

API сервиса Практикум.Домашка (Шпаргалка)

На серверах Практикума есть API, через который можно отслеживать изменение статуса вашей домашней работы, отправленной на ревью.

У API Практикум.Домашка есть лишь один эндпоинт:

https://practicum.yandex.ru/api/user_api/homework_statuses/ и доступ к нему возможен только по токену.

Получить токен можно по адресу: https://oauth.yandex.ru/authorize?response_type=token&client_id=1d0b9dd4d652455a9eb710d450ff456a.

Принцип работы API

Когда ревьюер проверяет вашу домашнюю работу, он присваивает ей один из статусов:

- работа принята на проверку,
- работа возвращена для исправления ошибок,
- работа принята.

С помощью API можно получить список ваших домашних работ, статус которых изменился за период от `from_date` до настоящего момента.

Для успешного запроса нужно:

- в заголовке запроса передать токен авторизации `Authorization: OAuth <token>` ;
- в GET-парамetre `from_date` передать метку времени в формате *Unix time*.

Формат времени *Unix time* — это стандарт в программном исчислении времени. Время определяется количеством секунд, прошедших с 1 января 1970 года (эту дату также называют «началом эпохи Unix»).

Например, Яндекс.Практикум открылся для студентов 12 февраля 2019 года (предположим, что двери открылись в полдень по московскому времени). В

формате Unix time этот момент времени будет определён как `1549962000`.

Пример запроса

В директории с проектами создайте папку `/homework_api`. В ней создайте файл `api_praktikum.py` и добавьте в него следующий код:

```
# homework_api/api_praktikum.py

import requests

url = 'https://practicum.yandex.ru/api/user_api/homework_statuses/'
headers = {'Authorization': f'OAuth {<ваш токен>}' }
payload = {'from_date': <временная метка в формате Unix time>}.

# Делаем GET-запрос к эндпоинту url с заголовком headers и параметрами params
homework_statuses = requests.get(url, headers=headers, params=payload)

# Печатаем ответ API в формате JSON
print(homework_statuses.text)

# А можно ответ в формате JSON привести к типам данных Python и напечатать и его
# print(homework_statuses.json())
```

Если в запросе переданы валидный токен и временная метка, то после выполнения программы в терминале будет напечатан ответ API.

Примеры ответов API

API Практикум.Домашка возвращает ответы в формате JSON. В ответе содержатся два ключа:

- `homeworks`: значение этого ключа — список домашних работ;
- `current_date`: значение этого ключа — время отправки ответа.

Ответ API содержит актуальный статус каждой домашней работы. В список попадают работы, которым был присвоен статус за период от `from_date` до настоящего момента. Следовательно, с помощью метки времени можно управлять содержанием этого списка:

- при `from_date = 0` в этот список попадут все ваши домашние работы;
- при `from_date`, равном «минуту назад», велик шанс получить пустой список `homeworks`;

- при других значениях этого параметра в списке будет ограниченный перечень домашних работ.

При запросе с параметром `from_date = 0` API вернёт список домашек за всё время:

```
{
  "homeworks": [
    {
      "id": 124,
      "status": "rejected",
      "homework_name": "username__hw_python_oop.zip",
      "reviewer_comment": "Код не по PEP8, нужно исправить",
      "date_updated": "2020-02-13T16:42:47Z",
      "lesson_name": "Итоговый проект"
    },
    {
      "id": 123,
      "status": "approved",
      "homework_name": "username__hw_test.zip",
      "reviewer_comment": "Всё нравится",
      "date_updated": "2020-02-11T14:40:57Z",
      "lesson_name": "Тестовый проект"
    },
    ...
  ],
  "current_date": 1581604970
}
```

Если за выбранный интервал времени ни у одной из домашних работ не появился новый статус — список работ будет пуст:

```
{"homeworks": [], "current_date": 1634074965}
```

При периодических запросах к API можно использовать значение, полученное под ключом `current_date`, в качестве параметра `from_date` в следующем запросе.

Статусы домашней работы

Статус домашки (значение по ключу `status`) может быть трёх типов:

- **reviewing**: работа взята в ревью;

- **approved**: ревью успешно пройдено;
- **rejected**: в работе есть ошибки, нужно поправить.

Если домашку ещё не взяли в работу — её не будет в выдаче.