


ХОЧУ ПОМОЧЬ
ПРОЕКТУ




practicum.yandex.ru

Бесплатный курс «Основы работы с базами данных и SQL»

Освойте основы SQL и сможете решить первые задачи в интерактивном тренажёре.


Узнать больше

РЕКЛАМА · 16+ 

Страница 3. / Модуль user_agents в Python

Парсинг пользовательского агента

Модуль `user_agents` предоставляет простой способ идентифицировать/обнаруживать устройства, такие как мобильные телефоны, планшеты и ПК. Он использует синтаксический анализ (браузер/HTTP) строк пользовательского агента. Цель состоит в том, чтобы:



- Определить тип устройства (браузера и его версию, версию операционной системы).
- Что представляет из себя *user agent*: мобильное устройство, планшет или ПК.
- Имеет ли *user agent* сенсорные возможности (имеет сенсорный экран).

Модуль `user_agents` полагается на отличный `ua-parser` для фактического синтаксического анализа необработанной строки пользовательского агента (*user agent*).

Установка модуля user_agents в виртуальное окружение:

```
# создаем виртуальное окружение
$ python3 -m venv .venv --prompt VirtualEnv
# активируем виртуальное окружение
$ source .venv/bin/activate
# ставим модуль user_agents
(VirtualEnv) Idea@Centre:~$ python -m pip install -U user_agents
```

Доступ к атрибутам браузера пользователя.

К различной базовой информации, которая может помочь идентифицировать посетителей, можно получить доступ к атрибутам браузера, устройства и ОС.

Примеры.

```
# Строка пользовательского агента iPhone
>>> ua_string = """Mozilla/5.0 (iPhone; CPU iPhone OS 5_1 like Mac OS X) AppleWebKit/534.46
(KHTML, like Gecko) Version/5.1 Mobile/9B179 Safari/7534.48.3"""

# Импортируем установленный модуль
>>> from user_agents import parse
# передаем строку пользовательского агента парсеру
>>> user_agent = parse(ua_string)
```

Доступ к атрибутам браузера:

```
>>> user_agent.browser
# Browser(family='Mobile Safari', version=(5, 1), version_string='5.1')
>>> user_agent.browser.family
# 'Mobile Safari'
>>> user_agent.browser.version
# (5, 1)
>>> user_agent.browser.version_string
# '5.1'
```

Доступ к свойствам операционной системы:

```
>>> user_agent.os
# OperatingSystem(family='iOS', version=(5, 1), version_string='5.1')
>>> user_agent.os.family
# 'iOS'
```

https://docs-python.ru/packages/modul-user-agents-python/

1/3

```
>>> user_agent.os.version
# (5, 1)
>>> user_agent.os.version_string
# '5.1'
```

Доступ к свойствам устройства:

```
>>> user_agent.device
# Device(family='iPhone', brand='Apple', model='iPhone')
>>> user_agent.device.family
# 'iPhone'
>>> user_agent.device.brand
# 'Apple'
>>> user_agent.device.model
# 'iPhone'

# Viewing a pretty string version
>>> str(user_agent)
# "iPhone / iOS 5.1 / Mobile Safari 5.1"
```

Определение класса устройства пользователя и его возможностей.

Модуль user_agents также предоставляет несколько других, более "сложных" атрибутов, которые являются производными от одного или нескольких основных атрибутов, определенных выше. На данный момент эти атрибуты должны правильно определять популярные платформы/устройства.

В настоящее время поддерживаются следующие атрибуты:

- is_mobile: идентифицируется ли пользовательский агент как мобильный телефон (iPhone, телефоны Android, устройства BlackBerry, Windows Phone и т. д.),
- is_tablet: идентифицируется ли пользовательский агент как планшет (iPad, Kindle Fire, Nexus 7 и т. д.),
- is_pc: идентифицирован ли пользовательский агент как работающий с традиционной "настольной" ОС (Windows, OS X, Linux),
- is_touch_capable: есть ли у пользовательского агента сенсорные возможности,
- is_bot: является ли пользовательский агент сканером/пауком/ботом.

Примеры.

Старое устройство BlackBerry без сенсорного управления:

```
>>> from user_agents import parse
>>> ua_string = """BlackBerry9700/5.0.0.862 Profile/MIDP-2.1 Configuration/CLDC-1.1
VendorID/331 UNTRUSTED/1.0 3gpp-gba"""
>>> user_agent = parse(ua_string)
>>> user_agent.is_mobile
# True
>>> user_agent.is_tablet
# False
>>> user_agent.is_touch_capable
# False
>>> user_agent.is_pc
# False
>>> user_agent.is_bot
# False
>>> str(user_agent)
# "BlackBerry 9700 / BlackBerry OS 5 / BlackBerry 9700"
```

Строка пользовательского агента Samsung Galaxy S3:

```
>>> from user_agents import parse
>>> ua_string = """Mozilla/5.0 (Linux; U; Android 4.0.4; en-gb; GT-I9300 Build/IMM76D)
AppleWebKit/534.30 (KHTML, like Gecko) Version/4.0 Mobile Safari/534.30"""
>>> user_agent = parse(ua_string)
>>> user_agent.is_mobile
# True
>>> user_agent.is_tablet
# False
>>> user_agent.is_touch_capable
# True
>>> user_agent.is_pc
# False
```

Вверх

```
>>> user_agent.is_bot
# False
>>> str(user_agent)
# "Samsung GT-I9300 / Android 4.0.4 / Android 4.0.4"
```

Строка пользовательского агента iPad:

```
>>> from user_agents import parse
ua_string = """Mozilla/5.0(iPad; U; CPU iPhone OS 3_2 like Mac OS X; en-us) AppleWebKit/531.21.10
(KHTML, like Gecko) Version/4.0.4 Mobile/7B314 Safari/531.21.10"""
>>> user_agent = parse(ua_string)
>>> user_agent.is_mobile
# False
>>> user_agent.is_tablet
# True
>>> user_agent.is_touch_capable
# True
>>> user_agent.is_pc
# False
>>> user_agent.is_bot
# False
>>> str(user_agent)
# "iPad / iOS 3.2 / Mobile Safari 4.0.4"
```

Строка пользовательского агента Kindle Fire:

```
>>> from user_agents import parse
>>> ua_string = """Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_6_3; en-us; Silk/1.1.0-80)
AppleWebKit/533.16 (KHTML, like Gecko) Version/5.0 Safari/533.16 Silk-Accelerated=true"""
>>> user_agent = parse(ua_string)
>>> user_agent.is_mobile
# False
>>> user_agent.is_tablet
# True
>>> user_agent.is_touch_capable
# True
>>> user_agent.is_pc
# False
>>> user_agent.is_bot
# False
>>> str(user_agent)
# "Kindle / Android / Amazon Silk 1.1.0-80"
```

Пользовательский агент устройства с ОС Windows 8 с сенсорным экраном:

```
>>> from user_agents import parse
>>> ua_string = 'Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.2; Trident/6.0; Touch)'
>>> user_agent = parse(ua_string)
>>> user_agent.is_mobile
# False
>>> user_agent.is_tablet
# False
>>> user_agent.is_touch_capable
# True
>>> user_agent.is_pc
# True
>>> user_agent.is_bot
# False
>>> str(user_agent)
# "PC / Windows 8 / IE 10"
```