

# get\_ICD10PCS

July 24, 2019

## 1 Translating ICD-9-PCS to ICD-10-PCS

```
[1]: import os
import requests
from bs4 import BeautifulSoup
os.getcwd()
```

```
[1]: '/Users/miaocai/Dropbox/RprojectMiao/Webscraping'
```

### 1.1 Define the function

```
[2]: headers={
    'authority': 'www.icd10data.com',
    'method': 'GET',
    'path': '/Convert/36.03',
    'scheme': 'https',
    'accept': 'text/html,application/xhtml+xml,application/xml;q=0.9,image/
→webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3',
    'accept-encoding': 'gzip, deflate, br',
    'accept-language': 'zh-CN,zh;q=0.9',
    'cache-control': 'max-age=0',
    'cookie': '_ga=GA1.2.1086716260.1563868202; _gid=GA1.2.1478799181.
→1563868202; __hstc=93424706.6b6b93f18691ef5d9414641b440bdd8c.1563868411367.
→1563868411367.1563868411367.1; hubspotutk=6b6b93f18691ef5d9414641b440bdd8c;
→__hssrc=1; __hssc=93424706.4.1563868411367',
    'upgrade-insecure-requests': '1',
    'User-Agent': 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_5) AppleWebKit/
→537.36 (KHTML, like Gecko) Chrome/75.0.3770.142 Safari/537.36',
}

def get_code(url):

    req=requests.get(url,headers=headers,timeout=50)
    bsObj=BeautifulSoup(req.text,'html.parser')
    ul=bsObj.find('ul',class_='ulConversion')#ul
    li_list=ul.find_all('li')#li
```

```

f=open('icd_10_pcs.txt','a+',encoding='utf-8')#txt

for li in li_list:
    a=li.find('a')
    alt=a.attrs['alt']
    pcs=alt.split(' ')[-1]
    #print(pcs)
    f.write('"' + pcs + '", ')
f.close()

```

## 1.2 Loop for a list of ICD-9-PCD codes

```

[3]: url_list=['https://www.icd10data.com/Convert/36.03',
              'https://www.icd10data.com/Convert/36.04',
              'https://www.icd10data.com/Convert/36.09']

for url in url_list:
    get_code(url)

```

## 1.3 Print the contents of the resulting .txt file

```

[4]: f0 = open('icd_10_pcs.txt','r',encoding='utf-8')
f0_contents = f0.read()
print(f0_contents)

```

```

"02700ZZ", "02710ZZ", "02720ZZ", "02730ZZ", "02C00ZZ", "02C10ZZ", "02C20ZZ",
"02C30ZZ", "3E07017", "3E070PZ", "3E07317", "3E073PZ", "02C03ZZ", "02C04ZZ",
"02C13ZZ", "02C14ZZ", "02C23ZZ", "02C24ZZ", "02C33ZZ", "02C34ZZ", "02700ZZ",
"02710ZZ", "02720ZZ", "02730ZZ", "02C00ZZ", "02C10ZZ", "02C20ZZ", "02C30ZZ",
"3E07017", "3E070PZ", "3E07317", "3E073PZ", "02C03ZZ", "02C04ZZ", "02C13ZZ",
"02C14ZZ", "02C23ZZ", "02C24ZZ", "02C33ZZ", "02C34ZZ", "02700ZZ", "02710ZZ",
"02720ZZ", "02730ZZ", "02C00ZZ", "02C10ZZ", "02C20ZZ", "02C30ZZ", "3E07017",
"3E070PZ", "3E07317", "3E073PZ", "02C03ZZ", "02C04ZZ", "02C13ZZ", "02C14ZZ",
"02C23ZZ", "02C24ZZ", "02C33ZZ", "02C34ZZ", "02700ZZ", "02710ZZ", "02720ZZ",
"02730ZZ", "02C00ZZ", "02C10ZZ", "02C20ZZ", "02C30ZZ", "3E07017", "3E070PZ",
"3E07317", "3E073PZ", "02C03ZZ", "02C04ZZ", "02C13ZZ", "02C14ZZ", "02C23ZZ",
"02C24ZZ", "02C33ZZ", "02C34ZZ", "02700ZZ", "02710ZZ", "02720ZZ", "02730ZZ",
"02C00ZZ", "02C10ZZ", "02C20ZZ", "02C30ZZ", "3E07017", "3E070PZ", "3E07317",
"3E073PZ", "02C03ZZ", "02C04ZZ", "02C13ZZ", "02C14ZZ", "02C23ZZ", "02C24ZZ",
"02C33ZZ", "02C34ZZ",

```