

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

Разработка

программных проектов

с использованием JAVA

736 академических часов

1 год длительность курса

368 часов аудиторных занятий

368 часов самостоятельной работы

Наши партнёры:

Yandex Cloud

hh карьерный маркетплейс

## 1 модуль: Core & Backend

### Язык программирования Java:

#### Java SE (Core):

- Синтаксис.
- ООП.
- Коллекции.
- Исключения.
- Потоки ввода/вывода (I/O).
- Многопоточность (Concurrency).
- Лямбда-выражения.
- Stream API.
- Модули (JPMS).

#### Java Build Tools:

- Maven или Gradle (для управления зависимостями и сборки проекта).

#### JDBC (Java Database Connectivity):

- Основы взаимодействия с БД.

#### Чему научишься:

- Писать код на Java как профессионал: от простых программ до сложной логики.
- Понимать ООП (объекты, классы, наследование) — это «мозг» Java.
- Работать с данными: списки, коллекции, файлы (чтение/запись).
- Обработать ошибки так, чтобы программа не падала.
- Делать многозадачность: чтобы приложение делало несколько дел одновременно (например, загрузка + анимация).
- Собирать проект как конструктор (Maven/Gradle).

Проще: Ты освоишь основы Java — как «говорить» на этом языке, чтобы создавать умные программы.

### Теория баз данных:

#### SQL (Structured Query Language):

- DDL.
- DML.
- DQL.
- DCL.
- TCL.
- Запросы (SELECT, JOIN, агрегатные функции, подзапросы).
- Индексы.
- Транзакции (ACID).
- Нормализация.

#### PostgreSQL:

- Установка.
- Настройка.
- Администрирование (базовое).
- Типы данных.
- Расширения.

#### ORM (Object-Relational Mapping):

- Hibernate (JPA реализация) - аннотации.
- Сущности.
- Связи.
- HQL/JPQL.
- Кэширование.

#### Чему научишься:

- Работать с базами данных (БД): создавать таблицы, искать данные, связывать их между собой.
- Писать SQL-запросы: это как «вопросы» к базе («покажи всех пользователей из Москвы»).
- Настроить PostgreSQL — одну из самых популярных БД в мире.
- Использовать Hibernate: чтобы Java-код сам общался с базой без кучи SQL.

Проще: Ты научишься хранить и управлять данными (пользователи, товары, заказы) — как цифровой библиотекарь.

### Разработка серверных решений с использованием Java (Spring)

#### Spring Framework Core:

- IoC (Inversion of Control) / DI (Dependency Injection).
- Beans.
- Конфигурация (JavaConfig, Аннотации).

#### Spring Boot:

- Auto-Configuration.
- Starter Dependencies.
- Embedded Servers (Tomcat).
- Actuator.
- Properties/YAML конфигурация.

#### Spring MVC:

- Контроллеры.
- RESTful веб-сервисы (REST API).
- Обработка запросов/ответов (HTTP, JSON/XML).
- Валидация.
- Межсетевой экран (CORS).

#### Spring Data JPA:

- Упрощение работы с JPA (Hibernate).
- Репозитории.
- Query Methods.
- @Query.

#### Spring Security:

- Аутентификация (Authentication).
- Авторизация (Authorization).
- JWT (JSON Web Tokens).
- OAuth2 (основы).

#### Чему научишься:

- Создавать «движок» сайта/приложения на Spring — это фреймворк №1 для Java.
- Делать REST API — «мостики» для обмена данными (например, между приложением и сервером).
- Автоматизировать рутину: Spring Boot ускоряет разработку в разы.
- Защищать приложения: логины, пароли, шифрование (Spring Security).
- Работать с базой ещё проще (Spring Data).

Проще: Ты сможешь создавать мощные серверные системы — как «мотор» для сайтов и мобильных приложений.

## 2 модуль: Frontend & Клиентская часть

### Разработка веб-страниц (HTML5 & CSS3):

#### HTML5:

- Семантические теги (header, nav, section, article, footer).
- Формы и валидация.
- Мультимедиа (audio, video).
- Canvas (основы).
- Геолокация (основы).

#### CSS3:

- Селекторы.
- Блочная модель.
- Flexbox.
- Grid Layout.
- Анимации (Transitions, Animations).
- Медиа-запросы (Responsive Web Design).
- Переменные (Custom Properties).
- Препроцессоры (SASS/SCSS).

#### Чему научишься:

- Верстать страницы с нуля (кнопки, формы, меню).
- Делать сайты адаптивными — чтобы они красиво смотрелись на телефоне и компьютере.
- Анимировать элементы: плавные переходы, эффекты.
- Использовать Flexbox/Grid — это магия для идеальной верстки.
- Писать чистый CSS с помощью SASS/SCSS.

Проще: Ты научишься «рисовать» интерфейсы — как дизайнер, который знает код.

### Разработка клиентских сценариев (JavaScript):

#### Core JavaScript:

- Синтаксис (ES6+).
- Переменные (let, const).
- Типы данных.
- Функции (включая стрелочные).
- Объекты.
- Массивы.
- DOM Manipulation.
- События (Events).
- Асинхронность (Callbacks, Promises, async/await).
- Fetch API / AJAX (работа с сервером).

#### Modern JS Features:

- Modules (ES Modules).
- Classes, Destructuring.
- Spread/Rest операторы.
- Template Literals.

#### Браузерные API:

- Хранение данных (LocalStorage, sessionStorage).
- History API.
- Geolocation API.

### Чему научишься:

- «Оживлять» вёрстку: например, чтобы кнопки реагировали на клики.
- Работать с данными в браузере (корзина покупок, история просмотров).
- Общаться с сервером без перезагрузки страницы (отправка форм, загрузка контента).
- Использовать современный JS (ES6+): стрелочные функции, асинхронность.
- Работать с геолокацией, историей браузера и др.

Проще: Ты сделаешь сайты интерактивными — как фокусник, который заставляет страницы «дышать».

## 3 модуль: Инструменты разработки и инфраструктура:

### Системы контроля версий:

- Git.
- Платформы (GitHub, GitLab).

### Тестирование:

#### Backend:

- JUnit 5.
- Mockito.
- Spring Test.
- Testcontainers (для интеграционных тестов с Postgres).

#### Frontend:

- Jest.
- Testing Library.

### Контейнеризация и деплоймент

- Docker.
- Облачные платформы.

### Применение нейросетей в контексте курса:

#### API Нейросетевых сервисов

- Интеграция через REST API (OpenAI API, Anthropic Claude API, Google Gemini API, YandexGPT API, GigaChat и т.д.).

#### Использование в Java/Spring

- Генерация текстового контента (описания, комментарии, документация).
- Анализ настроений (Sentiment Analysis) отзывов или комментариев.

### Чему научишься:

- Интегрировать нейросети (ChatGPT, GigaChat и др.) в Java-приложения через API.
- Генерировать контент: описания товаров, комментарии, документацию.
- Анализировать отзывы: например, определять позитивные/негативные мнения.

Проще: Ты добавишь «искусственный интеллект» в свои проекты — например, чтобы сайт сам писал тексты или анализировал эмоции пользователей.

### Кем может работать выпускник данного курса:

#### Java Backend Developer (Junior/Middle):

##### Почему:

Курс дает глубокую базу по Java, Spring (Boot, MVC, Data JPA, Security) и PostgreSQL (SQL, Hibernate). Это основной стек для огромного количества вакансий в корпоративном секторе, финтехе, enterprise. Выпускник сможет разрабатывать и поддерживать серверную логику, API, работать с базами данных.

#### FullStack Developer (Junior/Middle) с упором на Java:

##### Почему:

Курс покрывает весь стек: от верстки (HTML5/CSS3) и клиентской логики (JavaScript) до сервера (Java/Spring) и БД (PostgreSQL). Выпускник сможет создавать законченные веб-приложения «от и до», что очень ценится в стартапах и небольших/средних компаниях.

#### Web Developer (с сильным бэкендом):

##### Почему:

Знание всего цикла разработки веб-приложения (фронтенд + бэкенд + БД) делает выпускника универсальным специалистом, способным решать комплексные задачи веб-разработки, с глубоким пониманием серверной части.

#### Разработчик, интегрирующий решения на базе ИИ/Нейросетей (Junior):

##### Почему:

Это ключевое конкурентное преимущество данного курса. Выпускник не просто разработчик, а разработчик, понимающий, как практически применить нейросетевые API в реальных проектах.

##### Он сможет:

- Добавлять чат-ботов или AI-ассистентов в приложения.
- Интегрировать генерацию контента (текст, код, изображения).
- Реализовывать базовый анализ данных/текста (например, тональность отзывов).
- Автоматизировать рутинные задачи с помощью AI.

Это направление стремительно набирает обороты, и специалистов с практическими навыками интеграции AI еще относительно мало.

### Почему курс актуален:

#### Незыблемость Java и Spring в Enterprise:

Java остается одним из самых востребованных языков в мире, особенно для крупных, надежных, высоконагруженных систем (банки, телеком, госсектор, крупный e-commerce). Spring (и Spring Boot) – де-факто стандарт для Java backend. Спрос на этих специалистов стабильно высокий.

#### FullStack Универсальность:

Рынок ценит разработчиков, понимающих весь процесс создания приложения. Знание фронтенда (HTML/CSS/JS) в дополнение к сильному Java-бэкенду делает выпускника более гибким и востребованным в проектах разного масштаба.

#### PostgreSQL – Лидер OpenSource БД:

PostgreSQL – мощная, надежная и популярная реляционная БД. Умение проектировать БД, писать эффективные SQL-запросы и работать с ORM (Hibernate) – обязательный навык для backend/fullstack разработчика.

#### Фундамент для роста:

Полученные знания (Java, ООП, SQL, веб-принципы, работа с API) создают прочный фундамент для изучения более сложных тем в будущем (продвинутой AI/ML, облака, другие языки).

#### Взрывной спрос на Интеграцию Нейросетей (AI):

Это главный тренд и точка актуальности курса:

- **Необходимость:** Бизнес требует внедрения AI для автоматизации, улучшения пользовательского опыта, генерации контента, анализа данных. Разработчики, умеющие это делать, нужны прямо сейчас.
- **Практический подход:** Курс учит не теории нейросетей, а практической интеграции через API – именно то, что нужно большинству компаний на текущем этапе. Это навык, который можно применить немедленно.
- **Конкурентное преимущество:** Выпускник будет обладать уникальным сочетанием классических, востребованных навыков FullStack Java разработчика и актуальнейших навыков работы с AI API. Это резко повышает его ценность на рынке труда.

## ПО ОКОНЧАНИИ КУРСА СЛУШАТЕЛЬ БУДЕТ:

- Понимать и уметь использовать фундаментальные принципы создания программ с использованием Java.
- Уметь создавать, компилировать, и производить отладку приложений в IDE Eclipse.
- Уметь проектировать и реализовывать различные алгоритмы.
- Разбираться и использовать в проектах принципы ООП.
- Уметь проектировать классы различной степени сложности.
- Использовать механизмы generics для построения шаблонных классов.
- Понимать механизмы многопоточности Java.
- Уметь пользоваться системой контроля версий.
- Использовать принципы командного взаимодействия.
- Применять паттерны проектирования.
- Использовать юнит-тестирование.
- Разбираться в языке структурированных запросов SQL.
- Уметь создавать многотабличные запросы.
- Понимать принципы работы подзапросов и функций агрегирования.
- Разбираться в тонкостях реализации клиентских сценариев под разные браузеры.
- Понимать особенности применения HTML5 по отношению к JavaScript.
- Уметь сериализовать и парсить данные, используя JSON.
- Владеть принципами создания асинхронных запросов с помощью AJAX.
- Понимать фундаментальные принципы создания серверных решений с использованием Java.
- Уметь создавать, компилировать, и отлаживать веб-приложения.
- Уметь взаимодействовать с источниками данных.
- Использовать сетевые механизмы.
- Понимать и применять паттерн MVC.
- Уметь создавать JSP-решения.
- Использовать фреймворк Spring.
- Использовать библиотеку Hibernate.



17

лет на рынке



4,8

рейтинг  
на Я. Картах



500+

филиалов в России



2 800

преподавателей



150 000

студентов



24 000+

ОТЗЫВОВ



Также доступно  
онлайн-обучение

\*данные действительны  
на июль 2025 года