



Общество с ограниченной ответственностью «Ребреин»

ИНН 7727409582, ОГРН 1197746106161

Адрес: 123056, город Москва, Большая Грузинская ул, д. 36а стр. 5а, офис 13

Утверждено

Приказом № ПР-1 от 17.06.2025 г.

Генеральный директор

 Фролкина Е.А.  
«17» июня 2025 г.

ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА  
– ПРОГРАММА ПОВЫШЕНИЯ КВАЛИФИКАЦИИ  
«СИСТЕМЫ ПОТОКОВОЙ ОБРАБОТКИ ДАННЫХ НА БАЗЕ АРАСНЕ KAFKA»

**Срок реализации:** 1,5 месяца

**Количество часов:** 80 акад. ч.

**Форма обучения:** заочная форма

**Формат обучения:** с применением  
исключительно дистанционных технологий

**Возраст обучающихся:** для лиц старше 17  
лет, имеющих или получающих среднее  
профессиональное и (или) высшее  
образование

Москва, 2025 г.

## 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Настоящая дополнительная профессиональная программа – программа повышения квалификации «Системы потоковой обработки данных на базе Apache Kafka» (далее – Программа) разработана в соответствии с:

- Федеральным законом от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации»;
- Приказом Министерства образования и науки РФ от 1 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам»;
- Постановлением Правительства РФ от 11.10.2023 № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- Профессиональным стандартом 06.026 «Системный администратор информационно-коммуникационных систем», утвержден приказом Министерства труда и социальной защиты Российской Федерации от 26.10.2020 года № 60580;
- ФГОС высшего образования – бакалавриат по направлению 09.03.02 Информационные системы и технологии, утв. приказом Минобрнауки России от 19.09.2017 №926;
- Локальными нормативными актами ООО «Ребреин».

В данной программе учтены основные идеи формирования универсальных учебных действий учащихся и соблюдена преемственность с программами высшего и/или среднего профессионального образования.

**Направленность программы:** Программа имеет техническую направленность.

### Адресат:

- Data-инженеры (Middle/Senior). Участники научатся работать с большими потоками данных, используя Kafka как надёжный инструмент для их обработки, маршрутизации и передачи.
- Backend-разработчики. Разработчики освоят интеграцию Kafka в распределённые и микросервисные архитектуры, где важна асинхронная обработка сообщений, взаимодействие сервисов и обеспечение отказоустойчивости.
- DevOps-инженеры (Middle). Инженеры смогут разворачивать и поддерживать высоконагруженные кластеры Kafka, обеспечивать отказоустойчивость, мониторинг и безопасность брокеров сообщений.

### Требования к входным знаниям обучающегося:

- Опыт работы с Linux – уверенное владение командной строкой, базовые навыки администрирования серверов.
- Базовые знания сетевых технологий – понимание TCP/IP, портов, маршрутизации и протоколов передачи данных.
- Опыт работы с базами данных – понимание реляционных и/или NoSQL баз данных, знание SQL будет преимуществом.
- Язык программирования – опыт работы с Java, Python или другим языком для взаимодействия с API Kafka.
- Понимание архитектуры микросервисов и асинхронных систем – знания о очередях сообщений, продюсерах и консьюмерах.
- Опыт работы с контейнерами и оркестраторами (Docker, Kubernetes) – желателен, но не обязательен.

### **Актуальность реализации:**

Современные компании всё чаще работают с потоковыми данными: события от пользователей, телеметрия IoT, логирование микросервисов, финансовые транзакции и аналитика в реальном времени. Для надёжной обработки таких потоков необходимы распределённые, масштабируемые и отказоустойчивые системы. Apache Kafka — де-факто стандарт для решения этих задач в индустрии, используемый в банковской сфере, e-commerce, телекоммуникациях и крупных интернет-компаниях. Владение Kafka позволяет инженерам эффективно строить архитектуры потоковой обработки, обеспечивая высокую производительность и надёжность систем.

### **Отличительные особенности программы:**

- Практическая направленность — обучение проходит на реальных сценариях работы с потоками данных, от настройки брокеров до интеграции с микросервисами.
- Фокус на отказоустойчивость и масштабируемость — изучение multi-broker кластеров, репликации и партиционирования.
- Глубокое погружение в API Kafka — работа с Producer API, Consumer API, Kafka Streams и Kafka Connect.
- Безопасность и мониторинг — настройка аутентификации, шифрования и инструментов мониторинга для реальных нагрузок.
- Подходит для DevOps и Backend инженеров — сочетание администрирования кластера и разработки потоковых приложений.

**Объем и срок освоения программы:** 80 академ. ч. в течение 1,5 мес. (6 недель).

Доступ к материалам Программы у обучающихся остаётся и после окончания периода обучения. Это позволяет повторять изученный материал в удобное время, восполнять пробелы в знаниях, а также возвращаться к практическим заданиям при решении рабочих задач. Такой формат способствует более глубокому закреплению навыков и поддерживает профессиональное развитие выпускников даже после завершения обучения.

**Выдаваемый документ о квалификации:** удостоверение о повышении квалификации и/или сертификат об успешном освоении программы.

### **Цели и задачи программы:**

Освоение навыков проектирования, развертывания и эксплуатации систем потоковой обработки данных на базе Apache Kafka с обеспечением масштабируемости, отказоустойчивости и безопасности.

### **Программа направлена на решение следующих основных задач:**

Обучающие:

- Освоить архитектуру Apache Kafka, работу брокеров, топиков и партиций.
- Научиться создавать и настраивать продюсеров и консументов для обработки потоков данных.
- Разобраться с инструментами Kafka Connect, Kafka Streams и Schema Registry.
- Изучить методы мониторинга, логирования и обеспечения отказоустойчивости кластера.

Развивающие:

- Развивать навыки проектирования распределённых систем обработки данных.
- Научиться оптимизировать производительность потоковой обработки.
- Формировать умение выявлять и устранять проблемы в работе Kafka-кластера.

#### Воспитательные:

- Формировать ответственность за качество и безопасность потоковых данных.
- Развивать культуру планирования отказоустойчивых архитектур и соблюдения лучших практик.

#### Планируемые результаты:

##### Знания:

- Архитектура Apache Kafka и принципы работы распределённых брокеров.
- Типы данных и способы сериализации (Avro, JSON, Protobuf).
- Методы обеспечения безопасности (аутентификация, шифрование, ACL).
- Инструменты мониторинга и логирования Kafka.

##### Умения:

- Развёртывать multi-broker кластеры и настраивать топики, партиции и репликацию.
- Настраивать продюсеров и консументов для обработки потоков данных.
- Интегрировать Kafka с микросервисами и сторонними системами.
- Применять Kafka Streams и Kafka Connect для потоковой обработки.

##### Навыки:

- Диагностировать и устранять проблемы в работе кластера.
- Обеспечивать отказоустойчивость и масштабируемость систем потоковой обработки.
- Эффективно использовать CI/CD-процессы для деплоя и поддержки Kafka-инфраструктуры.

#### Перечень профессиональных компетенций, на получение которых направлено обучение:

На основе профстандарта 06.026 «Системный администратор информационно-коммуникационных систем»:

- В/02.5 Обеспечение работы технических и программных средств информационно-коммуникационных систем;
- С/05.6 Выполнение обновления программного обеспечения сетевых устройств информационно-коммуникационных систем;
- С/08.6 Планирование и проведение работ по распределению нагрузки между имеющимися ресурсами, снятию нагрузки на сетевые устройства информационно-коммуникационных систем перед проведением регламентных работ, восстановлению штатной схемы работы в случае сбоев.

Таким образом, в результате освоения программы у обучающихся формируются следующие профессиональные компетенции:

- ОПК-5. Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем;
- ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий;
- ОПК-7. Способен осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем.

## **Организационно-педагогические условия реализации программы дополнительного профессионального образования**

**Язык реализации образовательной программы:** обучение проводится на русском языке.

**Форма обучения:** заочная форма.

**Особенности реализации программы:** программа реализуется с использованием электронного обучения и исключительно дистанционных образовательных технологий.

**Условия набора:** на обучение принимаются все желающие лица, оплатившие обучение и заключившие договор об образовании. Обучение проходит в индивидуальном формате без формирования учебных групп. Обучающийся самостоятельно определяет время освоения Программы.

**Формы проведения занятий:**

- занятия в текстовом формате;
- практическая работа;
- самостоятельная работа с литературой;
- индивидуальные вопросы.

### **Материально-техническое оснащение**

#### **Материальное обеспечение программы**

Занятия проводятся в системе дистанционного обучения «Rebrain». Каждый обучающийся и педагог оснащены доступом к системе дистанционного обучения: <https://rebrainme.com/>. У педагога дополнительного профессионального образования имеется необходимое оборудование средства для реализации программы: ноутбук с подключением к интернету, программное обеспечение.

#### **Методическое обеспечение программы**

Программа обеспечена:

- учебно-методическими материалами (текстовые занятия, полезными материалами);
- практическими заданиями.

#### **Кадровое обеспечение:**

К реализации программы в качестве педагогов дополнительного образования допускаются лица:

1) отвечающее одному из требований:

а) имеющее высшее образование или среднее профессиональное образование в рамках укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования «Образование и педагогические науки»;

б) имеющее высшее образование либо среднее профессиональное образование в рамках иных укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования при условии его соответствия дополнительной общеобразовательной общеобразовательной программе, реализуемой ООО «Ребреин», и получение при необходимости дополнительного профессионального образования педагогической направленности;

в) успешно прошедшее промежуточной аттестации не менее чем за два года обучения по образовательным программам высшего образования по специальностям и направлениям

подготовки, соответствующей направленности дополнительной общеобразовательной общеразвивающей программе;

2) не имеющее ограничений на занятие педагогической деятельностью, установленных законодательством Российской Федерации;

3) прошедшее обязательный предварительный (при поступлении на работу) и периодические медицинские осмотры (обследования), а также внеочередные медицинские осмотры (обследования) в порядке, установленном законодательством Российской Федерации.

Реализация Программы также возможна лицами, привлекаемыми на условиях гражданско-правового договора в соответствии с действующим законодательством РФ.

## 2. УЧЕБНЫЙ ПЛАН

№ п/ п	Наименование модуля	Количество часов			Формы контроля / аттестация
		Всего	Теория	Практика	
1	Модуль 1 “Онбординг”	2	1	1	Входное тестирование
2	Модуль 2 “Инфраструктура”	28	8	20	Практическое задание
3	Модуль 3 “Разработка”	40	10	30	Практическое задание
4	Итоговая аттестация	10		10	Итоговое практическое задание

## 3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

№ п/ п	Наименование модуля	1 неделя	2 неделя	3 неделя	4 неделя	5 неделя	6 неделя
1	Модуль 1 “Онбординг”	2					
2	Модуль 2 “Инфраструктура”	12	13	3			
3	Модуль 3 “Разработка”			10	13	13	4
4	Итоговая аттестация						10   А

## 4. РАБОЧАЯ ПРОГРАММА

### Модуль 1. Онбординг

Теория 1 академ. ч. Практика 1 академ. ч.

Модуль состоит из следующих тем:

Тема 1: Онбординг

В модуле обучающемуся предоставляется вводный конспект, содержащий общую информацию о программе, структуре курса, форматах взаимодействия с материалами и ожидаемых результатах обучения.

Предусмотрено прохождение входного тестирования, включающего 7 вопросов, направленных на закрепление информации из онбординга. В рамках темы обучающийся выполняет задание по целеполаганию: формулирует свою цель прохождения программы, указывает желаемые навыки по окончании обучения, а также оценивает текущий уровень своих знаний, выбрав один из предложенных вариантов.

## **Модуль 2. Инфраструктура**

Теория 8 академ. ч. Практика 20 академ. ч.

Модуль состоит из следующих тем:

Тема 1: Знакомство с Kafka

Содержание: Шины данных. Об Apache Kafka. Возможности Kafka. Основные понятия. Архитектура Kafka. Кластер Kafka и его режим работы. Установка и базовая настройка Kafka. Практическое задание.

Тема 2: WebUI. Настройки Kafka

Содержание: Обзор WebUI. Архитектура с ZooKeeper. Архитектура с KRaft. Практическое задание.

Тема 3: Безопасность в Kafka: возможности и best practices

Содержание: Безопасность Kafka. Типы безопасности Kafka. Kerberos. Архитектура Kerberos + GSSAPI + Kafka. Реализация. Best practices. Практическое задание.

Тема 4: Мониторинг Kafka

Содержание: Мониторинг Kafka: архитектура. Настройка мониторинга Kafka на базе Prometheus & JMX exporter. Основные метрики Kafka. Практическое задание.

Тема 5: Отказоустойчивость Kafka. Multi-broker cluster

Содержание: Архитектура. Отказоустойчивость Kafka. Защищённый кластер. Hardware Administration. Практическое задание.

Модуль направлен на освоение архитектуры Apache Kafka и принципов её эксплуатации. Участники разберутся с установкой и базовой настройкой кластера, изучат возможности WebUI, познакомятся с архитектурами на ZooKeeper и KRaft. Особое внимание уделяется вопросам безопасности, мониторинга и организации отказоустойчивого multi-broker кластера. Практика позволяет закрепить полученные знания на примере настройки защищённых и масштабируемых решений.

## **Модуль 3. Разработка**

Теория 10 академ. ч. Практика 30 академ. ч.

Модуль состоит из следующих тем:

Тема 1: Producer API

Содержание: Какие клиенты для Kafka существуют. Конфигурация Producer. Транзакции в Producer. Serializers (сериализаторы). Практика. Практическое задание.

Тема 2: Consumer API

Содержание: Consumers и Consumer Groups. Консьюмер-группы и ребаланс партиций. Конфигурация Consumer. Коммиты и оффсеты. API. Poll loop. Немного о клиентах. Немного о транзакциях. Десериализация. Практическое задание.

### Тема 3: Schema Registry

Содержание: Как создавать сериализаторы и десериализаторы, какими возможностями они обладают. Зачем нужна структура сообщений она и как её отслеживать. Avro и Schema Registry. Event-driven архитектура. Практическое задание.

### Тема 4: Kafka connect. Connect API. Kafka Streams

Содержание: Kafka Connect. Kafka Streams. KSQL. Архитектура. Практическое задание.

### Тема 5: Kafka как распределённая система. Гарантии доставки

Содержание: Репликация. Гарантии доставки. Retention.

Модуль фокусируется на работе с API и инструментах для обработки потоковых данных. Участники научатся конфигурировать продюсеров и консьюмеров, работать с сериализаторами и десериализаторами, управлять оффсетами и транзакциями. Рассматриваются возможности Schema Registry и Avro для организации согласованных структур сообщений, а также Kafka Connect, Kafka Streams и KSQL для построения сложных интеграций и потоковой аналитики. Отдельное внимание уделяется гарантиям доставки сообщений и принципам работы Kafka как распределённой системы.

Каждая тема модуля включает текстовое занятие с теоретическим материалом и пошаговыми инструкциями, после изучения которого предлагается практическое задание. Практические задания рассчитаны на 1-2 академических часа. Выполнение заданий предполагает отправку решения на проверку через личный кабинет обучающегося. Критерии оценки прописаны в описании к каждому заданию. В случае корректного выполнения выставляется зачёт. Если работа содержит ошибки, задание возвращается на доработку. При повторной неудачной попытке (после двух доработок) обучающийся получает «незачёт».

### Итоговая аттестация.

Блок посвящён выполнению финального практического задания без предварительного теоретического блока.

## 5. МЕТОДИЧЕСКИЕ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Программа обеспечена системой дистанционного обучения <https://rebrainme.com/>.

Педагогические технологии:

- технология дифференцированного обучения;
- технология разноуровневого обучения;
- технология развивающего обучения;
- технология проблемного обучения;
- технология дистанционного обучения.

Методы обучения:

- словесный, наглядный практический;
- объяснительно – иллюстративный;
- частично-поисковый, исследовательский проблемный;
- игровой, дискуссионный.

**Электронно-библиотечные ресурсы и системы, информационно-справочные системы:**

1. Научная электронная библиотека eLIBRARY.RU.
2. Собственные учебные материалы: <https://rebrainme.com/courses/kafka>
3. Официальная документация Kafka [Электронный ресурс]: <https://kafka.apache.org/quickstart>

## **6. ОЦЕНКА КАЧЕСТВА ОСВОЕНИЯ ПРОГРАММЫ**

**Оценочные материалы:**

Для отслеживания результатов освоения программы среди слушателей проводится текущий контроль, промежуточный контроль и итоговое оценивание.

**Текущий контроль**

Осуществление текущего контроля проводится после занятий в виде написания практических заданий или тестирований. Тематика и условия выполнения практических заданий расписаны в личном кабинете обучающегося в СДО. Педагог проверяет решение и принимает решение о принятии решения (зачет), о необходимости доработать решение или о незачете. Если промежуточный контроль представлен в виде тестирования, подсчет верных ответов и выставление оценки «зачёт» и «незачёт» происходят в автоматическим решиме в СДО.

**Модуль 2 “Инфраструктура”**

**Тема 2: WebUI. Настройки Kafka**

**Задание**

В рамках практикума вам будет предоставляться виртуальное окружение, где необходимо выполнять задания. При нажатии кнопки «Начать задание» начнётся проект подготовки вашего персонального виртуального окружения. Вы получите к нему удалённый доступ по SSH.

Дальнейшие настройки Kafka вам необходимо выполнять в выданном окружении. После выполнения задания нажмите «Проверить задание», после чего задание будет проверено в автоматическом режиме.

В данном задании надо запустить Kafka без ZooKeper в режиме Kraft, после чего создать топик и записать сообщение в него. Также необходимо запустить UI для kafka.

1. Подключитесь к выделенной при старте задания машине по SSH. Архив с Kafka уже развернут в папке /opt/kafka.
2. Запустите Kafka без помощи Zookeeper на порту 9092. Так же как и в прошлом уроке используйте встроенные скрипты из папки /opt/kafka/bin. и конфиг /opt/kafka/config/kraft/server.properties (конфиг передается через аргумент) Помните что перед запуском Kafka необходимо создать id кластера с помощью скрипта /opt/kafka/bin/kafka-storage.sh и с его же помощью прописать данный id в конфиг /opt/kafka/config/kraft/server.properties
3. Запустите Kafka UI на порту 8082 через docker compose из директории /opt/docker/. Для корректной работы UI поправьте IP в файле /opt/docker/docker-compose.yaml. на IP своей машины, а в файле /opt/kafka/config/kraft/server.properties. пропишите IP своей машины в строку advertised.listeners=PLAINTEXT://{YOUR IP}:9092.

4. Создайте журнал (Topic) с названием test-events. с помощью встроенного скрипта /opt/kafka/bin/[kafka-topics.sh](#)
5. Подключите к test-events консольные клиенты Producer через встроенные скрипты Kafka в директории /opt/kafka/bin
6. Выполните отправку сообщения HelloWorld через консольный Producer.
7. Подключитесь к Kafka UI, через браузер по IP-адресу машины и 8082 порту, убедитесь в UI, что сообщения созданы.

## Модуль 3 “Разработка”

### Тема 3: Schema Registry

#### Задание

В данном задании мы познакомимся с сериализатором argo на python и попробуем провести данные через брокер используя schema registry.

1.Запустите Kafka с и убедитесь что работает корректно. В этом уроке воспользуемся эмуляцией кластера и запустим kafka в контейнере /opt/docker/kafka/ kafka в докере стартует с компонентами confluent в триальном режиме. по адресу <http://YourIP:9021> доступен контрол-центр.

2.После подключения добавьте следующие записи в файл /etc/hosts. Вместо 10.129.0.13 укажите ваш адрес.

```
10.129.0.32 kafka-centos-8
10.129.0.32 kafka1
10.129.0.32 kafka
10.129.0.32 schema-registry
10.129.0.32 control-center
10.129.0.32 mysql
10.129.0.32 kafka-connect
```

[...]

#### Итоговое оценивание

В конце программы обучающиеся сдают итоговую аттестацию.

#### Итоговое задание

1. Подключитесь к выделенным при старте задания машинам по SSH. Архив с Kafka уже развернут в папке /opt/kafka. После подключения добавьте следующие записи в файл /etc/hosts. На каждой машины добавьте в файл /etc/hosts следующую запись

```
<machine_1_ip> final1
<machine_2_ip> final2
<machine_3_ip> final3
```

2. Сгенерируйте uuid для кластера на одной из машин и распространите его среди остальных. Убедитесь, что на каждой машине создался файл /tmp/kraft-combined-logs/meta.properties и в нем содержится верная информация.
3. На каждой машине запустите kafka в kraft режиме для дальнейшего объединения их в кластер. Для этого в файле /opt/kafka/config/server.properties измените следующие параметры:

node.id=1 (в зависимости от номера узла 1,2 или 3)

listeners=PLAINTEXT://{{MACHINE IP}}:9092,CONTROLLER://{{MACHINE IP}}:9093

advertised.listeners=PLAINTEXT://{{MACHINE IP}}:9092

Убедитесь, что на каждой машине создался файл /tmp/kraft-combined-logs/meta.properties, и в нем содержится верная информация.

Если это не так, поправьте node.id на указанный в файле настроек.

4. Создайте топик kraft-test на 3 патриции с фактором репликации 3.
5. На второй машине запустите Kafka UI в docker контейнера с помощью compose файла по пути /opt/kafka/ui. Предварительно пропишите в него IP всех машин кластера.
6. На 3м узле в директории /opt/python дополните скрипт consumer.py так, чтобы он считывал сообщения из всех трех инстансов kafka из топика kraft-test.
7. Убедитесь, что при пуше сообщений в топик kraft-test в любой из инстансов, вы сможете читать его с помощью consumer.py.
8. Перед отправкой на проверку, убедитесь, что очередь сообщений пуста.

Результаты текущего контроля и итогового оценивания отображаются в личном кабинете слушателя в системе дистанционного обучения <https://rebrainme.com/>.

По результатам сдачи текущего контроля, промежуточного контроля и итогового оценивания педагог даёт обратную связь слушателям, отмечает их сильные стороны и обращает внимание на зоны для развития. При необходимости педагог может повторить пройденные темы со слушателями, если установлен факт плохого закрепления и усвоения темы у слушателей.