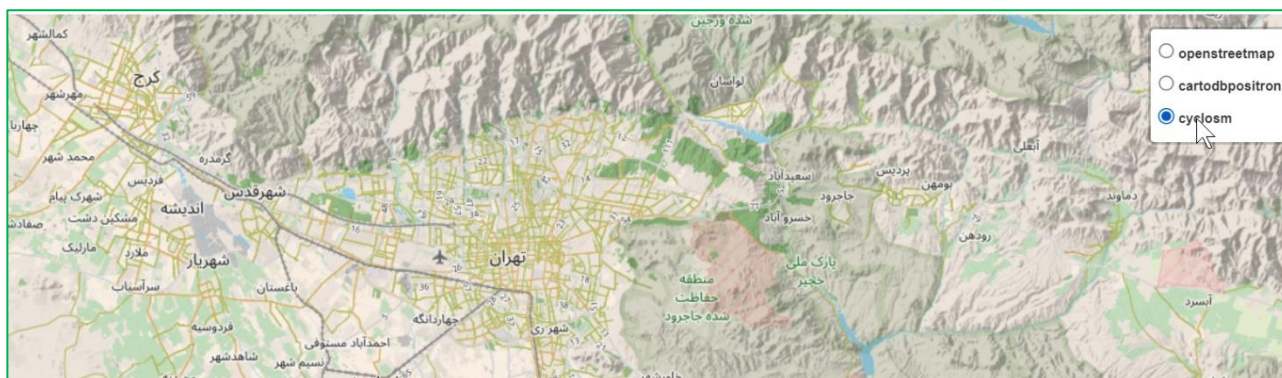


تو کد زیر بذارش. میتونی بهش یه اسم هم کنار ویژگی اختصاص بدی.

```
tile_layer = folium.TileLayer('https://{s}.tile-cyclosm.openstreetmap.fr/cyclosm/{z}/{x}/{y}.png', name =  
'cyclosm', attr='cyclosm').add_to(mapObj)
```

```
mapObj = folium.Map(location=[35.7077402, 51.182841], zoom_start=10)  
tile_layer = folium.TileLayer('CartoDB Positron', attr='CartoDB Positron').add_to(mapObj)  
tile_layer = folium.TileLayer('https://{s}.tile-cyclosm.openstreetmap.fr/cyclosm/{z}/{x}/{y}.png', name = 'cyclosm', attr='cyclosm').add_to(mapObj)  
folium.LayerControl().add_to(mapObj)  
display(mapObj)
```

حالا سه تا tile داری.



## اضافه کردن Shapefile به نقشه

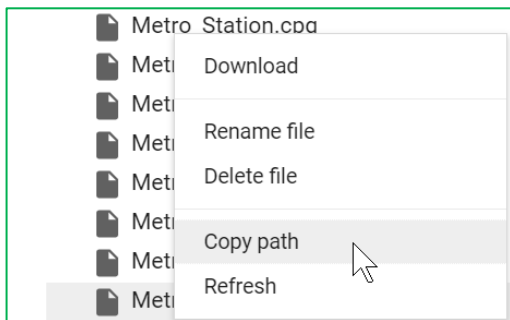
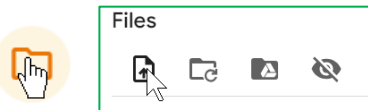
کتابخانه mapclassify رو باید با pip! به شیوه زیر نصب کنی که بتونی shapefile به محیط کاریت اضافه کنی.

```
!pip install folium matplotlib mapclassify
```

حالا از کتابخانه mapclassify ماژول classify رو وارد کن.

```
import folium
from IPython.display import display
import geopandas as gpd
import matplotlib.pyplot as plt
from mapclassify import classify
```

روی پوشه نارنجی بزن و برو روی گزینه file برو و shapefiles رو بیار تو محیط کاری.



برای آوردن shp ها توی خط کد، باید مسیرش رو بدونی. روی بخشی که پسوند shp. داره کلیک راست کن و copy path رو بزن. بعد مسیرش رو توی کد زیر با gpd.read\_file() وارد کن.

```
Station = gpd.read_file('/content/Metro_Station.shx')
```

با دستور explore() میتونی لایه رو وبی کنی.

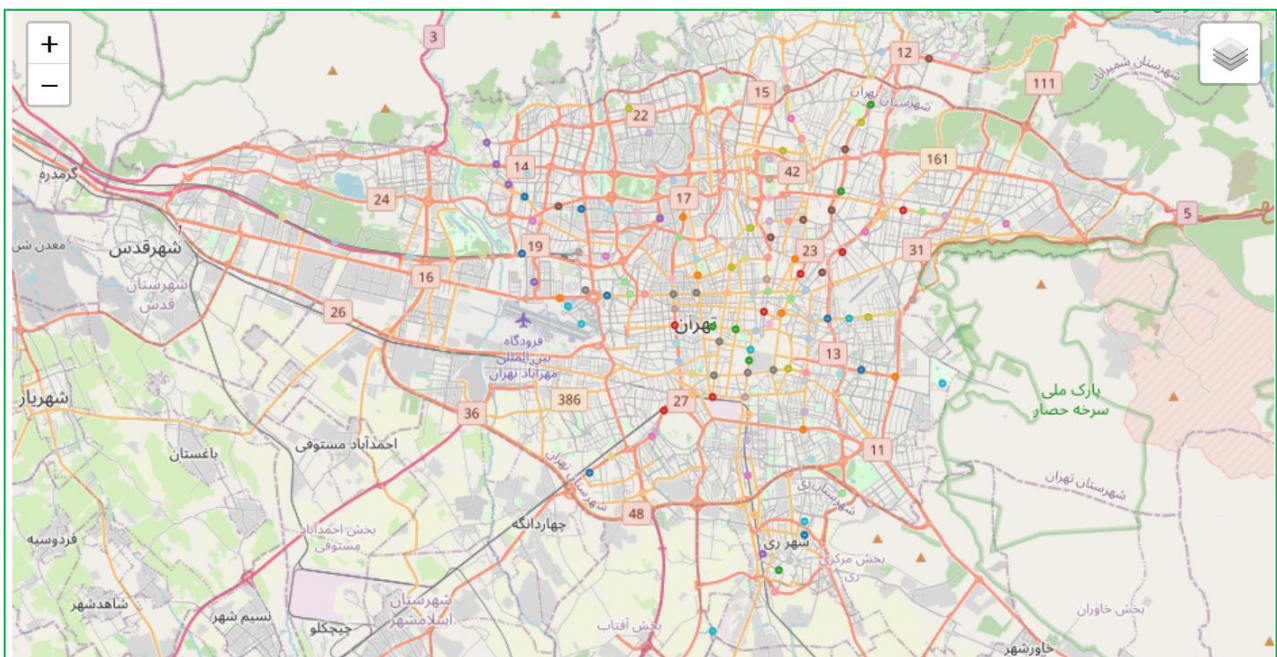
```
m = Station.explore("Name", legend=False)
```

با دستور folium.LayerControl.addTo() یه کنترل لایه به نقشه اضافه کن.

```
folium.LayerControl().addTo(m)
```

با دستور display() نمایشش بده.

```
display(m)
```





حالا اگه بخوای چندتا لایه رو با هم بیاری مثل بالا اول مسیر هر کدوم رو کپی کن و با دستور `gpd.read_file()` بخونشون.

```
Station = gpd.read_file('/content/Metro_Station.shx')
```

```
Metro = gpd.read_file('/content/Metro_Line.shp')
```

```
Mahale = gpd.read_file('/content/Mahalat.shp')
```

با دستور `explore()` یکی از لایه‌ها رو وبی کن و به یه اسم اختصاصش بده مثلا `m`.

```
m = Station.explore("Name", legend=False)
```

لایه‌های بعدی هم باید به همین اسم اختصاص داده بشن و تو پرانتز `explore()` باید بگیم که `m=m` هست که همه‌شون روی هم بیفتن.

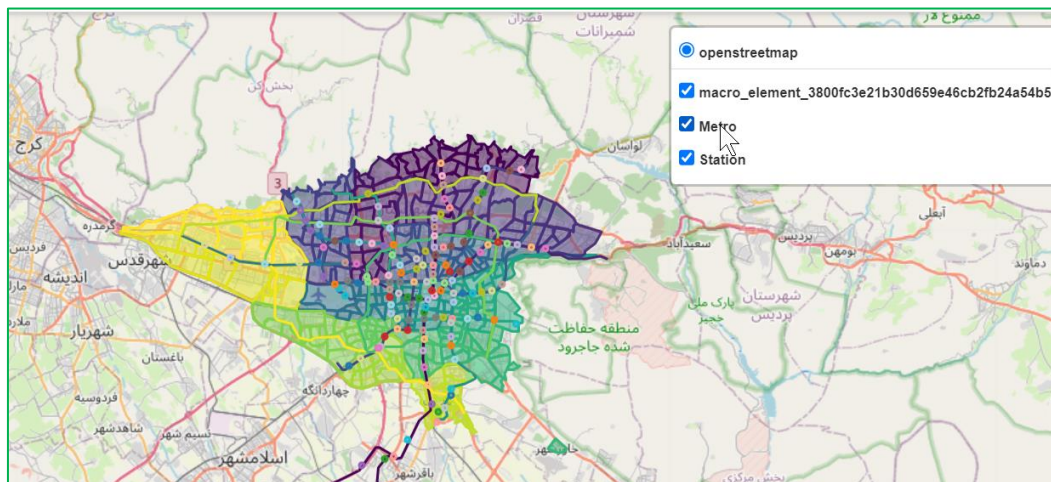
```
m = Metro.explore(m=m, name = "Metro", column= "Line", legend=False)
```

```
m = Mahale.explore(m=m, name = "Mahale", column= "REGION", legend=False)
```

حالا روی نقشه نمایششون بده.

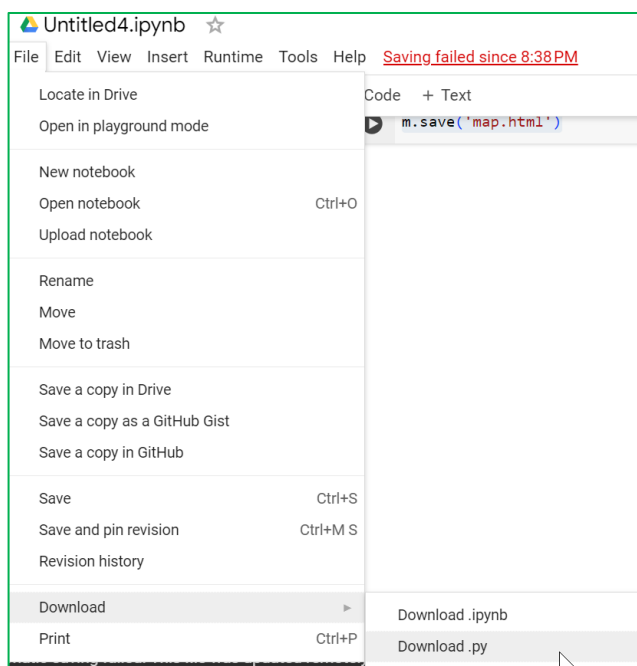
```
folium.LayerControl().add_to(m)
```

```
display(m)
```



## ذخیره notebook به فرمت ipynb و .py

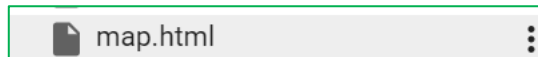
کدت رو میتونی به دو فرمت مختلف نوت بوک یا `ipynb` و پایتون یا `.py` دانلود کنی و جاهای دیگه ازش استفاده کنی. روی گزینه `file\download` برو و به فرمتی که میخوای دانلودش کن.



## ذخیره نقشه در یک فایل HTML

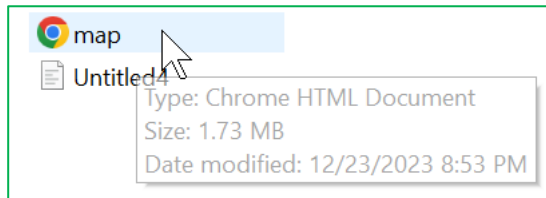
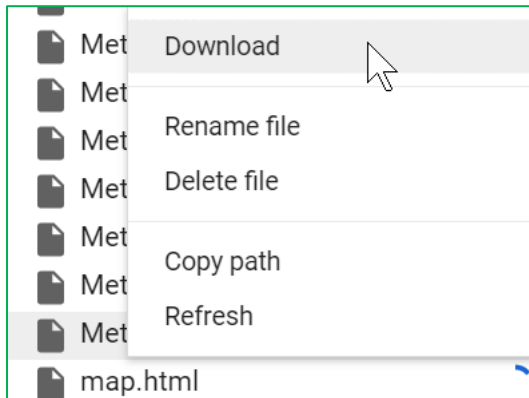
میتونی کل نقشه رو به صورت یه فایل html ذخیره کنی و در اختیار بقیه قرار بدی. با گزینه `map.save('map.html')` میشه لایه رو تبدیل کرد.

`m.save('map.html')`



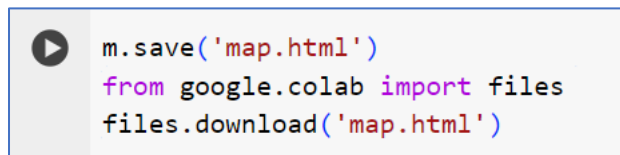
روی اسم لایه ساخته شده کلیک راست کن و download

رو کلیک کن.



اگه فایل مورد نظر توی لیست محتویات لایه دیده نشد باید از دستور زیر استفاده کنید که روی سیستمتون ذخیره شه.

```
from google.colab import files
files.download('map.html')
```



فایل تو درایوی ذخیره میشه که برای دانلود فایلهای سیستم معرفی کردید.

## اضافه کردن دایره روی نقشه

میشه با folium روی بخشی از نقشه دایره انداخت و طوری تنظیمش کرد که با زوم کردن روی نقشه دایره کوچیک و بزرگ شه. میشه خود دایره رو رنگ داد یا بهش ضخامت دور داد و ... همه این کارها با `folium.Circle()` انجام میشه. تو پرانترش میتونی همه تنظیماتی که در ادامه می‌آد رو انجام بدی.

نکته جذابش گذاشتن popup روی دایره هست. برای مثال میخوای عکس یکی از ایستگاه‌های مترو به همراه مشخصاتش روی دایره بیفته. طول و عرض جغرافیایی این ایستگاه اعداد زیر هست:

[۵۱٫۴۳۴۱۰۳۳۴۸۹۹۹, ۳۵٫۷۶۰۰۵۰۸۸۰۸۵۹]

میتونی از دستور `folium.popup()` استفاده کنی. تو پرانترش میتونی با html کد بنویسی مثلاً سرتیتر بهش بدی یا یه عکس بهش لینک کنی.

فرق tooltip با popup اینه که اگه تولتیپ داشته باشی با بردن موس روی عارضه بهت اطلاعاتش رو نشون میده ولی با popup باید روی عارضه کلیک کنی که اطلاعاتش رو ببینی. کد زیر رو بنویس و اطلاعات خودت رو جایگزین اطلاعاتش کن.

```
folium.Circle(radius=۵۰۰,
              location=[۳۵٫۷۶۰۰۵۰۸۸۰۸۵۹, ۵۱٫۴۳۴۱۰۳۳۴۸۹۹۹],
              fill=True,
```

```

tooltip="This is a Circle",
popup=folium.Popup( ""<h2>This is a popup</h2></br> This is a <b>new line</b></br> """,
max_width=۵۰۰)
).add_to(mapObj)

```

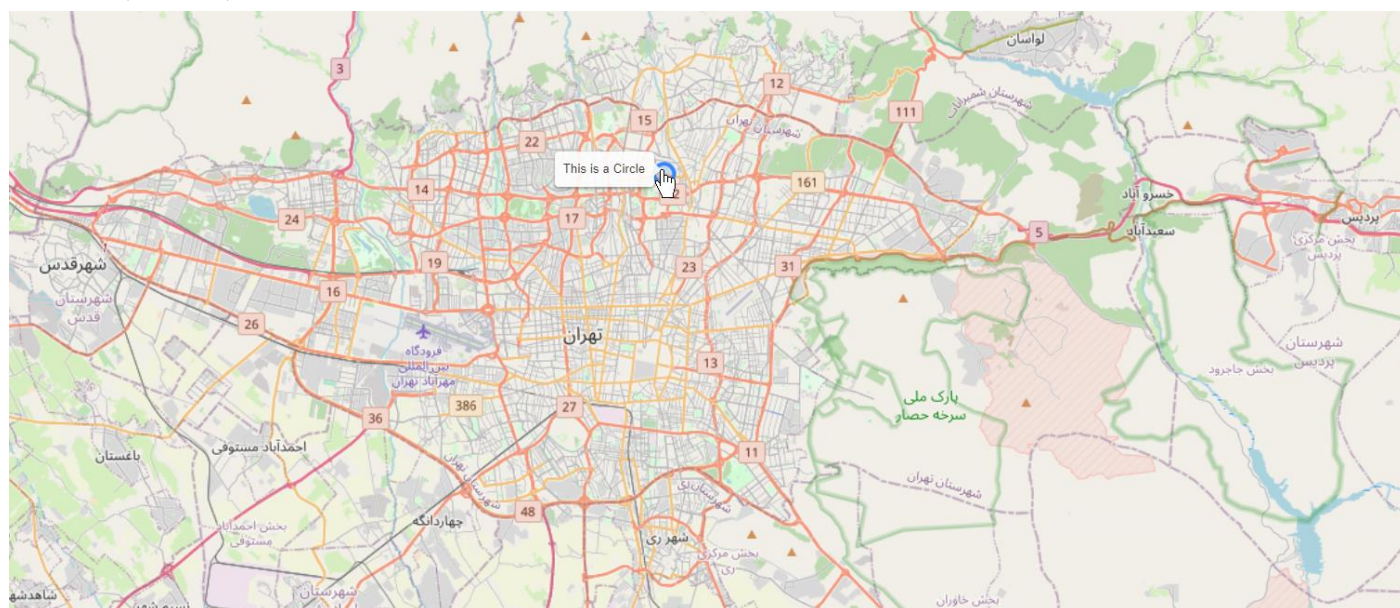
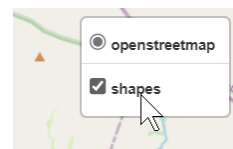
اگه بخوای که دایره روی نقشه نمونه میتونی بهش یه لایه اختصاص بدی که هر وقت خواستی خاموش و روشنش کنی. این کار با دستور **folium.FeatureGroup()** انجام میشه که بهتر هست به یه اسم اختصاصش بدی و انتهایش باید بگی به نقشه اضافه شه یعنی `add_to(mapObj)`. بعد کد بالا رو زیرش بنویسی و اینبار بگی به اسمی که اختصاصش دادم اضافه اش کن.

تهش هم باید `folium.LayerControl().add_to()` رو بنویسی و به `mapObj` اختصاصش بدی. بعد با `display()` نمایشش بدی. مثل کد زیر.

```

mapObj = folium.Map(location=[۳۵,۷۰۷۷۴۰۲, ۵۱,۱۸۲۸۴۱], zoom_start=۱۰)
shapesLayer = folium.FeatureGroup(name="shapes").add_to(mapObj)
folium.Circle(radius=۵۰۰,
location=[۳۵,۷۱۰۰۵۰۸۸۰۸۵۹, ۵۱,۴۳۴۱۰۳۳۴۸۹۹۹],
fill=True,
tooltip="This is a Circle",
popup=folium.Popup( ""<h2>This is a popup</h2></br> This is a <b>new line</b></br> """,
max_width=۵۰۰)
).add_to(shapesLayer)
folium.LayerControl().add_to(mapObj)
display(mapObj)

```

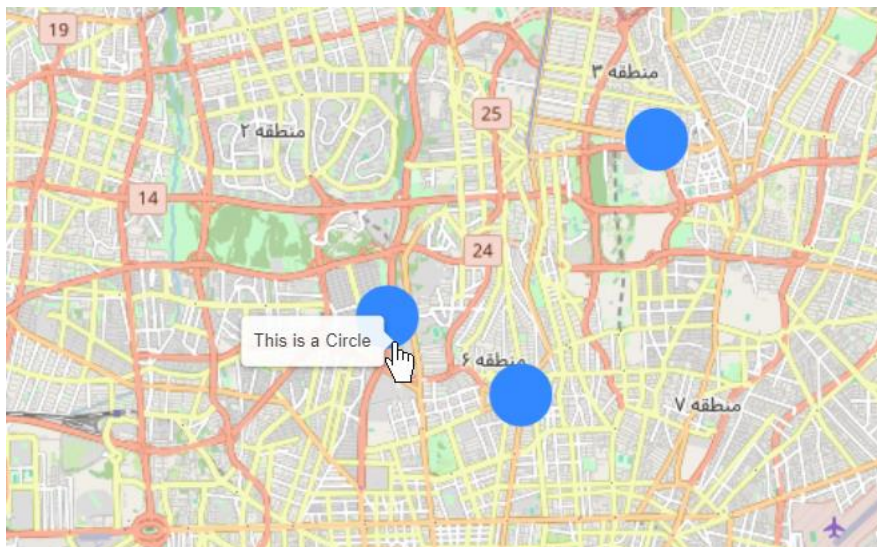


میتونی براش یه حلقه for بنویسی و چند تا دایره رو همزمان بندازی روی نقشه. برای مثال فرض کن x,y و همینطور شعاع (۵۰۰) سه تا نقطه رو داری:

```
mapObj = folium.Map(location=[۳۵,۷۰۷۷۴۰۲, ۵۱,۱۸۲۸۴۱], zoom_start=۱۰)
shapesLayer = folium.FeatureGroup(name="shapes").add_to(mapObj)
circlesData = [
    [۳۵,۷۶۰۰۵۰۸۸۰۸۵۹, ۵۱,۴۳۴۱۰۳۳۴۸۹۹۹, ۵۰۰],
    [۳۵,۷۲۰۶۲۳۲۸۳۷۸۱, ۵۱,۴۰۸۲۰۵۳۱۸۶۵۸, ۵۰۰],
    [۳۵,۷۳۲۶۵۳۵۰۴۹۷۱, ۵۱,۳۸۲۷۵۵۶۸۶۸۴۷, ۵۰۰]
]
for d in circlesData:
    folium.Circle(
        radius=d[۲],
        location=[d[۰], d[۱]],
        fill=True,
        fill_opacity=۱,
        tooltip="This is a Circle",
        popup=folium.Popup("""<h۲>This is a popup</h۲><br> This is a <b>new line</b></br>""",
        max_width=۵۰۰)
    ).add_to(shapesLayer)
```

```
folium.LayerControl().add_to(mapObj)
display(mapObj)
```

تو کد بالا d[۰] اشاره به x داره و d[۱] اشاره به y. d[۲] هم در واقع عدد سوم یا مقدار شعاع دایره هست. fill رو False بده که داخل دایره‌ها رنگی شه و با fill\_opacity بهشون شفافیت بده که اطلاعات زیرش دیده شه.



## نحوه مارکر گذاشتن روی نقشه

از دستور folium.Marker() میتونی روی نقشه مارکر بذاری. میتونی به مارکر tooltip و popup اضافه کنی.

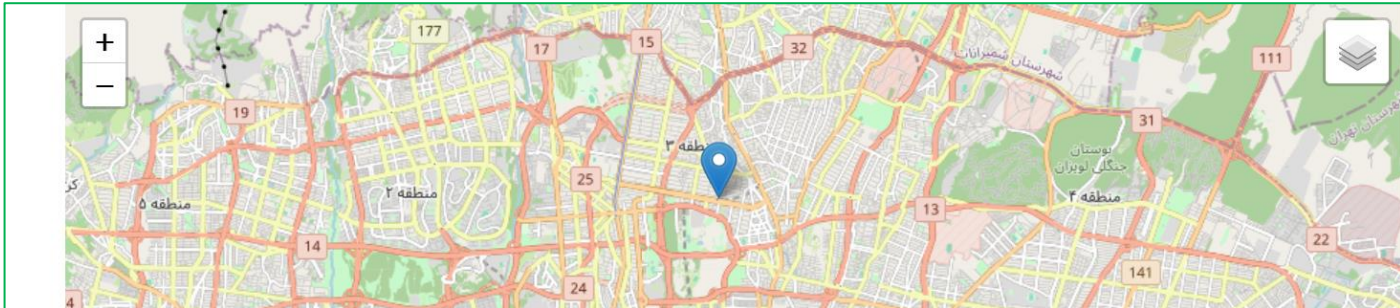


```
myMap = folium.Map(location=[۳۵,۷۰۷۷۴۰۲, ۵۱,۱۸۲۸۴۱], zoom_start=۱۰)
```

```
folium.Marker(location=[۳۵,۷۶۰۰۵۰۸۸۰۸۵۹, ۵۱,۴۳۴۱۰۳۳۴۸۹۹۹],  
    tooltip='This is a tooltip',  
    popup=folium.Popup('This is a <b>pop</b> up', max_width=۵۰۰)).add_to(myMap)
```













```
folium.LayerControl().add_to(myMap)
```

```
display(myMap)
```



میتونی شکل مارکر رو شخصی سازی کنی. به این نمادها گلیفیکون میگوین. از لینک زیر میتونی بهشون دسترسی داشته باشی.

[Components · Bootstrap \(getbootstrap.com\)](https://getbootstrap.com/components/)

 glyphicon glyphicon-glass	 glyphicon glyphicon-music	 glyphicon glyphicon-search	 glyphicon glyphicon-heart
 glyphicon glyphicon-star	 glyphicon glyphicon-star-empty	 glyphicon glyphicon-user	 glyphicon glyphicon-film
 glyphicon glyphicon-th-large	 glyphicon glyphicon-th	 glyphicon glyphicon-th-list	 glyphicon glyphicon-ok

برای مثال میخوایم مارکر ستاره رو استفاده کنیم. از دستور `folium.Icon()` استفاده میکنیم و توی پرانتزش اسم گلیفیکون رو مینویسیم.

```
folium.Icon(icon = 'glyphicon-star').add_to(myMap)
```

واسه استفاده از آیکنهای مشخص برای مارکر از سایت زیر استفاده کن:

[Font Awesome ۴,۵,۶ Class Explorer \(artlung.com\)](https://fontawesome.com/class-explorer)

fa-glass 🍷	fa-music 🎵	fa-search 🔍	fa-envelope-o ✉	fa-heart ❤
fa-star ★	fa-star-o ☆	fa-user 👤	fa-film 🎬	fa-th-large 📊
fa-th 📊	fa-th-list 📋	fa-check ✔	fa-remove ✕	fa-close ✕
fa-times ✕	fa-search-plus 🔍	fa-search-minus 🔍	fa-power-off 🔌	fa-signal 📶
fa-gear ⚙	fa-cog ⚙	fa-trash-o 🗑	fa-home 🏠	fa-file-o 📄
fa-clock-o 🕒	fa-road 🛣	fa-download 📥	fa-arrow-circle-o-down ⬇	fa-arrow-circle-o-up ⬆
fa-inbox 📧	fa-play-circle-o ⏮	fa-rotate-right ↻	fa-repeat 🔁	fa-refresh ↻
fa-list-alt 📋	fa-lock 🔒	fa-flag 🚩	fa-headphones 🎧	fa-volume-off 🔇

کدش رو باید به صورت زیر بنویسی:

```
folium.Icon(Icon = 'calendar', prefix='fa', color='red').add_to(myMap)
```

کد زیر فرودگاه مهرآباد رو به همراه تصویر و یه لینک معرفی فرودگاه آورده.

```
myMap = folium.Map(location=[۳۵,۷۰۷۷۴۰۲, ۵۱,۱۸۲۸۴۱], zoom_start=۱۰)
```

```
folium.Marker(location=[۳۵,۶۸۹۱۶۷, ۵۱,۳۱۳۳۳۲],
```

```
icon= folium.Icon(icon='glyphicon-plane', color='green'),
```

```
popup = folium.Popup(
```

```
"""
```

```
<br/>
```

```
<h۴>
```

```
glyphicon-plane icon from bootstrap.<br/>
```

```
</h۴>
```

```
<h۵>Check out more <a
```

```
href="https://www.bing.com/search?q=%d۹%۸۱%d۸%b۱%d۹%۸۸%d۸%af%da%af%d۸%a۷%d۹%۸۷+%d۹%۸۵%d۹%۸۷%d۸%b۱%d۸%a۷%d۸%a۸%d۸%a۷%d۸%af&FORM=HDRSC۱" target="_blank">here</a></h۵>
```

```
""",
```

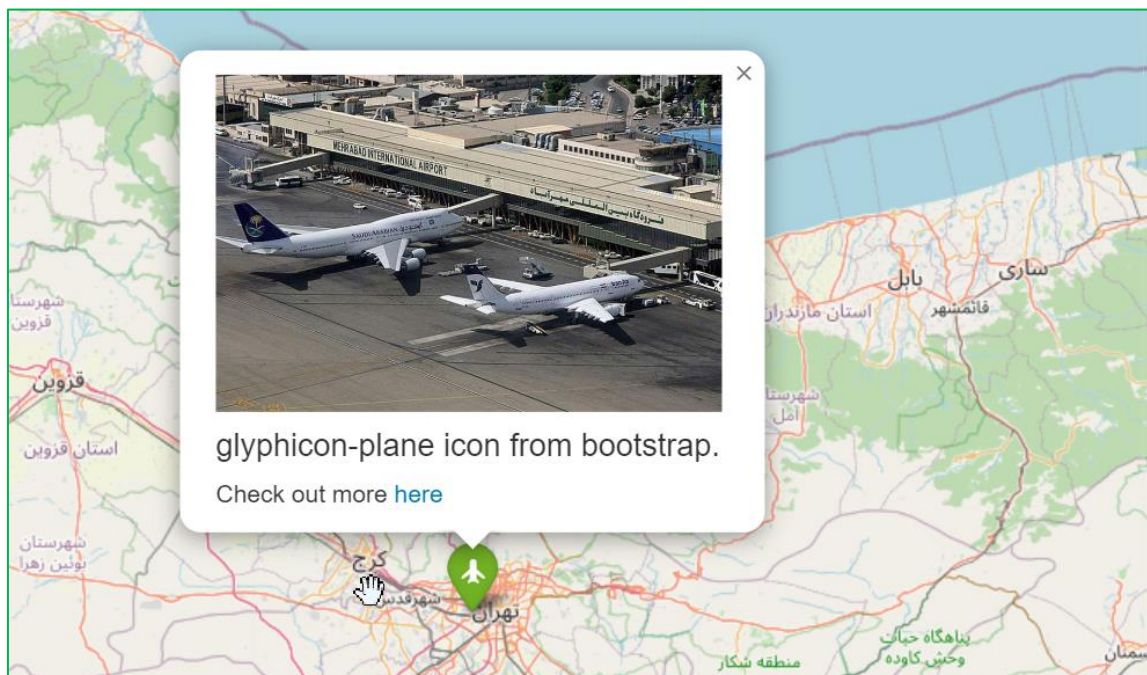
```
max_width=۳۰۰),
```

```
tooltip='Bootstrap example'
```

```
).add_to(myMap)
```

```
folium.LayerControl().add_to(myMap)
```

```
display(myMap)
```



## ساخت نقشه از اکسل

جدول اکسل باید اطلاعات x و y داشته باشد. فرض کن میخوایم جدول اکسلی رو بیاریم که اطلاعات طول و عرض جغرافیایی نقاط ایستگاه های مترو رو داره. میتونی ازش bubble map بسازی. واسه اینکار باید از کتابخانه های زیر استفاده کنی.

```
import pandas as pd
import geopandas as gpd
import folium
from shapely.geometry import Point
from IPython.display import display
```

فایل اکسل رو با کد زیر بخون.

```
df = pd.read_excel('/content/Station_XY.csv.xlsx')
df.head()
```

```
df = pd.read_excel('/content/Station_XY.csv.xlsx')
df.head()
```

	No	X	Y
0	1	536092	3935441
1	2	538386	3938990
2	3	538003	3940777
3	4	537662	3942565
4	5	540615	3943308

با کد زیر جدول اکسل رو باید تبدیلش کنیم به لایه ای جغرافیایی یعنی لایه ای که ستونی به اسم Geometry داره. برای اینکار به ستونهای X و Y نیاز داریم.

```
geometry = [Point(xy) for xy in zip(df['X'], df['Y'])]
```

```
gdf = gpd.GeoDataFrame(df, geometry=geometry, crs='EPSG:۳۲۶۳۹')
```

با کد زیر میتونی این لایه مکانی رو به Shapefile تبدیل کنی.

```
ST = 'Station.shp'
```

```
gdf.to_file(ST)
```

اگه خط کد زیر رو ننویسیم فایل shapefile درست خونده نمیشه در واقع با تعیین کدگذاری (encoding) به صورت صریح میشه Shapefile رو خوند.

```
Gdf = gpd.read_file(ST, encoding='utf-۸')
```

با کد زیر با folium یه نقشه میسازیم و لایه GeoJSON رو بهش اضافه میکنیم.

```
mapObj = folium.Map(location=[۳۵,۷۰۷۷۴۰۲, ۵۱,۱۸۲۸۴۱], zoom_start=۱۰)
```

```
folium.GeoJson(gdf).add_to(mapObj)
```

با کد زیر هم کنترل لایه اضافه میکنیم و نقشه رو نمایش میدیم.

```
folium.LayerControl().add_to(mapObj)
```

```
display(mapObj)
```

## ساخت نقشه heatmap با folium

برای ساخت نقشه داغ یا heatmap باید HeatMap رو از پلاگینهای folium بیاری.

```
Import folium
```

```
from folium.plugins import HeatMap
```

مثل مراحل قبل یه نقشه درست کن.

```
mapObj = folium.Map(location=[۳۵,۷۰۷۷۴۰۲, ۵۱,۱۸۲۸۴۱], zoom_start=۱۰)
```

میتونی اطلاعات طول و عرض جغرافیایی و مقدار فیلدی که میخوای بر اساس اون نقشه ساخته شه رو به صورت زیر وارد کنی. فرض کن نقاط زیر نقاطی هست که در اون سرقت اتفاق افتاده. عدد اول عرض و عدد دوم طول و عدد سوم تعداد جرم رو نشون میده (داده‌ها فرضی هستن).

```
data = [
    [35.719404, 51.404184, 8],
    [35.715020, 51.407536, 2],
    [35.711839, 51.378626, 5],
    [35.708615, 51.406760, 4],
    [35.712736, 51.424012, 2],
    [35.692807, 51.397442, 1],
    [35.686513, 51.356849, 7],
    [35.746660, 51.370383, 9],
    [35.722920, 51.320382, 3],
    [35.734606, 51.440806, 2]
]
```

واسه ساخت HeatMap باید از دستور HeatMap() استفاده کنی.

```
HeatMap(data).add_to(mapObj)
```

یه کنترل لایه درست کن و نقشه رو نمایش بده.

```
folium.LayerControl().add_to(mapObj)
```



display(mapObj)



## پلاگینها در فولیوم

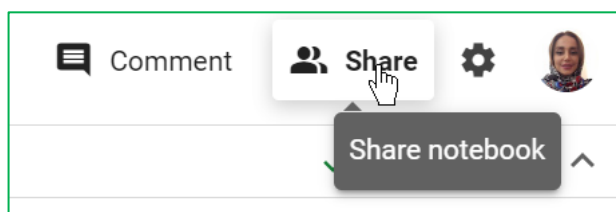
فولیوم دارای پلاگینهای متفاوتی هست که میشه باهاش کارهای متفاوتی انجام داد. در زیر لیستشون رو میتونی ببینی.

- [BoatMarker](#)
- [BeautifyIcon](#)
- [Draw](#)
- [DualMap plugin](#)
- [FeatureGroupSubGroup](#)
- [FloatImage](#)
- [Fullscreen](#)
- [Geocoder](#)
- [GroupedLayerControl](#)
- [Heatmap](#)
- [HeatMapWithTime Plugin](#)
- [LocateControl](#)
- [MarkerCluster](#)
- [MiniMap](#)
- [MeasureControl](#)
- [MousePosition](#)

- [Pattern plugins](#)
- [PolylineOffset](#)
- [PolylineTextPath and AntPath](#)
- [Realtime plugin](#)
- [ScrollZoomToggler](#)
- [Search](#)
- [SemiCircle](#)
- [SideBySideLayers](#)
- [TagFilterButton](#)
- [Terminator](#)
- [TimeSliderChoropleth](#)
- [TimestampedGeoJson](#)
- [Vector tiles using VectorGridProtobuf](#)
- [TimestampedWmsTileLayers](#)

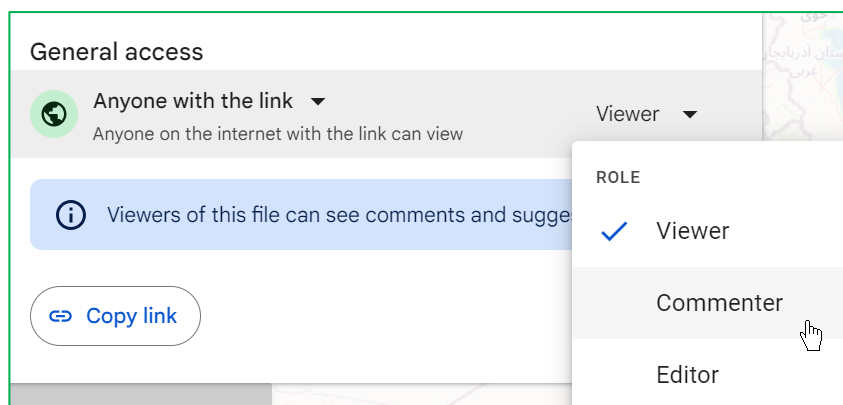
این ابزارها بیشتر برای اضافه کردن ویژگی‌های مختلف به نقشه‌ها استفاده می‌شوند. برای مثال، ابزار MarkerCluster برای گروه‌بندی نشانگرها در نقشه برای افزایش خوانایی استفاده می‌شود یا ابزار MeasureControl برای اندازه‌گیری فاصله و مساحت در نقشه استفاده می‌شود. هر کدام از این ابزارها و پلاگین‌ها وظایف و کاربردهای خاص خودشان را دارند که به بهبود و توسعه قابلیت‌های نقشه‌ها کمک می‌کنند. با کمک هوش مصنوعی می‌تونی به کد برای هر کدام بنویسی و کاربردهاش را روی نقشه ببینی.

## نکات تکمیلی در مورد Colab



لینکی که اگر بخوای نوت بوکت رو با کسی به اشتراک بذاری باید روی گزینه Share کلیک کنی.

تو این پنجره می‌تونی تعیین کنی که افرادی که لینک رو بهشون میدی چه کاری می‌تونن روی کد انجام بدن. برای مثال می‌تونن صرفاً ببینند باشند. می‌تونن کامنت بذارن یا اینکه اجازه ویرایش داشته باشن.



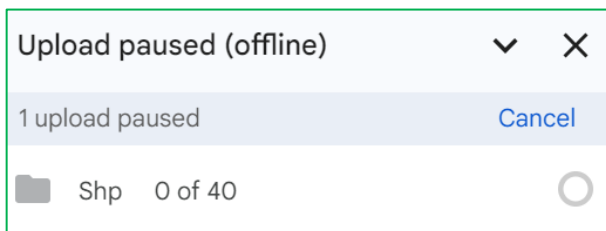
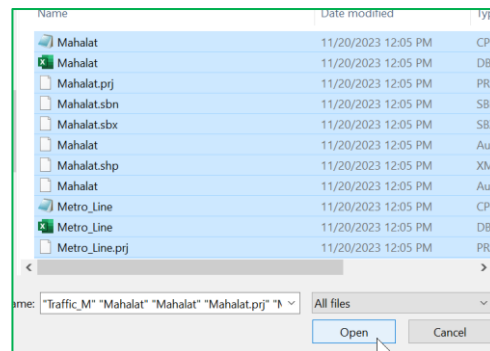
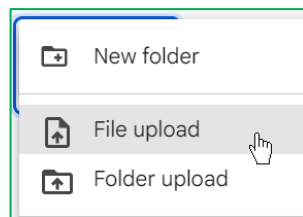
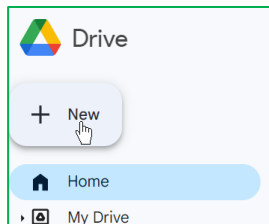
از بخش copy link رو کپی کن و روی گزینه done کلیک کن. لینک زیر رو میتونی در اختیار دیگران قرار بدی که کدت رو مشاهده کنن.

<https://colab.research.google.com/drive/1XWg-FJIGMOBffEnD^TgcNYP3aMk2nvYi?usp=sharing>

نکته‌ای که باید مد نظرت باشه اینه که اگه لینک رو روی Editor بذاری و به فرد اجازه ویرایش بدی باید لایه‌ها رو به جای اینکه از روی سیستم بخونی باید داده‌ها رو روی google drive ذخیره کنی که دیگران هم بتونن دسترسی داشته باشن. داده‌ها رو بیار تو لینک زیر.

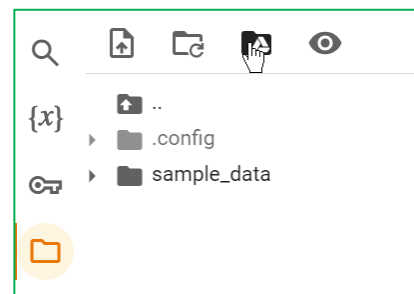
### [Home - Google Drive](#)

روی گزینه New بزن و از گزینه File upload لایه‌ها رو بیار یا روی گزینه Folder upload بزن و فولدری که لایه‌ها داخلش هست رو اضافه کن.



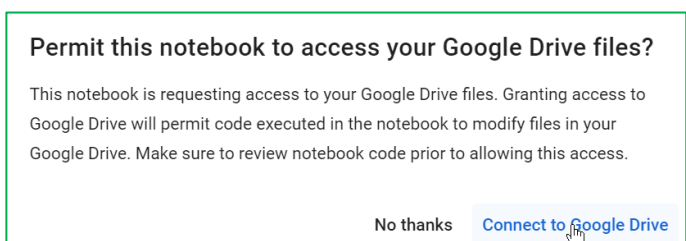
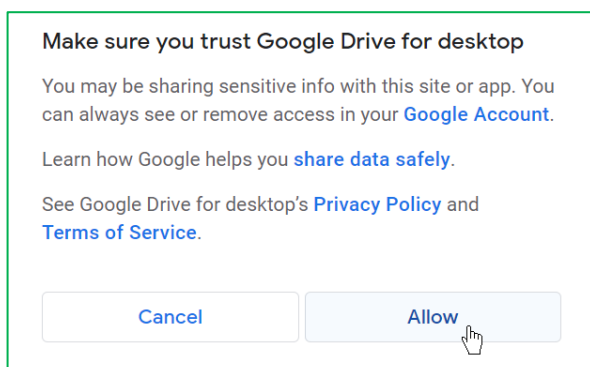
صبر کن که آپلود شه.

حالا به گوگل کولب برگرد و از بخش file روی Mount Drive برو. ازت میخواد که اجازه دسترسی به گوگل درایو رو بهش بدی. یه پیام می‌آد که میگه این سلول رو run کن که به گوگل درایو وصل شی.



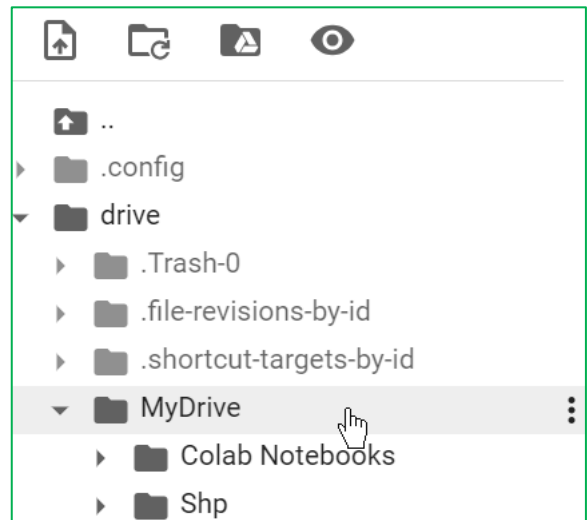
```
from google.colab import drive
drive.mount('/content/drive')
```

به اکانت Gmail وصل شو.



پایین صفحه روی گزینه Allow کلیک کن.

فولدر shp به این بخش اضافه میشه و میتونی روی لایه‌هاش کلیک کنی و copy path بگیری و به جای مسیرهای قبلی این مسیر رو بنویسی.



تو جزوه بعدی با یه کتابخونه دیگه پایتون برای مصورسازی داده‌های جغرافیایی به اسم rasterio آشنا میشی.