

ipyleaflet

- ① Windows Powershell あるいはコマンドプロンプトを用いて、`pip install ipyleaflet`
- ② jupyter notebookを起動

```
from ipyleaflet import Map
```

```
m=Map(center=(35.68, 139.75), zoom=12)  
m
```

北緯 東経

変数=Map()

center マップのセンター
zoom 0~18でズーム

マウスでドラッグして表示位置も変えられる

クリックして
拡大・縮小可能→



マーカー描画

変数=Maker()

[Map]+=[Maker]

```
from ipyleaflet import Map, Marker

(lat, lon)=(35.682, 139.765)
m=Map(center=(lat, lon), zoom=17)
for n in range(-3, 4, 1):
    mark= Marker(location=(lat, lon +n/2000))
    m +=mark
m
```

```
from ipyleaflet import Map, Marker
from ipyleaflet import Map, MarkerCluster
from ipyleaflet import Map
```

```
(lat, lon)=(35.682, 139.765)
mrks=[ ]
m=Map(center=(lat, lon), zoom=17)
for n in range(-3, 4, 1):
    mrks.append(Marker(location=(lat, lon +n/1000)))
marker_cluster=MarkerCluster(markers=mrks)
m +=marker_cluster
m
```

拡大

縮小



直線描画

変数=Polyline(locations=位置情報, radius=半径)

```
from ipyleaflet import Map, Polyline
```

```
(lat, lon)=(35.682, 139.765)
```

```
m=Map(center=(lat, lon), zoom=15)
```

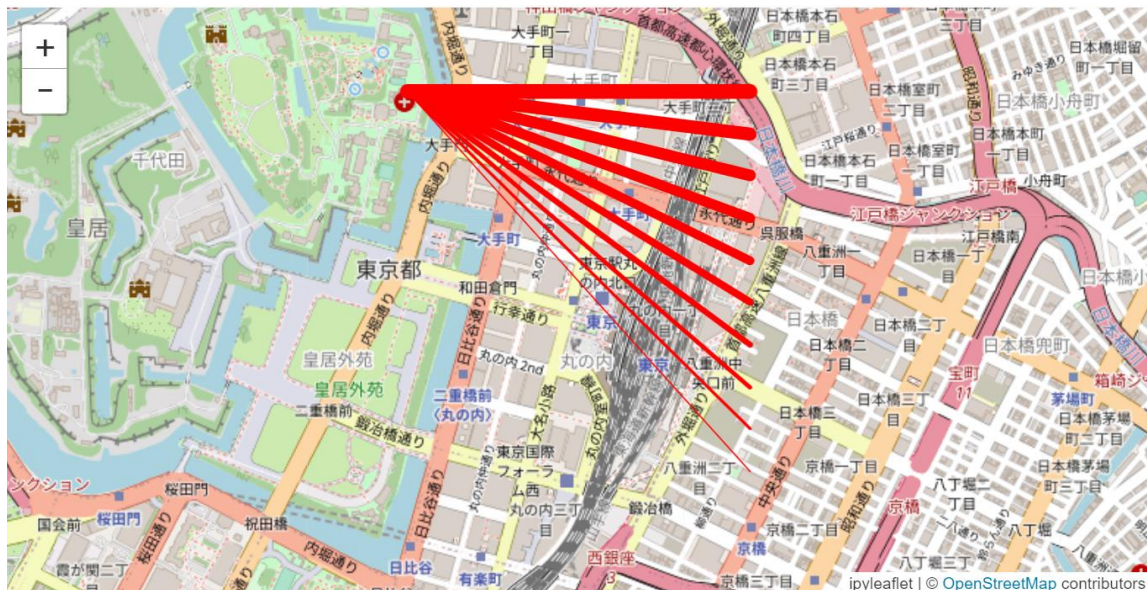
```
for n in range(0, 10):
```

```
    pl=Polyline(locations=[(lat+0.005, lon-0.005),  
                           (lat+0.005-0.001*n, lon+0.005)],  
               weight=10-n, color='#f00')
```

```
    m +=pl
```

```
m
```

Weihhtを変えて線の太さが細くなる



四角形 変数=Rectangle(bounds=領域)

円 変数= Circle(locations=位置, radius=半径)

イメージ

変数=ImageOverlay(url=ファイル, bounds=領域)

```
from ipyleaflet import Map, ImageOverlay
```

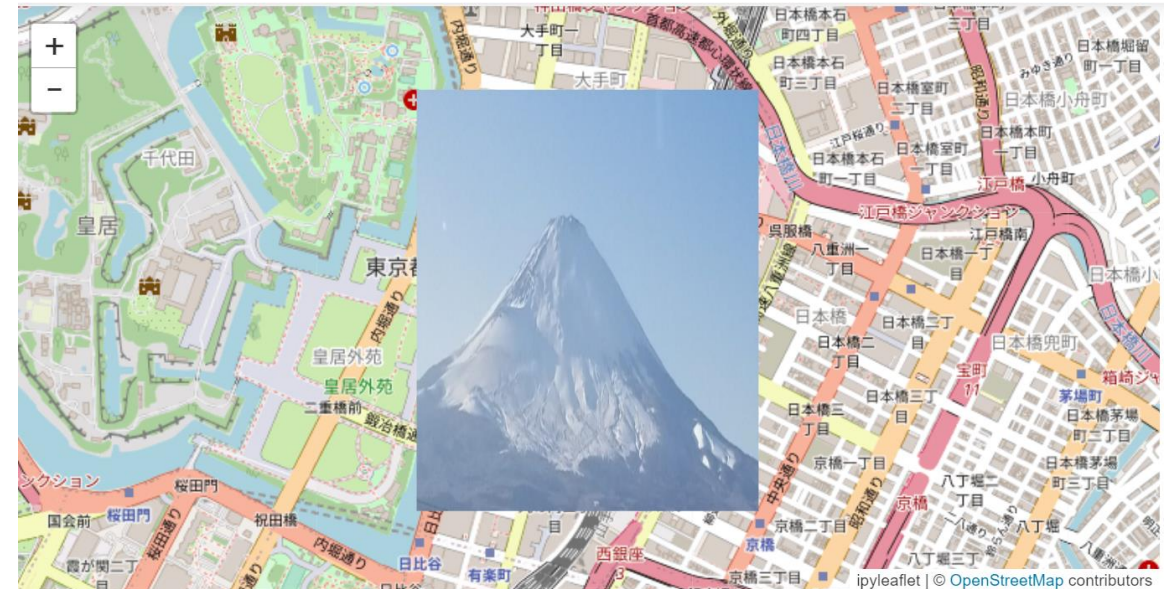
```
(lat, lon)=(35.682, 139.765)
```

```
m=Map(center=(lat, lon), zoom=15)
```

```
io=ImageOverlay(url='富士山.jpg',  
               bounds=[(lat-0.005, lon-0.005), (lat+0.005, lon+0.005)])
```

```
m +=io
```

```
m
```



描画コントロール

```
from ipyleaflet import Map, DrawControl
```

```
(lat, lon)=(35.682, 139.765)  
m=Map(center=(lat, lon), zoom=17)
```

```
dc=DrawControl()  
m.add_control(dc)  
m
```

線描画 →
マーカー →



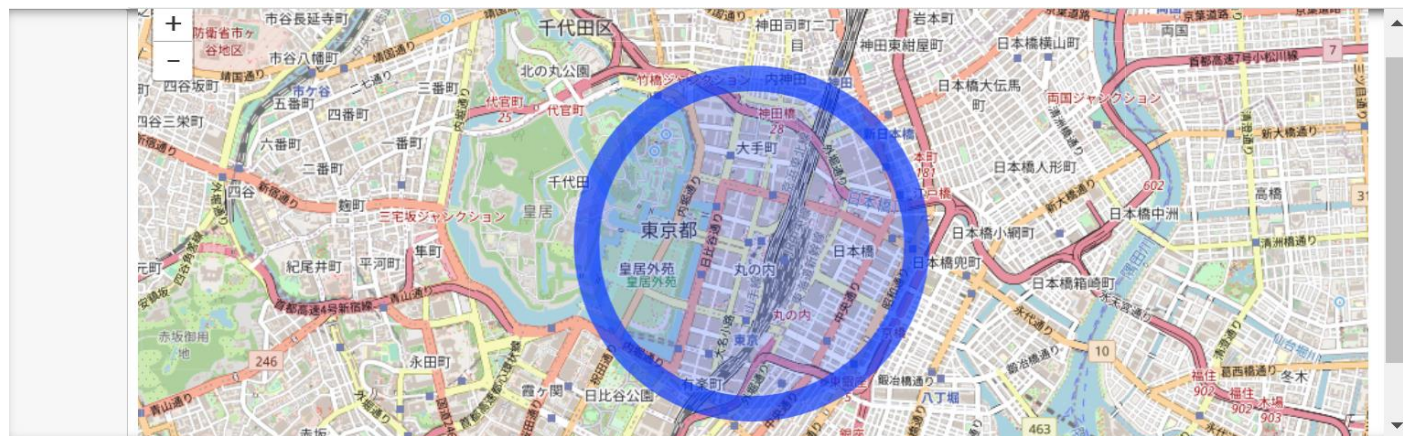
スライダー設定

```
from ipyleaflet import Map,Circle
from ipyleaflet import Map
(lat,lon)=(35.682,139.765)
m=Map(center=(lat,lon),zoom=17)
m
```



次のセルに以下スクリプト入力

```
c=Circle(location=m.center)
m +=c
c.interact(weight=(0,25,1),opacity=(0.0,1.0,0.01))
```



```
In [25]: c=Circle(location=m.center)
m +=c
c.interact(weight=(0,25,1),opacity=(0.0,1.0,0.01))
```

weight 20 opacity 0.65



```
=Circle(location=m.center)
+=c
interact(weight=(0,25,1),opacity=(0.0,1.0,0.01))
```

weight 2 opacity 1.00



```
>=Circle(location=m.center)
n +=c
.interact(weight=(0,25,1),opacity=(0.0,1.0,0.01))
```

weight 25 opacity 0.55

マーカー表示

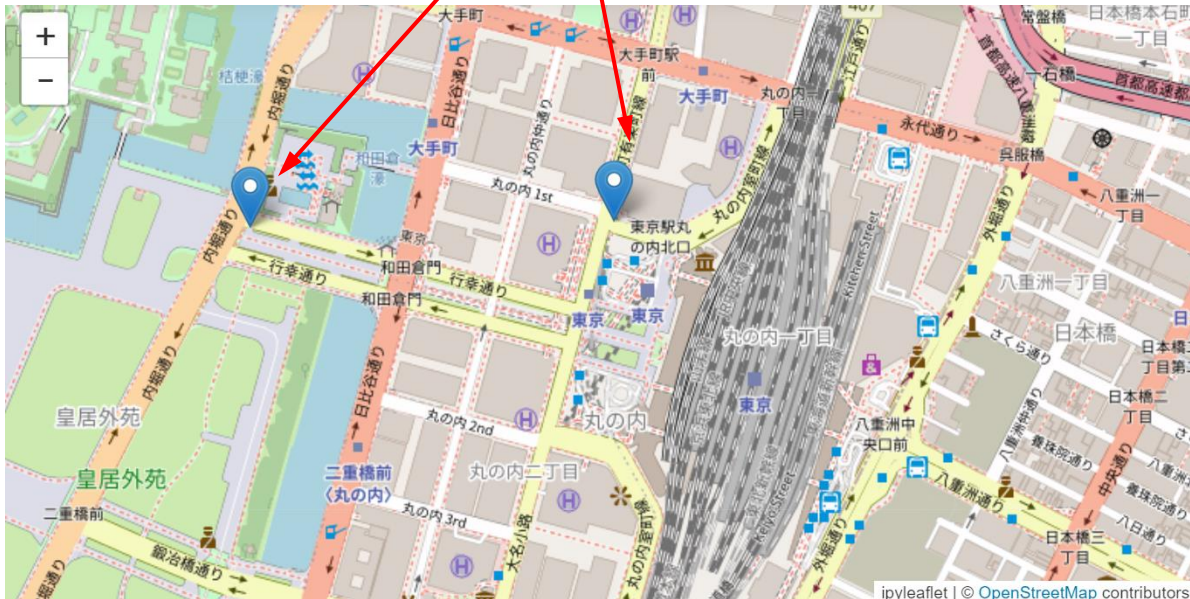
```
from ipyleaflet import Map, Marker

def event_handle(**kwargs):
    if kwargs['type'] == 'click':
        global m
        mk=Marker(location=kwargs['coordinates'])
        m +=mk
(lat, lon)=(35.682, 139.765)
m=Map(center=(lat, lon), zoom=17)
m.on_interaction(event_handle)
m
```

[map].on_interaction(関数)
def event_handle(**kwargs)

kwargs['type'] == 'click'
mk=Marker(location=kwargs['coordinates'])
クリックした時のマウスポインタの位置を
基にマーカー作成

クリックした位置にマーカー表示



リアルタイム表示

```
from ipyleaflet import Map, Marker
from ipywidgets import Label

label1=Label()
label2=Label()
display(label1, label2)

def event_handle(**kwargs):
    if kwargs['type']=='mousemove':
        here=kwargs['coordinates']
        label1.value='Latitude:'+str(here[0])
        label2.value='Longitude:'+str(here[1])
(lat, lon)=(35.682, 139.765)
m=Map(center=(lat, lon), zoom=17)
m.on_interaction(event_handle)
m
```

`kwargs['type']=='mousemove'`

Display(ウィジェット、ウィジェット、・・・)

マウスポインタ位置の経度・緯度表示

Latitude:35.685634712135766
Longitude:139.76113542156818



```
from ipywidgets import Label, Button
```

```
label=Label()  
label.value='This is Label.'  
display(label)
```

This is Label.

Interact(関数、因数の指定)

```
from ipywidgets import interact
```

```
def fn(x):  
    return x*2
```

```
interact(fn, x=100)
```

x  50

100

ipywidgets

Button.on_click(handle_event)

```
from sympy import *  
from ipywidgets import Label, IntText, Button  
label=Label()  
input=IntText()  
label.value='please slide'  
  
x=Symbol('x')  
re=x**2-x*3  
  
def handle_event(target):  
    global re, label, input  
    label.value='result:'+str(re.subs(x, input.value))  
  
button=Button(description='click')  
button.on_click(handle_event)  
  
display(label, input, button)
```

please slide

5

click

result:10

5

click

真偽の値操作

```
from ipywidgets import *  
label=Label(value='slide me!')  
check=Checkbox(description='check now')  
toggle=ToggleButton(description='Toggle')  
display(label, check, toggle)  
  
def on_change(change):  
    global check, toggle  
    label.value='check:' +str(check.value)+' ,toggle:' +str(toggle.value)  
  
check.observe(on_change)  
toggle.observe(on_change)
```

slide me!

☐ check now



Toggle

check:True ,toggle:False

☒ check now



Toggle

check:True ,toggle:True

☒ check now

Toggle

複数項目から選択

```
from ipywidgets import *
label=Label(value='slide me!')
radio=RadioButtons(description='Platform:',¥
    options=['Windows','macOS','ChromeOS'])
drop=Dropdown(description='Mobile',¥
    options=['Android','iOS','Flutter'])
display(label,radio,drop)

def on_change(change):
    global radio,drop
    label.value='Platform: ' +str(radio.value)+' , Mobile: ' +str(drop.value)

radio.observe(on_change)
drop.observe(on_change)
```

slide me!

Platform: macOS , Mobile: Android

Platform: ☒ Windows
☐ macOS
☐ ChromeOS

Mobile

Platform: ☐ Windows
☒ macOS
☐ ChromeOS

Mobile

Platform: macOS , Mobile: Android

Platform: ☐ Windows
☒ macOS
☐ ChromeOS

Mobile

Platform: macOS , Mobile: iOS

Platform: ☐ Windows
☒ macOS
☐ ChromeOS

Mobile

Dropdown

1 RadioButtons

Accordion

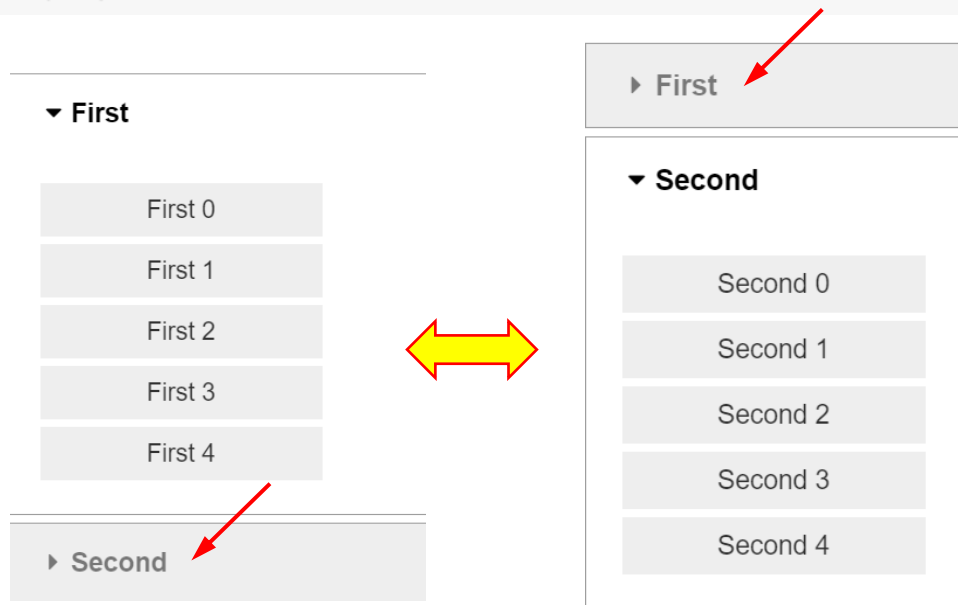
```
from ipywidgets import *

btnsH=[]
for n in range(0,5):
    btnsH.append(Button(description='First ' + str(n)))
box1=VBox(btnsH)

btnsV=[]
for n in range(0,5):
    btnsV.append(Button(description='Second ' + str(n)))
box2=VBox(btnsV)

accordion=Accordion(children=[box1, box2])
accordion.set_title(0, 'First ')
accordion.set_title(1, 'Second ')

display(accordion)
```



Tab

```
from ipywidgets import Tab

btnsH=[]
for n in range(0,5):
    btnsH.append(Button(description='First ' + str(n)))
box1=VBox(btnsH)

btnsV=[]
for n in range(0,5):
    btnsV.append(Button(description='Second ' + str(n)))
box2=VBox(btnsV)

tab=Tab(children=[box1, box2])
tab.set_title(0, 'First ')
tab.set_title(1, 'Second ')

display(tab)
```

同じスクリプト

置き換えるのみ

