*АНО «Развитие человеческого капитала»*
*Обособленное подразделение «Техноград»*

УТВЕРЖДЕНО
Приказом Генерального директора
АНО «Развитие человеческого капитала»
№ ОД-329/25 от «1» октября 2025 г.

# ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

**Наименование программы**

**«Специалист по тестированию (QA). Ступень II»**

**Категория слушателей:** тестировщики безопасности, разработчики и аналитики в области программного обеспечения, техники, программисты, тест-дизайнеры, QA-инженеры, руководители отделов и служб тестирования, мануальные тестировщики, автоматизаторы, тест-менеджеры, руководители команд тестовых инженеров, архитекторы программных продуктов, тестировщики производительности, лица старше 18 лет, заинтересованные в получении новых компетенций, студенты образовательных учреждений СПО и вузов, обучающиеся по направлениям подготовки «Информатика и вычислительная техника», «Информационные системы и технологии», «Программная инженерия»

**Объем:** 32 часа.

**Форма обучения:** очная с применением дистанционных технологий.

**Форма документа, выдаваемого по результатам освоения программы:** сертификат о прохождении курса.

# УЧЕБНЫЙ ПЛАН

| Наименование компонентов программы | Обязательные аудиторные учебные занятия (ч.) | | Внеаудиторная (самостоятельная) учебная работа (ч.) | Всего учебной нагрузки (ч.) |
|---|---|---|---|---|
| | всего | в т.ч. практических и семинарских занятий | | |
| **Тема 1.** Клиент-серверная архитектура, база данных. | **4** | 3 | - | **4** |
| **Тема 2.** Способы фиксации информации в БД. | **4** | 3 | - | **4** |
| **Тема 3.** Программный интерфейс (Backend). | **4** | 3 | - | **4** |
| **Тема 4.** Технологии подготовки тестов для проверки API сайта. | **4** | 3 | - | **4** |
| **Тема 5.** Переменные и авторизация. | **4** | 3 | - | **4** |
| **Тема 6.** Скрипты и автотесты. | **4** | 3 | - | **4** |
| **Тема 7.** Итоговый разбор сервисов Postman. | **4** | 3 | - | **4** |
| **Тема 8.** Программирование кода автотеста на Selenium. | **4** | 3 | - | **4** |
| **ИТОГО:** | **32** | **24** | **0** | **32** |

## СОДЕРЖАНИЕ ПРОГРАММЫ, СТРУКТУРИРОВАННОЕ ПО ТЕМАМ, РАЗДЕЛАМ (МОДУЛЯМ)

| Наименование тем, разделов (разделов) | Содержание тем, разделов (модулей) |
|---|---|
| **Тема 1.** Клиент-серверная архитектура, база данных. | Модель организации ПО, пользовательский интерфейс клиента, логика обработки запросов клиента, получения данных из хранилища и обработки данных для предоставления клиенту, метод «серого ящика», язык запросов SQL, база данных (проверка целостности данных, анализ ошибок через логи БД, тестирование API и сложных запросов, взаимодействие с другими системами, нагрузочное тестирование, работа с миграциями и бэкапами).<br>Минимум требований к тестировщику, SQL-запросы, реляционные и нереляционные БД (ключ, колоночные, графовые, иерархические), набор взаимосвязанных таблиц (записи, поля), СУБД, основные типы данных, взаимосвязи в БД (первичный, внешний ключ), типы связей (один к одному, один ко многим, многие ко многим, многие к одному), примеры БД, установка Dbeaver для практических занятий с СУБД, SQL-скрипт, выборка (с условиями) из одной таблицы, операторы сравнения, логические операторы, комбинирование условий.<br>**Практическое задание № 1. Вывести из базы данных первые десять записей из таблицы track.** |
| **Тема 2.** Способы фиксации информации в БД. | Ключевые слова IN, BETWEEN, LIKE, маска подстроки, сортировка в SQL (по одному полю: по возрастанию, по убыванию; по нескольким полям), LIMIT (вывод определенного количества строк), вычисления в SQL, наименование результативных полей, AS, ORDER BY, запросы в БД с группировкой, агрегатные вычисления, GROUP BY, SUM, агрегатные функции.<br>**Практическое задание № 2. Выгрузить топ-10 самых длинных по названию групп в формате по убыванию.** |
| **Тема 3.** Программный интерфейс (Backend). | Тестирование backend части взаимодействия клиента с сервером, общение посредством HTTP протокола, прямые запросы, структура запросов и ответов (метод, URL, заголовки, тело, статус), защищенный протокол HTTPS, обзор сервисов инструмента Postman, разбор кодов ответов при запросе, обращение через API, виды API (rest, soap, graphQL, websocket, |

| | |
|---|---|
| | gRPC), формат Json, типы значений. Основные методы HTTP запросов, get, post. Запросы с разными типами Join (с условием, несколькими таблицами, full Join).<br>**Практическое задание № 3. Сделать по три запроса post и get с помощью API для обновления данных из разделов сайта.** |
| **Тема 4.** Технологии подготовки тестов для проверки API сайта. | Подготовка специальных папок в коллекции при выполнении HTTP запросов, способы (add a request, контекстное меню, new, напрямую через импорт curl), проверка достоверности информации с сервера, SQL-команда drop table (удаление базы данных), SQL-инъекция, дублирование методов, raw, formdate, добавление метаданных, query параметр, ключ status, доступные значения, токен авторизации.<br>**Практическое задание № 4. Прописать четыре основных HTTP-запроса к сущностям store и user.** |
| **Тема 5.** Переменные и авторизация. | Глобальные переменные для использования в запросах, environment в Postman, создание локальной переменной в рамках коллекции (вкладка variable, workspace), подключение 2-го проекта для проверки авторизации, сервис DummyAPI, получение через запрос списка пользователей, app-id, процедура получения токена, окружение переменной (интеграция в коллекцию).<br>**Практическое задание № 5. Создать коллекцию на тестовый API, добавить авторизацию (pid и токен).** |
| **Тема 6.** Скрипты и автотесты. | Разбор создания автотестов на запросы к API, механика подготовки автотеста на запрос post, написание скрипта в разделе post-response, обращение к объекту, получение ответа в виде статус-кода, test results, добавление сниппетов, pm.response, проверки статуса ответа и значения поля в теле ответа, response body, тесты на get, Json data, value check, проверка статуса кода, доступности и правильного id, ключи значения, возвращение массива объектов, тестирование offset, limit.<br>**Практическое задание № 6. Проработать минимум по два теста к каждому запросу из коллекции pet store, запустить тесты через коллекцию для проверки.** |
| **Тема 7.** Итоговый разбор сервисов Postman. | Проверка соответствия требованиям структуры ответа, наличие полей в ответах сервера, сниппет use tiny validator for Json data, переменная validSchema, сравнение фактической структуры ответа от API с ожидаемой схемой, рандомные данные в теле запроса, механизм |

| | |
|---|---|
| | шаблонов для генерации динамических тестовых данных, создание переменной окружения в автотесте, вкладка pre-request, set an environment variable.<br>**Практическое задание № 7. Добавить ко всем сущностям тесты на валидность схемы, применить шаблоны рандомов, использовать переменные окружения.** |
| **Тема 8.** Программирование кода автотеста на Selenium. | Работа с цифровыми инструментами в интерфейсе Visual Studio Code, написание скрипта кода для автотеста на Python, проведение парсинга с помощью библиотеки Selenium с целью управления браузером Chrome, типы Selenium (Grid, IDE, RC), кроссплатформенная и кроссбраузерная автоматизация, активация виртуального окружения, установка Selenium, импорт модуля драйвера для работы с проверками внешних ресурсов, WebElement, класс By, интеграция информации с сайта в консоль, локаторы, преимущества и недостатки использования Selenium.<br>**Практическое задание № 8. Составить скрипт кода с применением библиотеки** Selenium **для тестирования извлечения информации из отдельных разделов сайта.** |