# -\*- coding: utf-8 -\*-# -\*- coding: utf-8 -\*-

import scrapy

import json

class PiplComSpider(scrapy.Spider):

name = 'pipl\_com'

allowed\_domains = ['pipl.com']

def start\_requests(self):

ids = [IDS GO HERE

]

for id in ids:

url = 'https://api.pipl.com/search/?user\_id=' + id + '@facebook&key=r9kk2uozkanlebdjfenp6bud'

yield scrapy.Request(url=url, callback=self.parse, meta={

'user\_id': id,

})

def parse(self, response):

json\_data = json.loads(response.text)

itm = {}

gender = None

email = None

address\_country = None

phone = None

username = None

name = None

if 'person' in json\_data:

if 'gender' in json\_data['person']:

gender = json\_data['person']['gender']['content']

if 'emails' in json\_data['person'] and len(json\_data['person']['emails']) > 0:

email = json\_data['person']['emails'][0]['address']

if 'names' in json\_data['person'] and len(json\_data['person']['names']) > 0:

name = json\_data['person']['names'][0]['display']

if 'phones' in json\_data['person'] and len(json\_data['person']['phones']) > 0:

phone = json\_data['person']['phones'][0]['display\_international']

if 'addresses' in json\_data['person'] and len(json\_data['person']['addresses']) > 0:

address\_country = json\_data['person']['addresses'][0]['country']

if 'usernames' in json\_data['person'] and len(json\_data['person']['usernames']) > 0:

username = json\_data['person']['usernames'][0]['content']

if gender == 'female' and address\_country == 'US' and email is not None:

itm['user\_id'] = response.meta.get('user\_id')

itm['gender'] = gender

itm['address\_country'] = address\_country

itm['name'] = name

itm['phone'] = phone

itm['username'] = username

itm['email'] = email

yield itm

import scrapy

import json

class PiplComSpider(scrapy.Spider):

name = 'pipl\_com'

allowed\_domains = ['pipl.com']

def start\_requests(self):

ids = [IDS GO HERE

]

for id in ids:

url = 'https://api.pipl.com/search/?user\_id=' + id + '@facebook&key=r9kk2uozkanlebdjfenp6bud'

yield scrapy.Request(url=url, callback=self.parse, meta={

'user\_id': id,

})

def parse(self, response):

json\_data = json.loads(response.text)

itm = {}

gender = None

email = None

address\_country = None

phone = None

username = None

name = None

if 'person' in json\_data:

if 'gender' in json\_data['person']:

gender = json\_data['person']['gender']['content']

if 'emails' in json\_data['person'] and len(json\_data['person']['emails']) > 0:

email = json\_data['person']['emails'][0]['address']

if 'names' in json\_data['person'] and len(json\_data['person']['names']) > 0:

name = json\_data['person']['names'][0]['display']

if 'phones' in json\_data['person'] and len(json\_data['person']['phones']) > 0:

phone = json\_data['person']['phones'][0]['display\_international']

if 'addresses' in json\_data['person'] and len(json\_data['person']['addresses']) > 0:

address\_country = json\_data['person']['addresses'][0]['country']

if 'usernames' in json\_data['person'] and len(json\_data['person']['usernames']) > 0:

username = json\_data['person']['usernames'][0]['content']

if gender == 'female' and address\_country == 'US' and email is not None:

itm['user\_id'] = response.meta.get('user\_id')

itm['gender'] = gender

itm['address\_country'] = address\_country

itm['name'] = name

itm['phone'] = phone

itm['username'] = username

itm['email'] = email

yield itm