We use imgur as external service other than redis. Imgur is an online photo album site, launched in 2009, which offers free storage services for photo and video.

links of tutorial: https://apidocs.imgur.com

Example Request:

```
import requests

url = "https://api.imgur.com/3/image"

payload = {'image': 'R0lGODlhAQABAIAAAAAAAP///yH5BAEAAAAALAAAAAABAAEAAAIBRAA7'}
files = [

]
headers = {
  'Authorization': 'Client-ID {{clientId}}'
}

response = requests.request("POST", url, headers=headers, data = payload, files = files)

print(response.text.encode('utf8'))
```

Example Response:


```
{
  "data": {
    "id": "orunSTu",
    "title": null,
    "description": null,
    "datetime": 1495556889,
    "type": "image/gif",
    "animated": false,
    "width": 1,
    "height": 1,
    "size": 42,
    "views": 0,
    "bandwidth": 0,
    "vote": null,
    "favorite": false,
    "nsfw": null,
    "section": null,
    "account_url": null,
    "account_id": 0,
    "is_ad": false,
    "in_most_viral": false,
    "tags": [],
    "ad_type": 0,
    "ad_url": "",
    "in_gallery": false,
    "deletehash": "x70po4w7BVvSUzZ",
    "name": "",
    "link": "http://i.imgur.com/orunSTu.gif"
  },
  "success": true,
  "status": 200
}
```

From the example request, it contains a parameter called 'clientId'. Actually, this id does not mean the account of imgur. Instead, it is the id of an application which can be registered from https://api.imgur.com/oauth2/addclient. After the register, we can get the 'clientId' and 'secret', which can upload pictures and video to personal server via them. We want to use this service to reply some matplotlib pictures in chart bot.

Specifically, we use 'pip install PyImgur' first. Then, we can define a function to get some image from matplotlib.pyplot, and then save them as png format, and then upload the image to imgur server as well as get the 'link'(url) from the response. Finally, the line chart bot can reply image message. Base on above, we design that the ordinate of the bar chart is the number, and the abscissa has three bars of different colors, which are the number of infections, the number of cures, and the number of deaths. Finally, with the imgur server, our chart bot will reply bar chart image of infections number in an area when it receives message of area name.

Specific function code is as follows:

```
243 def bar_chart_en(x):
244     pt.clf()
245     pt.cla()
246     a=writeinjson(ncovcity)
247     newslist=list(a["newslist"])
248     for i in range(0,34):
249         if translate(x) in newslist[i].values():
250             city=newslist[i]
251             con=city.get('confirmedCount')
252             cur=city.get('curedCount')
253             dead=city.get('deadCount')
254             data=[con,cur,dead]
255             labels=['Confirmed','Cured','Dead']
256             pt.bar(range(len(data)), data, color='rgb',tick_label=labels)
257             pt.xlabel(x),
258             pt.ylabel("Count")
259             pt.savefig('send.png')
260             CLIENT_ID = "135f2074e557c95"
261             PATH = "send.png"
262             im = pyimgur.Imgur(CLIENT_ID)
263             uploaded_image = im.upload_image(PATH, title="Uploaded")
264             return uploaded_image.link
```