



# Java QA Engineer. Basic

Тестировщик-автоматизатор на Java. Базовый курс

Длительность курса: 108 академических часов

# 1 Знакомство с тестированием

## 1 Знакомство

### Цели занятия:

понять текущее состояние рынка IT, нишу Java QA.

### Краткое содержание:

что такое IT;  
направления в IT (администрирование, разработка, менеджмент, автоматизация процессов, дизайн, аналитика и т.д.);  
Frontend или backend;  
для каких задач удобна Java;  
какие программные продукты нам будут нужны на курсе.

---

## 2 Задачи и цели тестирования

### Цели занятия:

познакомиться с задачами по тестированию;  
понять зачем мы тестируем.

### Краткое содержание:

что такое тестирование и откуда оно появилось;  
кто такой тестировщик и что он делает;  
модели разработки ПО;  
жизненный цикл тестирования;  
чек листы и тест-кейсы;  
дефекты.

---

### 3 Методы тестирования

#### Цели занятия:

узнать различные методы тестирования;  
понимать в каких случаях их применять;  
научиться составлять тест кейсы;

#### Краткое содержание:

граничные значения;  
классы эквивалентности;  
PairWise;  
тест кейсы;

---

### 4 Виды тестирования

#### Цели занятия:

знать чем отличаются основные виды тестирования

#### Краткое содержание:

статическое тестирование;  
динамическое тестирование;  
модульное тестирование;  
интеграционные тестирование;  
системное тестирование;  
метод белого/черного/серого ящика;  
тестирование по природе приложения;  
тестирование по целям и задачам;  
тестирование производительности;

---

## Цели занятия:

научиться использовать актуальные системы ведения тест-кейсов и баг-трекер.

## Краткое содержание:

JIRA;  
заведение задач, дефектов;  
рабочие столы в JIRA;  
TestIT.

## Домашние задания

### 1 Использование TestIT и JIRA

Цель: Научится пользоваться TestIT и JIRA и применять знания полученные на прошлых уроках.

Перейти в TestIT(<http://185.177.92.66/>), создать секцию со своей фамилией и далее создавать все тест-кейсы и чек-листы в нем

- 1) создать общий шаг авторизации в отус и перехода в ЛК
- 2) Создать чек -лист с критическими проверками(где-то 10-15 проверок)
- 3) Создать ТК для тестирования раздела "О себе" в ЛК
- 4) Создать Тестовый план в который будут включены ТК из п3 и Чект Лист из п2

В JIRA:

Завести дефекты на условный интернет-магазин ver 1.0.0.1

- 1) Не работает добавления товара в корзину
- 2) Баннер на стартовой странице съехал на пол экрана
- 3) При добавлении 42 позиций товаров корзина обнуляется

Все дефекты назначить на преподавателя.

К заданию приложить ссылку на JIRA

---

## 6 Git и GitHub

### Цели занятия:

освоить базовые приемы работы в git.

### Краткое содержание:

Version Control и его история, зачем вести историю кода;

принципы работы Git, branch-и и commit-ы;

основные команд;

GitHub, создание аккаунта;

Pull Request.

---

## 7 Web технологии для тестировщика

### Цели занятия:

познакомимся с HTML, CSS, BackEnd, FrontEnd

### Краткое содержание:

HTML;

CSS;

BackEnd;

FrontEnd.

1 **Знакомство с IDE, пакеты, Hello World**

**Цели занятия:**

написать первую программу на Java.

**Краткое содержание:**

IDE и зачем она нам нужна;  
как запустить скомпилированный класс.

---

2 **Переменные, их типы и простые операции с ними**

**Цели занятия:**

научиться работать с массивами и основными типами данных

**Краткое содержание:**

тип данных;  
основные типы данных;  
операции над строками;  
операции с числами;  
константы;  
массивы в Java и зачем они нужны;  
функции, методы.

---

**3 Ветвления,  
циклы,  
функции**

**Цели занятия:**

научиться работать с циклами и условными операторами.

**Краткое содержание:**

условные операторы if, switch;  
короткая запись  $x=(1==1)?1:2$ ;  
циклы;  
for/foreach;  
While;  
do-while;  
функции;  
пара слов про рекурсию.

---

**4 Абстрактные  
классы,  
интерфейсы,  
примеры,  
реализация**

**Цели занятия:**

понимать наследование;  
работать с классами и интерфейсами;

**Краткое содержание:**

абстрактные классы;  
интерфейсы;  
примеры, реализация.

---

## 5 Основы ООП

### Цели занятия:

познакомиться с основами концепции ООП и реализацией ООП в Java.

### Краткое содержание:

полиморфизм и его виды;  
ООП как частный случай полиморфизма;  
почему ООП стало популярным;  
Java и ООП;  
классы и интерфейсы в Java;  
пару слов о функциональном программировании и отличие его от ООП.

---

## 6 Java Collections

### Цели занятия:

понять что такое Collections, какие виды и имплементации коллекций бывают, получить ответы на частые вопросы на собеседованиях по теме Java Collections

### Краткое содержание:

Java Collections;  
имплементации Collections;  
ответы на частые вопросы на собеседованиях.

---

## 7 Stream API

### Цели занятия:

научиться использовать stream API для обработки данных;  
работа с файлами

### Краткое содержание:

stream API;  
работа с файлами.

---

## 8 Практика. Подготовка к

### Цели занятия:

**Краткое содержание:**

практика по изученному материалу;  
подготовка к ДЗ.

**Домашние задания****1** Java

Цель: В результате выполнения ДЗ вы закрепите знания, полученные в блоке Java.

#1) Создайте родительский класс

Animal

##Свойства:

name, age, weight, color

##Методы:

Getter and Setter

Say(Вывод на экран: "Я говорю"),

Go(Вывод на экран: "Я иду"),

Drink(Вывод на экран: "Я пью"),

Eat(Вывод на экран: "Я ем")

##Переопределите метод toString (Возврат строки: "Привет! меня зовут name, мне age лет(год/года), я вешу - weight кг, мой цвет - color") лет или год, или года должно быть выбрано в зависимости от числа

Примеры:

Привет! меня зовут Бобик, мне 5 лет, я вешу - 15 кг,  
мой цвет - черный

Привет! меня зовут Бобик, мне 1 год, я вешу - 15 кг,  
мой цвет - черный

Привет! меня зовут Бобик, мне 2 года, я вешу - 15  
кг, мой цвет - черный

#2) Создайте интерфейс Flying

##Методы:

Fly

#3) Создайте класс Cat унаследуйте его от Animal

Переопределить метод Say(Вывод на экран: "Мяу")

#4) Создайте класс Dog унаследуйте его от Animal

Переопределить метод Say(Вывод на экран: "Гав")

#5) Создайте класс Duck унаследуйте его от

Animal, реализуйте интерфейс Flying

Переопределить метод Say(Вывод на экран: "Кря")

Реализовать метод Fly(Вывод на экран: "Я лечу")

#6) В основной программе:

Создайте ArrayList Animal

##Создайте в консоли меню, при входе в приложение на экран выводится запрос команды add/list/exit

Команды оформить в enum

При вводе команды она должна быть регистронезависимой и обрезать пробелы в начале и конце

##Если add

спросить какое животное(cat/dog/duck)

Спросить имя, возраст, вес, цвет

Инициализировать класс, добавить экземпляр в ArrayList и вызвать метод Say()

Вернуться к главному меню

##Если list

Вывести на экран метод toString() для всех элементов массива

##Если exit

выйти из программы

## 1 Введение в SQL

### Цели занятия:

познакомиться с БД;  
освоить простые запросы на языке SQL.

### Краткое содержание:

что такое БД;  
где применяются БД на проекте;  
Select;  
Insert;  
Update;  
Delete.

---

## 2 Более сложные запросы в SQL

### Цели занятия:

составлять более сложные запросы.

### Краткое содержание:

Where, Group by, Order By;  
выборка из нескольких таблиц;  
Join, Join Left, Join Right...

---

## 3 Java + SQL, Gradle vs Maven

### Цели занятия:

познакомиться с основными алгоритмами и структурами данных.

### Краткое содержание:

фреймворк JDBC.

---

#### 4 Практика. Подготовка к ДЗ

#### Цели занятия:

освоить материал модуля на практике.

#### Краткое содержание:

практика по изученному материалу;  
подготовка к ДЗ.

#### Домашние задания

##### 1 SQL

Цель: закрепить знания, полученные в блоке SQL.

Все должно быть сделано внутри программы на JAVA в качестве sql сервера использовать MySQL

- 1) Создать таблицу Student  
Колонки id, fio, sex, id\_group
- 2) Создать таблицу Group  
Колонки id, name, id\_curator
- 3) Создать таблицу Curator  
Колонки id, fio

- 4) Заполнить таблицы данными(15 студентов, 3 группы, 4 куратора)
- 5) Вывести на экран информацию о всех студентах включая название группы и имя куратора
- 6) Вывести на экран количество студентов
- 7) Вывести студенток
- 8) Обновить данные по группе сменив куратора
- 9) Вывести список групп с их кураторами
- 10) Используя вложенные запросы вывести на экран студентов из определенной группы(поиск по имени группы)

## 1 Настраиваем окружение

### Цели занятия:

выбрать тестовый фреймворк;  
создать проект.

### Краткое содержание:

создание gradle-проекта;  
подключение Selenium-зависимостей;  
сравнительный анализ junit и TestNg;  
добавление web driver manager;  
добавление и настройка логирования Log4j.

---

## 2 Локаторы

### Цели занятия:

научиться использовать локаторы для взаимодействия с UI веб сайта.

### Краткое содержание:

типы локаторов;  
CSS-локаторы;  
XPath-локаторы;  
сравнение CSS и XPath;  
валидация локаторов в браузере;  
подходы к хранению локаторов в проекте.

---

## 3 Команды чтения и управления

### Цели занятия:

взаимодействовать с элементами UI.

### Краткое содержание:

элементы UI.

---

**Цели занятия:**

производить стартовую настройку драйвера.

**Краткое содержание:**

конфигурирование драйвера при старте;  
практика.

**Домашние задания**

1 WebDriver

Цель: Практика в работе с WebDriver и Библиотеками Log2j2, web driver manager

1)

- Открыть Chrome в headless режиме
- Перейти на <https://duckduckgo.com/>
- В поисковую строку ввести ОТУС
- Проверить что в поисковой выдаче первый результат Онлайн-курсы для профессионалов, дистанционное обучение

2)

- Открыть Chrome в режиме киоска
- Перейти на [https://demo.w3layouts.com/demos\\_new/template\\_demo/03-10-2020/photoflash-liberty-demo\\_Free/685659620/web/index.html?\\_ga=2.181802926.889871791.1632394818-2083132868.1632394818](https://demo.w3layouts.com/demos_new/template_demo/03-10-2020/photoflash-liberty-demo_Free/685659620/web/index.html?_ga=2.181802926.889871791.1632394818-2083132868.1632394818)
- Нажать на любую картинку
- Проверить что картинка открылась в модальном окне

3)

- Открыть Chrome в режиме полного экрана
  - Перейти на <https://otus.ru>
  - Авторизоваться под каким-нибудь тестовым пользователем(можно создать нового)
  - Вывести в лог все cookie
-

## 5 Ожидания

### Цели занятия:

настраивать ожидания и работать с ними.

### Краткое содержание:

явные ожидания;  
зачем они нужны;  
неявные ожидания;  
зачем они нужны;  
практика.

---

## 6 Работа с нативными окнами

### Цели занятия:

работать с нативными окнами.

### Краткое содержание:

работа с alert, prompt, confirm, basic auth.

---

## 7 JavaScript

### Цели занятия:

познакомиться с JavaScript.

### Краткое содержание:

отличие JS от Java;  
взаимодействие с элементами.

---

## 8 Executing JavaScript

### Цели занятия:

реализовать исполнение JS кода в WebDriver.

### Краткое содержание:

зачем нам нужен Executing JavaScript;  
практика.

### Домашние задания

#### 1 Первый автотест

Цель: В результате выполнения дз вы реализуете автоматический тест, используя Java + Selenium.

Шаги теста:

- Открыть <https://otus.ru>
- Авторизоваться на сайте
- Войти в личный кабинет
- В разделе "О себе" заполнить все поля "Личные данные" и добавить не менее двух контактов
- Нажать сохранить
- Открыть <https://otus.ru> в "чистом браузере"
- Авторизоваться на сайте
- Войти в личный кабинет
- Проверить, что в разделе "О себе" отображаются указанные ранее данные

Домашнее задание принимается в виде ссылки на GitHub репозиторий

---

## 9 Тестирование API

### Цели занятия:

тестировать restAPI с помощью restAssured

### Краткое содержание:

API;  
restAssured;

## 1 Архитектура

### Цели занятия:

применять паттерны проектирования при разработке проекта.

### Краткое содержание:

основные паттерны проектирования:  
Chain of Command/Chain of Responsibility;  
Singleton;  
Factory;  
listener;  
прототип;  
адаптер;  
разделение тестов на различные слои.

---

## 2 Принцип разработки

### Цели занятия:

познакомиться с основными принципами написания кода

### Краткое содержание:

KISS;  
SOLID;  
DRY.

---

## 3 Page Object

### Цели занятия:

познакомиться с основным паттерном разработки авто-тестов.

### Краткое содержание:

проблематика;  
реализация;  
использование;  
Screen Object;  
Fluent change;

## Домашние задания

### 1 WebDriverFactory

Цель: В результате выполнения дз вы реализуете автоматический тест, используя Java + Selenium

Создайте класс WebDriverFactory со статическим методом create();

Метод create() принимает обязательный параметр webDriverName и необязательный параметр options, а возвращает соответствующий имени вебдрайвер с заданными (если были) options

Примеры использования

```
WebDriver wd =  
WebDriverFactory.createNewDriver("chrome");  
или  
FirefoxOptions options = new FirefoxOptions();  
WebDriver wd =  
WebDriverFactory.createNewDriver("firefox", options);
```

Шаги теста:

- Открыть <https://otus.ru>
- Авторизоваться на сайте
- Войти в личный кабинет
- В разделе "О себе" заполнить все поля "Личные данные" и добавить не менее двух контактов
- Нажать сохранить
- Открыть <https://otus.ru> в "чистом браузере"
- Авторизоваться на сайте
- Войти в личный кабинет
- Проверить, что в разделе "О себе" отображаются указанные ранее данные

## 1 Консультация по проекту и домашним заданиям

### Цели занятия:

получить ответы на вопросы по проекту, ДЗ и по курсу.

### Краткое содержание:

вопросы по улучшению и оптимизации работы над проектом;

затруднения при выполнении ДЗ;

### Домашние задания

#### 1 Проектная работа

Цель: Практика в работе с WebDriver и Библиотеками Log2j2, web driver manager.

1)

- Открыть Chrome в headless режиме
- Перейти на <https://duckduckgo.com/>
- В поисковую строку ввести ОТУС
- Проверить что в поисковой выдаче первый результат Онлайн-курсы для профессионалов, дистанционное обучение

2)

- Открыть Chrome в режиме киоска
- Перейти на [https://demo.w3layouts.com/demos\\_new/template\\_demo/03-10-2020/photoflash-liberty-demo\\_Free/685659620/web/index.html?\\_ga=2.181802926.889871791.1632394818-2083132868.1632394818](https://demo.w3layouts.com/demos_new/template_demo/03-10-2020/photoflash-liberty-demo_Free/685659620/web/index.html?_ga=2.181802926.889871791.1632394818-2083132868.1632394818)
- Нажать на любую картинку
- Проверить что картинка открылась в модальном окне

3)

- Открыть Chrome в режиме полного экрана
  - Перейти на <https://otus.ru>
  - Авторизоваться под каким-нибудь тестовым пользователем(можно создать нового)
  - Вывести в лог все cookie
-

**2 Защита  
проектных  
работ**

**Цели занятия:**

защитить проект и получить рекомендации экспертов.

**Краткое содержание:**

презентация проектов перед комиссией;  
вопросы и комментарии по проектам.

---

**3 Подведение  
итогов курса**

**Цели занятия:**

узнать, как получить сертификат об окончании курса,  
как взаимодействовать после окончания курса с OTUS и  
преподавателями, какие вакансии и позиции есть для  
выпускников и на какие компании стоит обратить  
внимание.

**Краткое содержание:**

организационные вопросы;  
рынок вакансий по направлению;  
статистика курса и вопросы по курсу.