

# Автоматизатор тестирования на Python

---

5 месяцев

продолжительность курса



## Бесплатная часть

---

00

[2 часа]

Познакомьтесь с форматом обучения: оцените подачу материала и тренажёр, в котором будете писать код. Сможете понять, хотите ли развиваться как инженер по тестированию.

## Темы

---

1. Введение в автоматизацию
2. Как устроен курс

## [2 недели]

Решите несколько комплексных задач на Python. Напишите программы, которые печатают текст, считают время и составляют таблицы.

Познакомьтесь с базовыми конструкциями языка: классами и объектами, функциями, ветвлениями и так далее. Будете сразу применять знания: писать код внутри уроков и в среде разработки PyCharm.

## Темы

1. Знакомство с Python

2. Ветвления

3. Списки

4. Циклы

5. Операции со строками

6. Словари и коллекции

7. Функции

8. Классы и объекты

9. PyCharm

# ООП: наследование, полиморфизм

## [2 недели]

С помощью принципов ООП решите несколько комплексных задач на Python.

Изучите основы объектно-ориентированного программирования (ООП). Это принципы, которые помогают решать сложные задачи, настраивать методы и классы, создавать логичные иерархии. ООП помогает писать программы, которые легко поддерживать и проверять.

## Темы

1. Наследование

2. Полиморфизм

3. Статические методы

4. Методы класса

5. Области видимости

## [2 недели]

Напишите консольное приложение с помощью ООП и PyCharm.

Продолжите изучать ООП. После этого модуля сможете решать комплексные сложные задачи в коде. Чтобы закрепить полученные навыки, напишете небольшое приложение.

## Темы

1. Лямбда-функции

2. Инкапсуляция

3. Обработка исключений

# Юнит-тесты

# 04

## [2 недели]

Покроете юнит-тестами небольшое консольное приложение.

Освоите важную часть автоматического тестирования — фреймворк Pytest. Его используют, чтобы не писать каждый раз код с нуля. Начнёте работать в Pytest сначала с самыми простыми тестами — юнит-тестами. Потом узнаете, как в Pytest тестировать веб.

## Темы

1. Консоль

2. Git

3. Основы юнит-тестов

4. Pytest

5. Ассерты

6. Параметризация

7. Фикстуры

8. Оценка покрытия

## [2 недели]

Протестируете UI-интерфейс сервиса заказа еды Stellar Burgers с помощью Selenium и Devtools. Плюс покроете тестами несколько веб-страниц.

Научитесь автоматизировать проверки, которые обычно проводятся вручную через браузер. Для этого изучите Selenium — с его помощью программы с автотестами «нажимают» на кнопки и «заполняют» поля ввода. Примените всё, что узнали о Python, для тестирования веб-приложений.

## Темы

1. Архитектура веб-приложений

2. Devtools

3. UI-тестирование

4. Selenium: поиск элементов

5. Selenium: базовые методы

# Page Object Model

# 06

## [2 недели]

Напишете тесты для веб-приложения с помощью Page Object Model. Построите информативный отчёт о тестировании в Allure.

Узнаете, как улучшать автотесты с помощью паттерна проектирования Page Object Model. Он помогает выстраивать правильную структуру тестов, чтобы их было легче читать и поддерживать. Построите отчёты о тестировании с помощью инструмента Allure, который предоставляет наглядную статистику о проверках.

## Темы

1. Page Object

2. Allure

3. Структура проекта

## [2 недели]

Протестируете API учебного веб-приложения.

Узнаете, как автоматизировать тестирование API. Научитесь применять Postman и Swagger для автоматизации, писать несколько видов запросов. Потренируетесь работать с моками и стабами — заглушками, которые имитируют поведение реальных сервисов. Например, с их помощью можно отправить письмо в автотесте, то есть не отправляя его на самом деле.

## Темы

1. Введение в API

3. Подготовка к автоматизации

4. Моки

2. Тестирование API с помощью Python

# Архитектура

## [2 недели]

Автоматизируете тестирование веб-приложения: UI-части и напишете юнит-тесты для бэкенда.

Мало покрыть приложение тестами — нужно позаботиться о том, чтобы процесс тестирования был исправен. Поэтому в этом модуле вы научитесь оценивать, насколько полно функциональность покрыта тестами. Это нужно, чтобы не было проблем при автоматизации.

## Темы

1. Архитектура

3. Мок-сервер для UI и API

5. Подготовка к итоговой работе

2. Покрытие тестами разных уровней программы

4. Рабочие задачи автоматизатора

[2 недели]

Это проект, который объединит весь изученный материал. Вам предстоит покрыть тестами веб-приложение, API и написать юнит-тесты на отдельную функциональность.

## Дополнительный модуль. База данных

[10 часов]

Узнаете, как устроен язык запросов SQL. Поймёте, как тестировать базы данных.

## Воркшопы

Воркшопы проводятся 1 раз за спринт. Они посвящены нюансам работы с кодом, инструментам автоматизации тестирования. На воркшопах вы сможете применить полученные знания на практике.

# Программа трудоустройства

---

[4 недели],  
[12 часов]

Программа становится доступна в конце обучения. В ней расскажем о том, как правильно составить резюме, создать портфолио и сделать свой отклик заметнее с помощью сопроводительного письма. Также обсудим процесс подготовки к собеседованию: на какие софт-скиллы и хард-скиллы сделать фокус.

# Автоматизатор тестирования на Python расширенный

---

6 месяцев

продолжительность курса



## Бесплатная часть

---

00

[2 часа]

Познакомитесь с форматом обучения: оцените подачу материала и тренажёр, в котором будете писать код. Сможете понять, хотите ли развиваться как инженер по тестированию.



**[2 недели],  
[проект]**

Решите несколько комплексных задач на Python. Напишите программы, которые печатают текст, считают время и составляют таблицы.

Познакомьтесь с базовыми конструкциями языка: классами и объектами, функциями, ветвлениями и так далее. Будете сразу применять знания: писать код внутри уроков и в среде разработки PyCharm.

## Темы

1. Знакомство с Python

2. Ветвления

3. Списки

4. Циклы

5. Операции со строками

6. Словари и коллекции

7. Функции

8. Классы и объекты

9. PyCharm

# ООП: наследование, полиморфизм

**[2 недели],  
[проект]**

С помощью принципов ООП решите несколько комплексных задач на Python.

Изучите основы объектно-ориентированного программирования (ООП). Это принципы, которые помогают решать сложные задачи, настраивать методы и классы, создавать логичные иерархии. ООП помогает писать программы, которые легко поддерживать и проверять.

## Темы

1. Наследование

2. Полиморфизм

3. Статические методы

4. Методы класса

5. Области видимости

[2 недели],  
[проект]

Напишете консольное приложение с помощью ООП и PyCharm.

Продолжите изучать ООП. После этого модуля сможете решать комплексные сложные задачи в коде. Чтобы закрепить полученные навыки, напишете небольшое приложение.

## Темы

1. Лямбда-функции

2. Инкапсуляция

3. Обработка исключений

# Юнит-тесты

# 04

[2 недели],  
[проект]

Покроете юнит-тестами небольшое консольное приложение.

Освоите важную часть автоматического тестирования — фреймворк Pytest. Его используют, чтобы не писать каждый раз код с нуля. Начнёте работать в Pytest сначала с самыми простыми тестами — юнит-тестами. Потом узнаете, как в Pytest тестировать веб.

## Темы

1. Консоль

2. Git

3. Основы юнит-тестов

4. Pytest

5. Ассерты

6. Параметризация

7. Фикстуры

8. Оценка покрытия

## [2 недели], [проект]

Протестируете UI-интерфейс сервиса заказа еды Stellar Burgers с помощью Selenium и Devtools. Плюс покроете тестами несколько веб-страниц.

Научитесь автоматизировать проверки, которые обычно проводятся вручную через браузер. Для этого изучите Selenium — с его помощью программы с автотестами «нажимают» на кнопки и «заполняют» поля ввода. Примените всё, что узнали о Python, для тестирования веб-приложений.

## Темы

1. Архитектура веб-приложений

2. Devtools

3. UI-тестирование

4. Selenium: поиск элементов

5. Selenium: базовые методы

# Page Object Model

# 06

## [2 недели], [проект]

Напишете тесты для веб-приложения с помощью Page Object Model. Построите информативный отчёт о тестировании в Allure.

Узнаете, как улучшать автотесты с помощью паттерна проектирования Page Object Model. Он помогает выстраивать правильную структуру тестов, чтобы их было легче читать и поддерживать. Построите отчёты о тестировании с помощью инструмента Allure, который предоставляет наглядную статистику о проверках.

## Темы

1. Page Object

2. Allure

3. Структура проекта

**[2 недели],  
[проект]**

Протестируете API учебного веб-приложения.

Узнаете, как автоматизировать тестирование API. Научитесь применять Postman и Swagger для автоматизации, писать несколько видов запросов. Потренируетесь работать с моками и стабами — заглушками, которые имитируют поведение реальных сервисов. Например, с их помощью можно отправить письмо в автотесте, то есть не отправляя его на самом деле.

## Темы

1. Введение в API

3. Подготовка к автоматизации

4. Моки

2. Тестирование API с помощью Python

# Архитектура

**[2 недели],  
[проект]**

Автоматизируете тестирование веб-приложения: UI-части и напишете юнит-тесты для бэкенда.

Мало покрыть приложение тестами — нужно позаботиться о том, чтобы процесс тестирования был исправен. Поэтому в этом модуле вы научитесь оценивать, насколько полно функциональность покрыта тестами. Это нужно, чтобы не было проблем при автоматизации.

## Темы

1. Архитектура

3. Мок-сервер для UI и API

5. Подготовка к итоговой работе

2. Покрытие тестами разных уровней программы

4. Рабочие задачи автоматизатора

[2 недели]

Это проект, который объединит весь изученный материал. Вам предстоит покрыть тестами веб-приложение, API и написать юнит-тесты на отдельную функциональность.

## Инфраструктура

# 10

[2 недели],  
[проект]

Узнаете, что такое контейнеризация и познакомитесь с процессами CI/CD.

Научитесь работать с Jenkins и Docker.

### Темы

1. Общие понятия и инструменты CI/CD

2. Работа с Jenkins на локальной машине

3. Работа с Jenkins на удалённом сервере

## Проектный месяц

# 11

[4 недели]

Сделаете два дополнительных комплексных проекта для отработки навыков тестирования.

# Дополнительный модуль. Базы данных

---

[10 часов]

Узнаете, как устроен язык запросов SQL. Поймёте, как тестировать базы данных.

## Воркшопы

---

Воркшопы проводятся 1 раз за спринт. Они посвящены нюансам работы с кодом, инструментам автоматизации тестирования. На воркшопах вы сможете применить полученные знания на практике.

## Индивидуальные онлайн-консультации

---

Восемь встреч с наставниками, на которых вы сможете обсудить вопросы по обучению, проконсультироваться по личным проектам и получить ответы на любые интересующие вас темы в сфере автоматизации.

## Программа трудоустройства

---

[4 недели],  
[12 часов]

Программа становится доступна в конце обучения. В ней расскажем о том, как правильно составить резюме, создать портфолио и сделать свой отклик заметнее с помощью сопроводительного письма. Также обсудим процесс подготовки к собеседованию: на какие софт-скиллы и хард-скиллы сделать фокус.

---