

СТУДЕНТЫ

15-55 ЛЕТ

top

КОМПЬЮТЕРНАЯ  
АКАДЕМИЯ



учёба с нуля  
и бесплатно



диплом о проф.  
переподготовке



трудоустройство  
во время обучения



живая помощь  
от преподавателей



платформа  
с упражнениями



обновленная  
программа  
2025

## Тестирование ПО

(QA) на Python / Java\*

\* Существует 2 вариации курса Тестирование ПО, на Python или Java.  
Подробности уточняйте у сотрудников приёмной комиссии

736 академических  
часов

1 год длительность  
курса

368 часов аудиторных  
занятий

368 часов самостоятельной  
работы

Наши партнёры:

Yandex Cloud

hh карьерный  
маркетплейс

## Обзор IT бизнеса: технологии и термины

## Фазы процесса разработки программного обеспечения

## Методологии процесса разработки программного обеспечения

- Водопадная, спиральная и интерактивная (agile, scrum, xp) модели, RUP, MSF.

## Анализ существующих моделей и методов разработки

## Роль и место QA в процессе разработки программного обеспечения, тестирование и QA

## Введение в тестирование

## Цели и задачи тестировщика в команде

## Теория тестирования

- Подход.
- Технологии.
- Уровни.
- Процесс.
- Компоненты.

## Дефекты

- Жизненный цикл дефекта.
- Типы дефектов.
- Описание дефектов.

## Тесты

- Типы тестов.
- Организация тестов.
- Создание тестов.

## Тестирование сложных программных решений и комплексных систем

## Практическое занятие: написание тестов, тестирование, описание

## От тестирования к QA

- Цели и задачи QA.
- Стоимость тестирования.
- Стоимость дефекта.

## Requirements, введение в bug tracking systems

## Введение в планирование

- Задачи планирования.
- Тест-план.

## QA процесс

- Инициализация.
- Цели.
- Приоритеты.
- Сроки.
- Риски.

## QA процессы

- Подготовка.
- Тестирование.
- Рапортование.
- Финальная фаза.

## Практическое занятие по QA процессам

## Тестирование специфических областей

- Usablity.
- I18N/L10N.
- MLU.
- Мобильные устройства.
- Особенности тестирования веб-приложений.

## Программное обеспечение процесса тестирования

## Обзор средств сопровождения процесса разработки и тестирования

## Системы контроля версий

- Что такое контроль версий и для чего он необходим.

## Базовые принципы работы с GIT

## Особенности SCRUM-методологии

- Вопросы коммуникации: внутри команды, с руководителем, с подчиненными, с заказчиками.

## Сертификация ISTQB

- Обзор процесса сертификации.
- Обзор материалов и требований.

## Собеседование

- Примеры вопросов.
- Стратегия успешного прохождения собеседования.
- Анализ ситуаций, возникающих при прохождении собеседования.

## Что такое программирование, введение в язык программирования Python / Java

- Программирование, язык программирования и алгоритмы.
- Понятие программного проекта.
- Разделение языков программирования на низкий, средний, высокий уровень.
- Разделение языков программирования на интерпретаторы и компиляторы.
- Введение в Python / Java.

## Переменные и типы данных

- Типы данных.
- Переменные.
- Строки. Использование строк.
- Организация ввода данных с консоли.
- Константы и литералы.
- Операторы.
- Примеры построения программ с использованием блок-схем.

## Логические операторы и операторы ветвлений

- Преобразование типов данных.
- Логические операторы.
- Таблица приоритетов операторов.
- Конструкции логического выбора. Операторы ветвлений.

## Введение в теорию баз данных

- Введение в теорию баз данных.
- СУБД MySQL 8. Достоинства и недостатки. Обзор версий и редакций.
- Установка MySQL 8 Express.
- Утилиты MySQL 8.
- Работа с базой данных.

- Основы взаимодействия с СУБД MySQL 8.
- Запросы.

## Запросы SELECT, INSERT, UPDATE, DELETE

- Оператор SELECT.
- Ключевые слова IN, BETWEEN, LIKE.
- Оператор INSERT, UPDATE, DELETE.

## Многотабличные базы данных

- Аномалии взаимодействия с однотобличной базой данных.
- Принципы создания многотабличной базы данных.
- Многотабличные запросы.

## Функции агрегирования

- Функции агрегирования.
- Понятие группировки. Ключевое слово GROUP BY.
- Ключевое СЛОВО HAVING. Сравнительный анализ HAVING и WHERE.
- Подзапросы.

## Объединения

- Операторы для использования в подзапросах.
- Объединение результатов запроса.
- Объединения JOIN.

## ЭКЗАМЕН

## Windows Server 2019/2022

- Установка.
- Настройка сервисов: IIS, DHCP, DNS, File Server, MS SQL Server.

## Практическое занятие по Windows

## Linux

- Установка.
- Настройка демонов: apache, dhcp, DNS, ftp, MySQL.

## Введение в сетевые технологии

- Модель OSI.
- Уровни модели OSI.
- IP-адресация.
- Принципы маршрутизации.
- Протоколы IP, UDP, TCP, DHCP.
- Почтовые протоколы SMTP, IMAP, POP3.
- Файловые протоколы FTP, TFTP.

## Что такое автоматизированное тестирование

- Автоматизация тестирования.
- Важность автоматизации тестирования.
- Цели и задачи автоматизированного тестирования.
- Место автоматизированного тестирования в процессе разработки продукта.
- Этапы автоматизированного тестирования.
- Подходы к автоматизированному тестированию.
- Необходимость владения основами HTML/CSS для тестировщика.

## Введение в Web-технологии, структура HTML, форматирование текста с помощью HTML

- Введение в HTML.
- Введение в язык разметки. Язык разметки гипертекста HTML.
- Теги — основной элемент структуры HTML. Правила записи тегов и их атрибутов в стандарте XHTML на примере тегов <b>, <i>, <u>, <font>, <sup>, <sub>, <br />. Синтаксические отличия HTML4 от XHTML.
- Основные ошибки в записи тегов.

- Основные ошибки в записи тегов.
- Основная структура XHTML-документа. Элементы и их назначение.
- Кодировки страницы и теги <meta>.
- Классификация тегов: линейные и блочные.
- Модель форматирования текста: заголовки и абзацы. Элементы <p>, <h1>, <h2>. Выравнивание текста в блочных элементах — атрибут align.
- Классификация тегов: логическое и физическое форматирования.
- Работа с изображениями и ссылками.
- Практика: создание web-страницы.

## Форматирование с помощью CSS

- CSS — каскадные таблицы стилей.
- Теги без форматирования: <div> — блочный, <span> — линейный.
- Аналогия HTML и CSS на примере линейных и блочных тегов.
- Использование атрибутов class и id для задания стилей.
- Использование внешних CSS-файлов стилей.
- Практика: форматирование текста с помощью CSS.

## Таблицы

- Создание простейшей таблицы. Теги <table>, <tr> и <td>.
- Объединение ячеек: атрибуты colspan, rowspan.
- Теги логического структурирования таблиц: <thead>, <tbody>, <tfoot>. Теги логического группирования столбцов: <colgroup>, <col>.
- Управление рамками таблицы — атрибуты frame, rules.
- Практика: создание сложных таблиц.
- Основы табличной верстки. Пример табличной верстки: ее минусы.

## Формы, фреймы

- Введение в формы.
- Управляющие элементы форм.
- Создание форм с помощью HTML.
- Форматирование элементов форм с помощью CSS.
- Фреймы и их структура (теоретические сведения).

## Введение в Selenium

- Selenium.
- Необходимость использования Selenium.
- Архитектура Selenium.
- Что такое WebDriver?
- Установка Selenium.

## Использование тестов и команд

- Что такое тест?
- Тест в Selenium.
- Запись и запуск теста.
- Selenese.
- Понятие команды в Selenium.
- Типы команд.
- Примеры использования команд.

## Верификация в Selenium, WebDriver

- Верификация в Selenium.
- Assert и verify.
- Примеры использования.
- Понятие «test suite».
- Примеры использования.
- WebDriver.
- Понятие Waits.



## Введение в XML, XPath в WebDriver

- Что такое XML?
- История создания XML.
- Цели и задачи XML.
- XML-документ.
- Синтаксис и структура XML-документа.
- Описание структуры XML-документа с помощью DTD.
- Пространства имен XML.
- Понятие схемы, отличия схем от DTD.
- Парсеры XML.
- Примеры создания XML-документов.
- XML-документация.
- XPath.

## Selenium Server

- Selenium Server.
- Цели и задачи Selenium Server.
- Selenium Grid.
- Цели и задачи Selenium Grid.
- Настройка и запуск Selenium Server.
- Использование Selenium Server.
- Настройка и запуск Selenium Grid.
- Использование Selenium Grid.

## Создание framework для тестирования в Selenium

- Что такое framework?
- Необходимость создания framework.
- Принципы создания framework.
- Архитектура framework.
- Примеры использования framework.

## Анализ продуктов для автоматизации тестирования, автоматизированное тестирование веб-сервисов и мобильных приложений

- Обзор различных инструментов для тестирования.
- Автоматизированное тестирование веб-сервисов.
- Автоматизированное тестирование мобильных приложений.

## ФИНАЛЬНЫЙ ПРОЕКТ

## ПО ОКОНЧАНИИ КУРСА ВЫ НАУЧИТЕСЬ:

- Использовать инструменты тестирования ПО для мобильных и десктопных приложений, а также веб-проектов.
- Тестировать различные приложения.
- Использовать различные системы контроля версий.
- Применять основы веб-технологий, программирования, системного администрирования, а также поймете принципы работы с базами данных для их использования в автоматизированном тестировании.
- Создавать тест-план.
- Работать с баг-трекерами.
- Проводить автоматизированное тестирование с использованием различного программного обеспечения.
- Понимать архитектуру и принципы использования Selenium.
- Анализировать структуру веб-страниц и взаимодействовать с ними.
- Подбирать оптимальные инструменты для проведения автоматизированного тестирования.
- Создавать скрипты в Selenium для тестирования веб-страниц.
- Проводить автоматизированное тестирование с использованием различного программного обеспечения.
- Выбирать оптимальные методы тестирования.
- Использовать методы нагрузочного тестирования.
- Разбираться в языке структурированных запросов SQL.
- Уметь создавать многотабличные запросы.
- Понимать принципы работы подзапросов и функций агрегирования.
- Уметь производить нормализацию баз данных.
- Использовать хранимые процедуры, триггеры, виды, пользовательские функции.
- Подготовитесь к интервью на должность QA-инженера.
- Использовать библиотеку Hibernate.
- \* Существует 2 вариации курса Тестирование ПО, на Python или Java. Подробности уточняйте у сотрудников приёмной комиссии



17

лет на рынке



4,8

рейтинг  
на Я. Картах



380

филиалов в России



2 800

преподавателей



120 000+

студентов



24 000+

ОТЗЫВОВ



Также доступно  
онлайн-обучение

\*данные действительны  
на январь 2025 года