

Автоматизация тестирования веб-приложений на Python

Лекция 1

Тестирование API в Python. DDT.





Знакомство и содержание урока





Сергей Куриленко

Руководитель группы тестирования в области
информационной безопасности



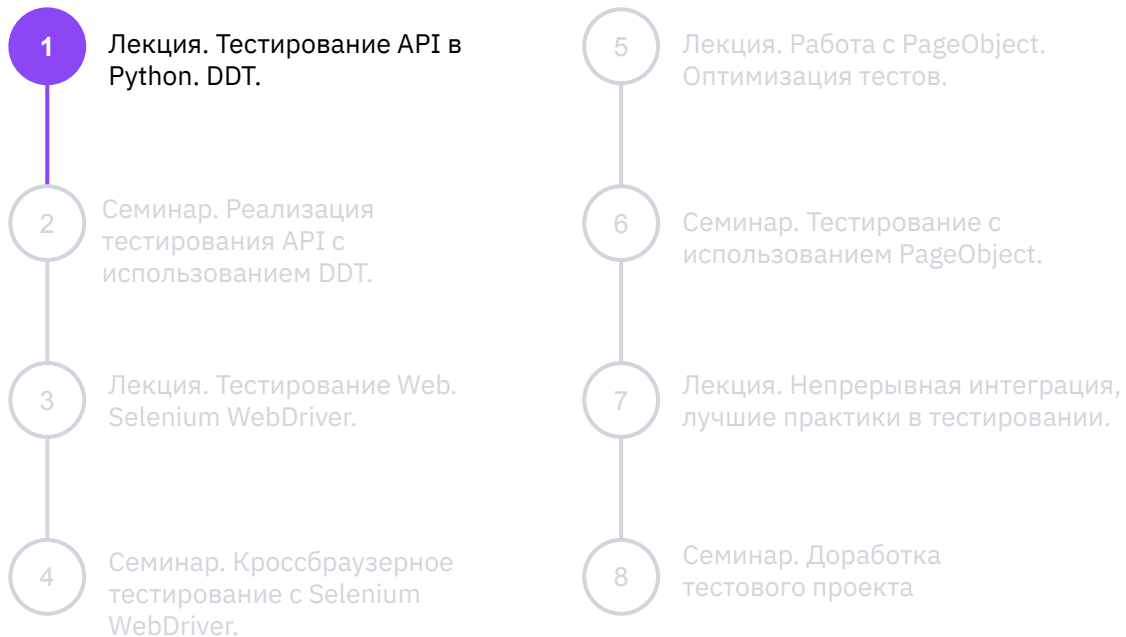
В тестировании с 2018 года



Занимаюсь автоматизацией на Python проектов,
связанных с информационной безопасностью
и электронным документооборотом



План курса



Что будет на уроке сегодня

- 📌 Особенности тестирования API, UI и UX
- 📌 Как автоматизировать запросы к REST API
- 📌 Как автоматизировать запросы к SOAP API
- 📌 Что такое DDT





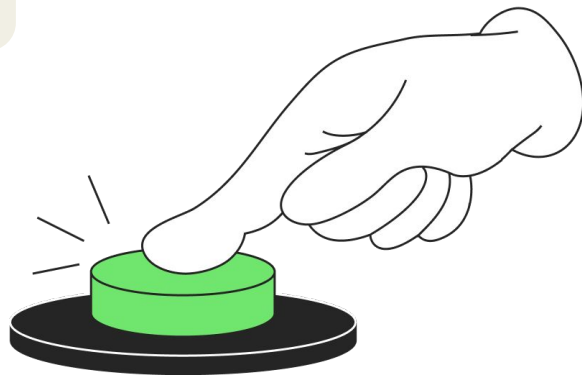
Поехали!



UI-тестирование

UI-тестирование — это этап в процессе тестирования ПО, проверяющий графический интерфейс.

Тестирование интерфейса помогает нам проверить функции приложения, имитируя действия пользователей.





UX-тестирование

Для того чтобы проверить насколько качественно разработан интерфейс приложения и насколько он удобен пользователю, проводят **тестирование на юзабилити** или **UX тестирование**.

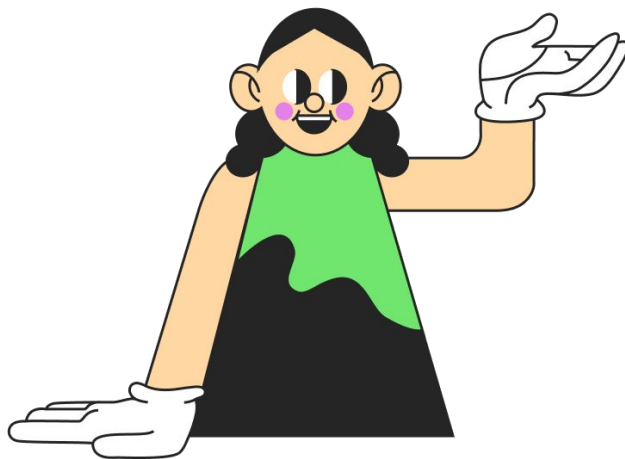
UX тестирование проводится для того, чтобы понять, насколько вы хорошо, понятно, логично, удобно, правильно разработали свой ресурс, работают ли на нем все разработанные вами элементы и функции.



Тестирование API

API — это интерфейс, позволяющий двум независимым компонентам программного обеспечения обмениваться информацией.

Не у всех веб-приложений имеется доступный для тестирования API, но там, где он есть, его **обязательно** нужно протестировать.





Вопрос

Как вы думаете, какой из этих видов тестирования лучше поддается автоматизации?

Расставьте их в порядке усложнения автоматизации.

Предоставьте ваши ответы в комментариях к видео.
Время на размышление — 1 минута.





Ответ





REST и SOAP API





Сравнение REST и SOAP API

SOAP API	REST API
Использование XML, WSDL	Ресурс ориентированная технология
Работа с методами	HTTP запросы
Поддержка транзакций и уровней безопасности	Для несложной бизнес-модели
Сложнее разрабатывать	Легче разрабатывать



Как мы будем тестировать API в Python?



Библиотека Requests





Установка

```
pip3 install requests
```



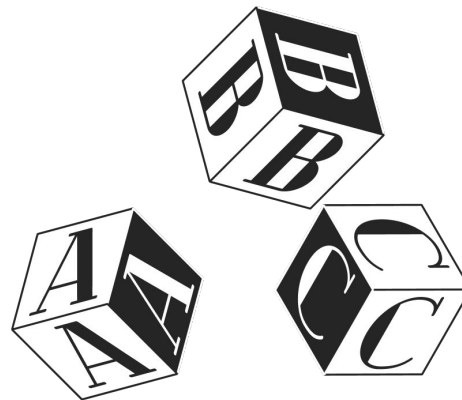
Пример запроса

```
response = requests.get('https://api.github.com')
```



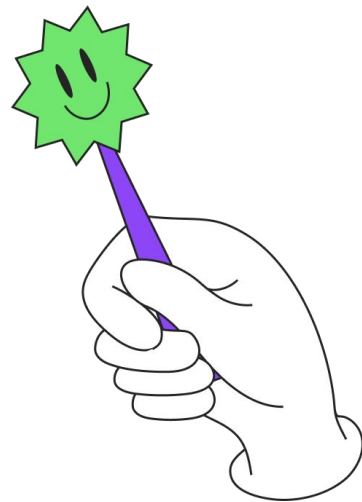
Код ответа

```
response.status_code
```



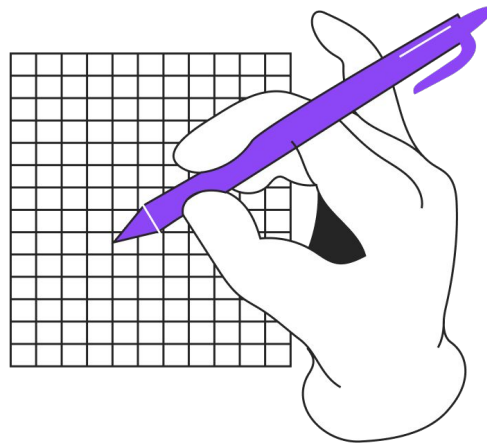
Примеры запросов:

- ✓ `requests.post('https://httpbin.org/post', data={'key':'value'})`
- ✓ `requests.put('https://httpbin.org/put', data={'key':'value'})`
- ✓ `requests.delete('https://httpbin.org/delete')`
- ✓ `requests.head('https://httpbin.org/get')`
- ✓ `requests.patch('https://httpbin.org/patch', data={'key':'value'})`
- ✓ `requests.options('https://httpbin.org/get')`



Получение содержимого ответа:

- ✓ `response.content`
- ✓ `response.text`
- ✓ `response.json()`





HEADERS

```
1 import requests
2
3 response = requests.get(
4     'https://api.github.com/search/repositories',
5     params={'q': 'requests+language:python'},
6     headers={'Accept': 'application/vnd.github.v3.text-match+json'},
7 )
```



Авторизация

```
1 from getpass import getpass
2 requests.get('https://api.github.com/user', auth=('username',
  getpass()))
```

```
1 import requests
2 headers = {'X-Yandex-API-Key': 'b1fb281f-
  b5ff-4257-8b94-35f249384a32'}
3 data = requests.get('https://api.weather.yandex.ru
  /v1/forecast?lat=55.466&lon=36.93&extra=true', headers=headers)
4 print (data.text)
```



Работа с wiki geosearch



Библиотека zeep для SOAP





Установка



`pip3 install zeep`

```
1 from zeep import Client
2
3 client = Client('http://www.webservices.net/ConvertSpeed.asmx?WSDL')
4 result = client.service.ConvertSpeed(
5     100, 'kilometersPerhour', 'milesPerhour')
6
7 assert result == 62.137
```



Вопрос

Что происходит в этом примере?

Предоставьте ваши ответы в комментариях к видео.
Время на размышление — 2 минуты.





Ответ

Перевод километров в час
в мили в час



Описание структуры

```
-<definitions name="SoapResponder" targetNamespace="http://www.SoapClient.com/xml/SoapResponder.wsdl">
  -<types>
    <schema targetNamespace="http://www.SoapClient.com/xml/SoapResponder.xsd"> </schema>
  </types>
  -<message name="Method1">
    <part name="bstrParam1" type="xsd:string"/>
    <part name="bstrParam2" type="xsd:string"/>
  </message>
  -<message name="Method1Response">
    <part name="bstrReturn" type="xsd:string"/>
  </message>
  -<portType name="SoapResponderPortType">
    -<operation name="Method1" parameterOrder="bstrparam1 bstrparam2 return">
      <input message="tns:Method1"/>
      <output message="tns:Method1Response"/>
    </operation>
  </portType>
```



Получение структуры

```
1 python3 -m zeep http://www.soapclient.com/xml/soapresponder.wsdl
```

Bindings:

Soap11Binding: {<http://www.SoapClient.com/xml/SoapResponder.wsdl>}SoapResponderBinding

Service: SoapResponder

Port: SoapResponderPortType (Soap11Binding: {<http://www.SoapClient.com/xml/SoapResponder.wsdl>}SoapResponderBinding)

Operations:

Method1(bstrParam1: xsd:string, bstrParam2: xsd:string) -> bstrReturn: xsd:string



Ослабление требований к ответу

```
1 settings = Settings(strict=False)
2 client = Client(wsdl=wsdl, settings=settings)
```



Проверка ЭЦП в SOAP



Что такое DDT





Data Driven Testing (DDT)

Подход к созданию/архитектуре автоматизированных тестов (юнит, интеграционных, чаще всего применимо к backend тестированию), при котором **тест умеет** принимать набор входных параметров, и эталонный результат или эталонное состояние, **с которым он должен сравнить результат**, полученный в ходе прогонки входных параметров.

Такое сравнение и есть assert такого теста.

Притом как часть входных параметров, могут передаваться опции выполнения теста, или флаги, которые влияют на его логику.





Реализация DDT



Что мы узнали сегодня

- 📌 Особенности тестирования API, UI и UX
- 📌 Как автоматизировать запросы к REST API
- 📌 Как автоматизировать запросы к SOAP API
- 📌 Что такое DDT





Спасибо за внимание!