

ДЕТСКАЯ ОНЛАЙН-ШКОЛА ПРОГРАММИРОВАНИЯ

HELLO WORLD



#ШПАРГАЛОЧКИ



ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ PYTHON

Средний уровень

Материалы подготовлены отделом методической разработки

Больше полезных материалов и общения в нашем комьюнити в Telegram: https://t.me/hw_school



Р.3.8. Бот-игра “Города”. Геоданные

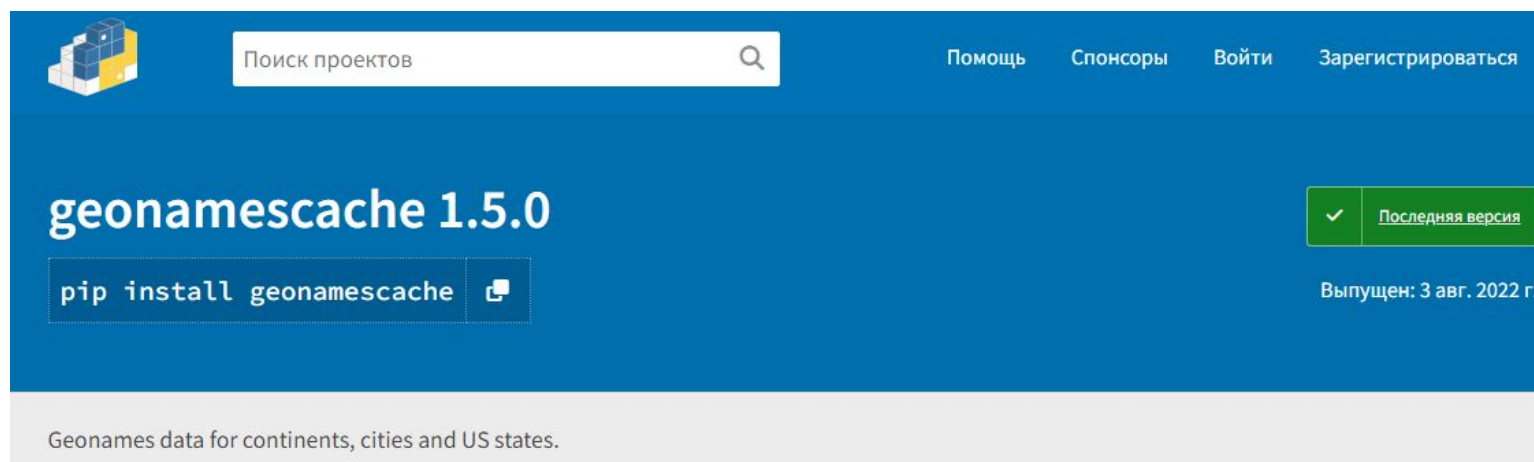


Библиотека для получения геоданных

Библиотека **geonamescache** предоставляет функции для получения названий, континентов, стран, а также штатов и округов США в виде словарей Python. Наборы данных по странам и городам также включают данные о населении и географические данные.

Ссылка на описание <https://pypi.org/project/geonamescache/>

Команда для установки **pip install geonamescache**





Получение списка всех городов

`import geonamescache` # импорт библиотеки

`geoNamesCache = geonamescache.GeonamesCache()` # создание объекта

`all_cities = geoNamesCache.get_cities()` # метод получения информации о городах

`print(all_cities)` # словарь, ключ: id города, значение: словари с информацией о городе

```
{'3040051': {'geonameid': 3040051, 'name': 'les Escaldes', 'latitude': 42.50729, 'longitude': 1.53414, 'countrycode': 'AD', 'population': 15853, 'timezone': 'Europe/Andorra', 'admin1code': '08', 'alternatenames': ['Ehskal'des-Ehndzhordani', 'Escaldes', 'Escaldes-Engordany', 'Les Escaldes', 'esukarudesu=engorudani jiao qu', 'lai sai si ka er de-en ge er da', 'Эскальдес-Энджордани', 'エスカルデス=エンゴルダニ教区', '萊塞斯卡爾德-恩戈爾達', '萊塞斯卡爾德-恩戈爾達']}, '3041563': {'geonameid': 3041563, 'name': 'Andorra la Vella', 'latitude': 42.50779, 'longitude': 1.52109, 'countrycode': 'AD', 'population': 20430, 'timezone': 'Europe/Andorra', 'admin1code': '07', 'alternatenames': ['ALV', 'Ando-la-Vyey', 'Andora', 'Andora la Vela', 'Andora la Velja', 'Andora lja Vehl'ja', 'Andoro Malnova', 'Andorra', 'Andorra Tuan', 'Andorra a Vella', 'Andorra la Biella', 'Andorra la Vella', 'Andorra la Vièlha', 'Andorra-a-Velha', 'Andorra-la-Vel'ja', 'Andorra-la-Vielye', 'Andorre-la-Vieille', 'Andò-la-Vyèy', 'An dòrra la Vièlha', 'an dao er cheng', 'andolalabeya', 'andwra la fyla', 'Ανδόρρα', 'Андора ла Веля', 'Андора ла Велья', 'Андора ла Вэлья', 'Андорра-ла-Велья', 'הלוו הלווה הלווה', 'اليف ال ارون', 'אנדורא-לא-וועליא', 'アンドラ・ラ・ヴェリヤ', '安道爾城', '안도라 라베야']}, '290594': {'geonameid': 290594, 'name': 'Umm Al Quwain City', 'latitude': 25.56473, 'longitude': 55.55517, 'countrycode': 'AE', 'population': 62747, 'timezone': 'Asia/Dubai', 'admin1code': '07', 'alternatenames': ['Oumm al Qaiwain', 'Oumm al Qaiwaïn', 'Um al Kawain', 'Um al Quweim', 'Umm Al Quwain City', 'Umm al Qaiwain', 'Umm al Qawain', 'Umm al Qaywayn', 'Umm al-Quwain', 'Umm-ehl'-Kajvaj', 'Yumul al Quwain', 'am alqywyn', 'mdynt am alqywyn', 'Умм-эль-Кайвайн', 'أمّ القيوين', 'امّ القيوين']}, '291074': {'geonameid': 291074, 'name': 'Ras Al Khaimah City', 'latitude': 25.78953, 'longitude': 55.9432, 'countrycode': 'AE', 'population': 351943, 'timezone': 'Asia/Dubai', 'admin1code': '05', 'alternatenames': ['Julf', 'Khaimah', 'RAK C
```




```
user_cities = geoNamesCache.search_cities(city) # метод поиска города по названию
print(user_cities ) # список словарей с информацией о городах с таким названием
```

```
[{'geonameid': 524901, 'name': 'Moscow', 'latitude': 55.75222, 'longitude': 37.61556, 'countrycode': 'RU', 'population': 10381222, 'timezone': 'Europe/Moscow', 'admin1code': '48', 'alternatenames': ['MOW', 'Maeskuy', 'Maskav', 'Maskava', 'Maskva', 'Mat-xco-va', 'Matxcova', 'Matxcova', 'Mosca', 'Moscfā', 'Moscha', 'Mosco', 'Moscou', 'Moscova', 'Moscovo', 'Moscow', 'Moscop', 'Moscu', 'Moscua', 'Moscū', 'Moscó', 'Moscù', 'Moscú', 'Moskva', 'Moska', 'Moskau', 'Mosko', 'Moskokh', 'Moskou', 'Moskov', 'Moskova', 'Moskovu', 'Moskow', 'Moskowa', 'Mosku', 'Moskuas', 'Moskva', 'Moskvo', 'Moskwa', 'Moszkva', 'Muskav', 'Musko', 'Mát-xcō-va', 'Mòskwa', 'Məskeu', 'Məskəy', 'masko', 'maskw', 'mo si ke', 'moseukeuba', 'mosko', 'mosukuwa', 'mskw', 'mwskva', 'mwskw', 'mwsqbh', 'mx s ko', 'Μόσχα', 'Ма́скуы', 'Маскав', 'Масква', 'Москва', 'Москова', 'Москох', 'Москъва', 'Му́скав', 'Му́ско', 'Mäскей', 'Mäскәй', 'ᲤᲗᲘᲠ', 'מאָסקווע', 'מוסקווע', 'הבִּקְסוּם', 'وکسمام', 'وکسم', 'وڪسوم', 'اؤكسووم', 'موسکو', 'मस्को', 'मुस्कौ', 'ਮਸਕੋ', 'ماسکوی', 'மோஸ்கோ', 'മോസ്കോ', 'モスクワ', '莫斯科', '모스크바']}, {'geonameid': 5601538, 'name': 'Moscow', 'latitude': 46.73239, 'longitude': -117.00017, 'countrycode': 'US', 'population': 25060, 'timezone': 'America/Los_Angeles', 'admin1code': 'ID', 'alternatenames': ['Hog Heaven', 'Moscow', 'Moskou', 'Moskov', 'Moskva', 'Paradise Valley', "S'maqw'l", 'masko', 'moseuko', 'mosuko', 'mskw aydahw', 'mwskw', 'Москва', 'Москов', 'Mockoy', 'وه اڏى آء وڪسم', 'وه اڏى آء وڪسموم', 'मस्को', 'მოსკო', 'モスコー', 'моско']]
```



Отрицательные индексы в списках и строках

В Python можно использовать **отрицательные индексы** для получения элементов списка или символов строки:

-1 - это **последний** элемент,
-2 - **предпоследний** элемент
и т.д., как будто перебираешь с конца.

0	1	...		-2	-1
М	о	с	к	в	а

```
city = "Москва"
print(city[-1]) # буква а
print(city[-2]) # буква в
```

```
cities = ["Москва", "Берлин", "Амстердам", "Дели"]
print(cities [-1]) # Дели
print(cities [-2]) # Амстредам
```



Заглавные и строчные буквы

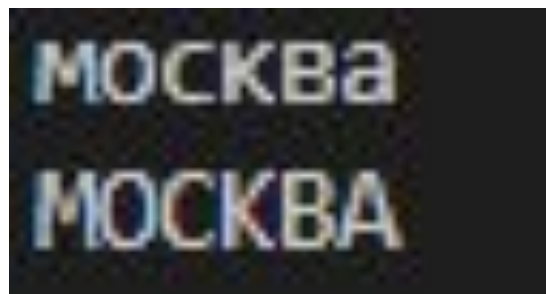
upper() - метод строки для преобразования в **заглавные** буквы

lower() - метод строки для преобразования в **строчные** буквы

```
test_str = "Москва"
```

```
print(test_str.lower()) # москва
```

```
print(test_str.upper()) # МОСКВА
```





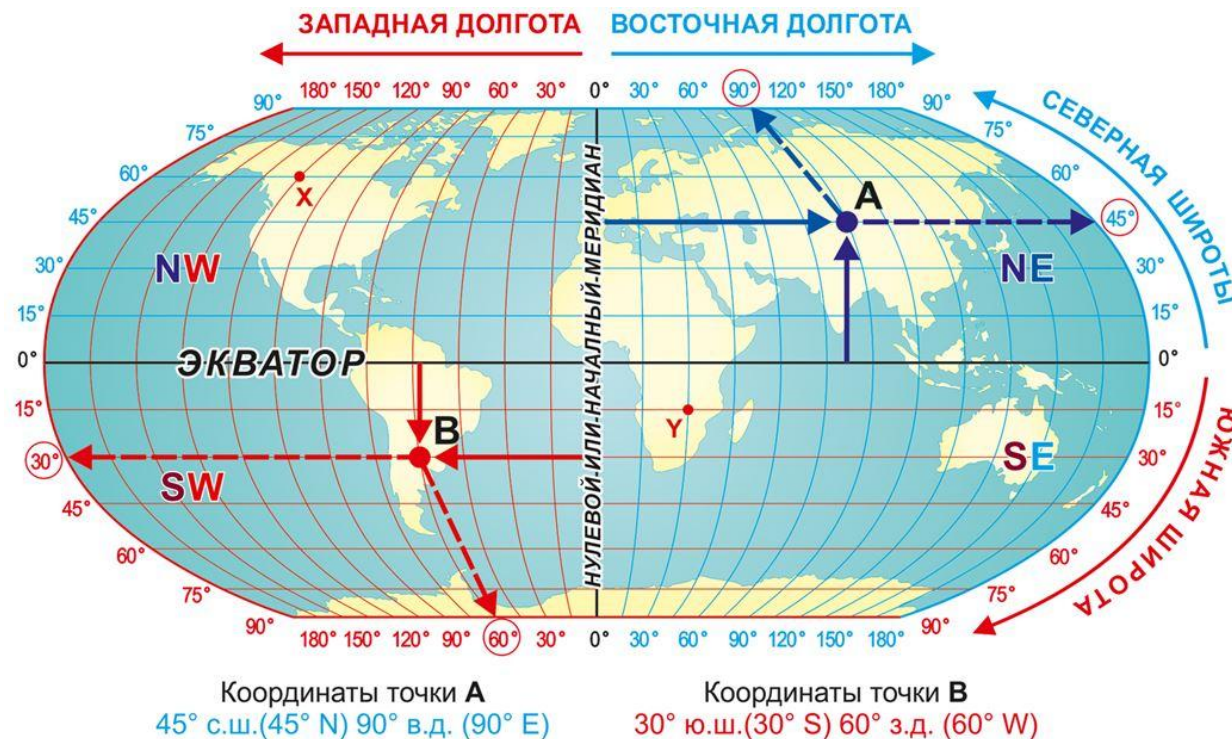
Географическая широта и долгота

На поверхности земного шара проведены воображаемые линии-окружности, параллельные экватору, они называются **параллелями**. Они аналогичны координатной сетке по x.

Кроме того по поверхности земного шара проведены линии, соединяющие северный и южный полюс, они называются **меридианами**. Они аналогичны координатной сетке по y.

Отсчет широты идет от **экватора**, а долготы от **нулевого (гринвичского) меридиана** и производится в **градусах**.

Меридианы и параллели переносятся и на географическую карту.





Географическая широта

Географическая широта — величина дуги меридиана в градусах от экватора до заданной точки.

Все пункты, лежащие на одной параллели, имеют одинаковую широту.

Географическую широту отсчитывают от **экватора**, поэтому его широта — **0° ш.**
Значения широты на полюсах — **90° ш.**



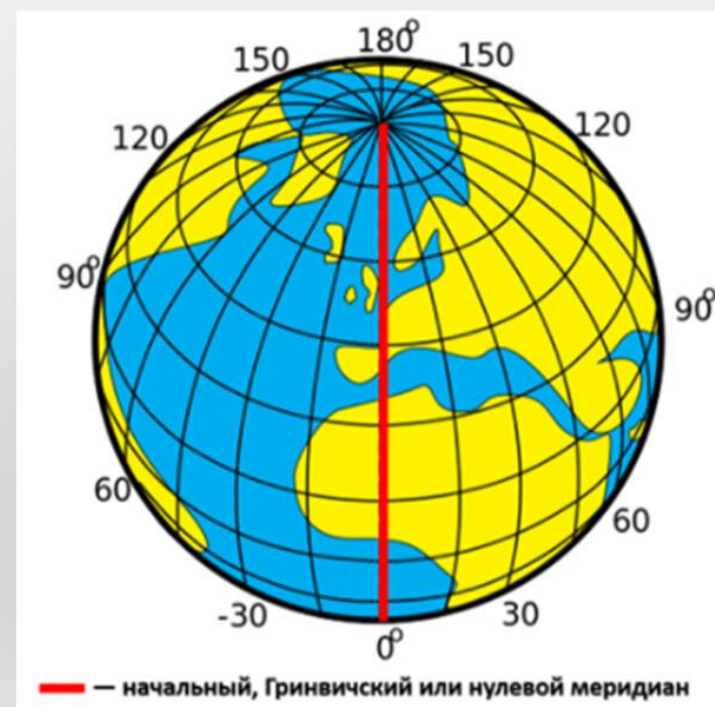


Географическая долгота

Географическая долгота — величина дуги параллели в градусах от начального меридиана до заданной точки.

Все точки, лежащие на одном меридиане, имеют одинаковую долготу.

Поскольку долготу отсчитывают от начального (Гринвичского) меридиана, то все точки, лежащие на нём, имеют долготу 0° д. Поэтому начальный меридиан имеет ещё одно название — нулевой.





Метод отправки геолокации

`bot.send_location(message.chat.id, latitude, longitude)` # метод отправки геолокации принимает 3 обязательных параметра: id чата, широту и долготу

