

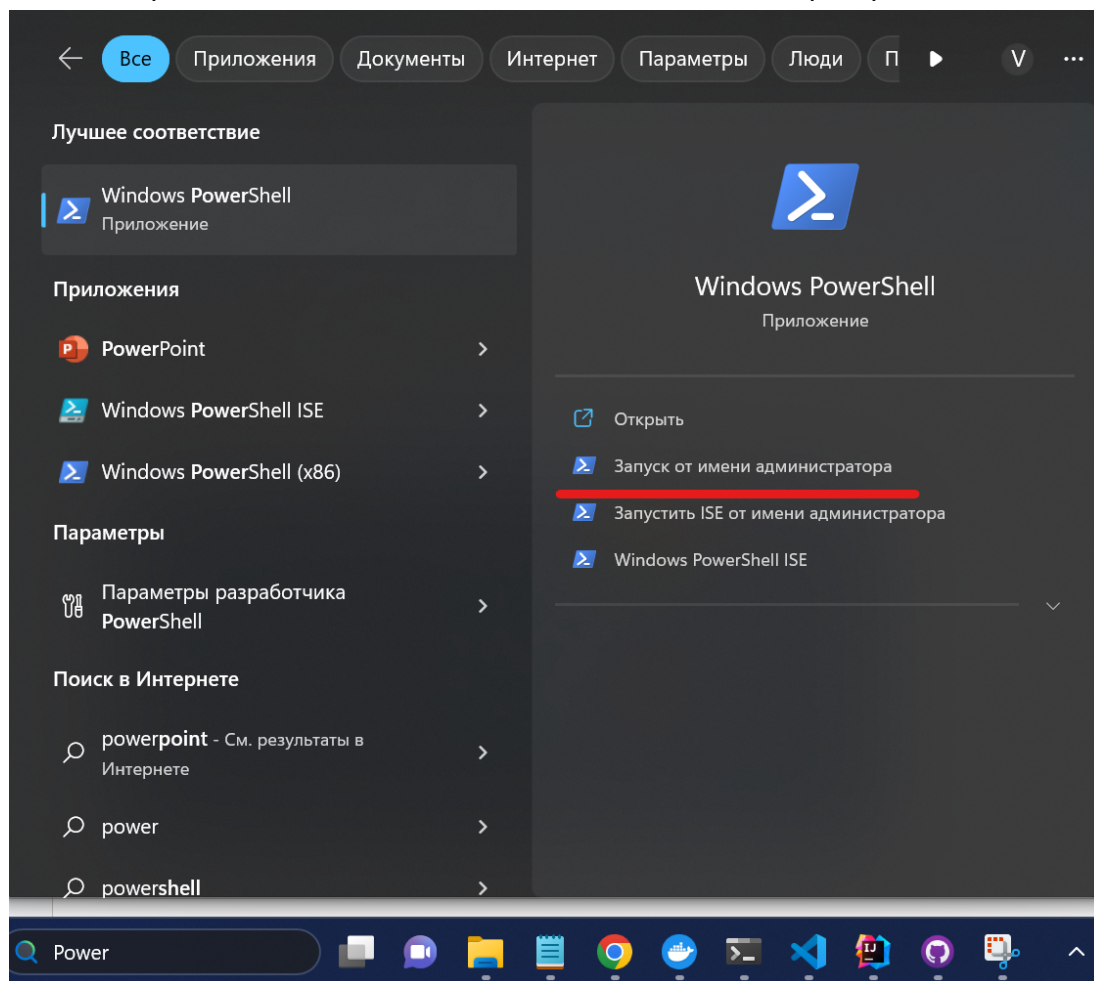
Оглавление

Оглавление.....	1
Полная инструкция по запуску сервера через IntelliJ IDEA и Docker.....	2
Docker (Windows PowerShell).....	2
IntelliJ IDEA - клонирование репозитория на компьютер.....	3
IntelliJ IDEA - настройка ресурсов.....	5
IntelliJ IDEA - настройка базы данных.....	6
Загрузка HTML веб-страниц.....	12
Работа веб-приложения.....	14
Авторизация / регистрация.....	14
Загрузка файла и его воспроизведение стримом.....	15
Устройство базы данных.....	17
Даталогическая модель базы данных.....	17
Скрипт для сброса всех данных Базы Данных.....	17

Полная инструкция по запуску сервера через IntelliJ IDEA и Docker

Docker (Windows PowerShell)

1. В поисковой строке Windows пишем PowerShell
2. Открываем Windows PowerShell от имени администратора



3. Прописываем команду `Get-Service -Name *docker*`
4. Получаем информацию о нашем Docker

```
PS C:\WINDOWS\system32> Get-Service -Name *docker*

Status      Name                DisplayName
-----
Stopped     com.docker.service Docker Desktop Service
```

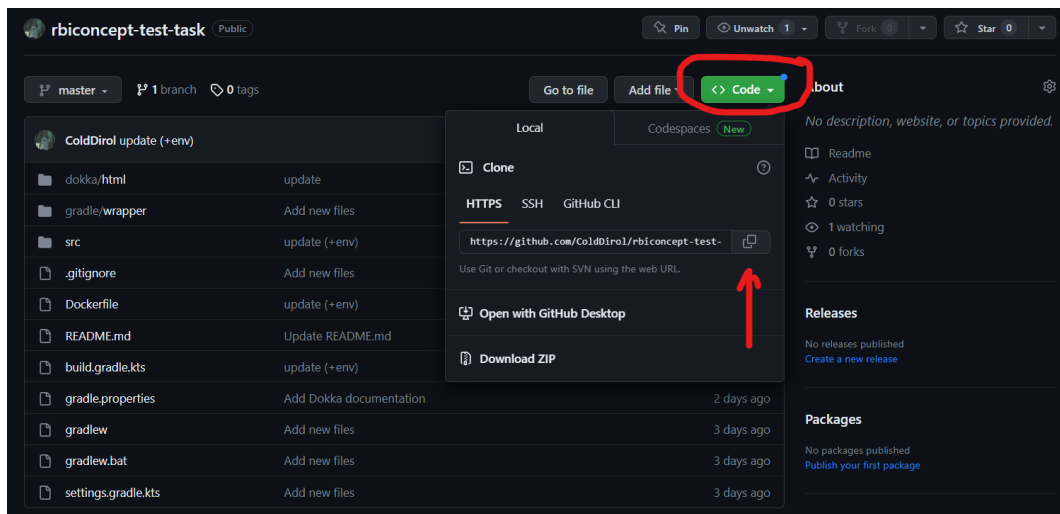
5. ! Если в поле Status прописано **Stopped** - переходим к 6 шагу
! Если в поле Status прописано **Running** - переходим к 7 шагу
6. Только для Stopped:
Запускаем Docker командой `Start-Service -Name *docker*`
7. Запускаем приложение Docker Desktop и возвращаемся обратно к PowerShell
8. Устанавливаем образ Minio S3 командой `docker pull minio/minio`
9. Запускаем образ Minio S3 на порту **9000** с логином - **colddirol** и паролем - **colddirol** (можно указать другие данные) командой (необходимо прописать вручную из-за переноса строки):

```
docker run -p 9000:9000 --name minio-s3 -e "MINIO_ACCESS_KEY=colddirol" -e "MINIO_SECRET_KEY=colddirol" -d minio/minio server /data
```

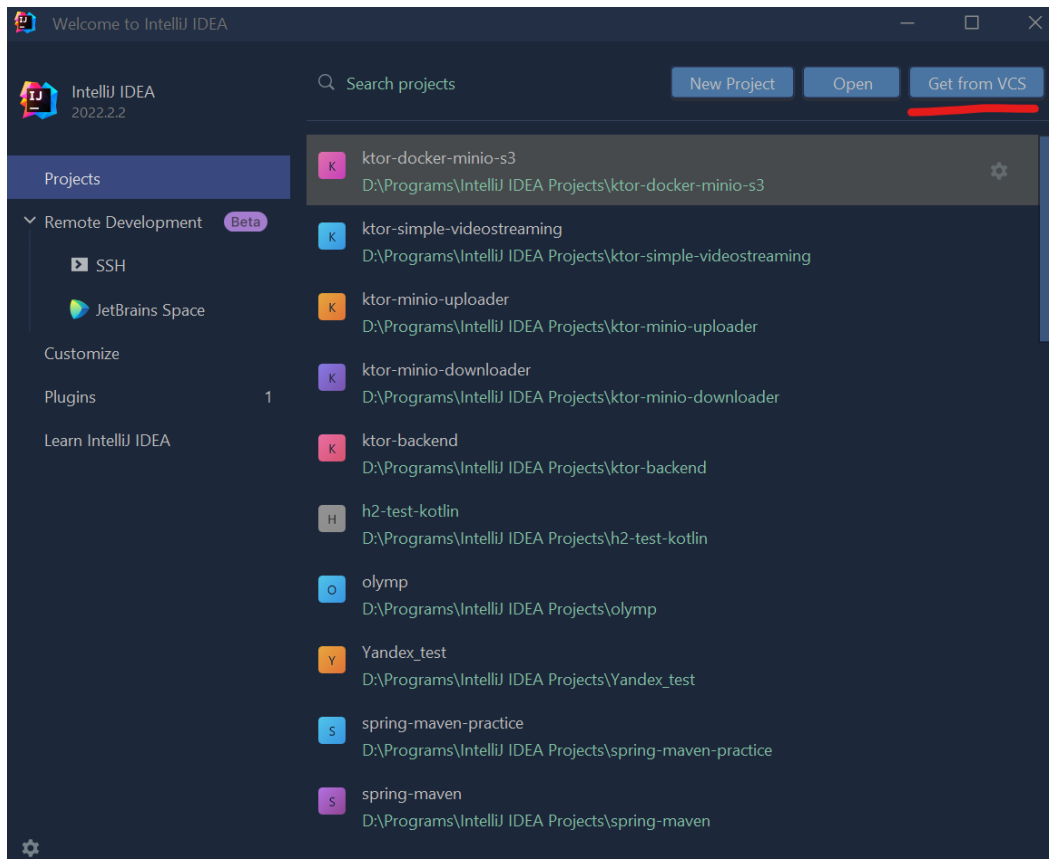
10. Создавать bucket не нужно, переходим к открытию проекта в **IntelliJ IDEA**

IntelliJ IDEA - клонирование репозитория на компьютер

1. Открываем проект в [rbiconcept-test-task](#) в GitHub
2. Нажимаем на зеленую кнопку **Code** -> копируем URL адрес

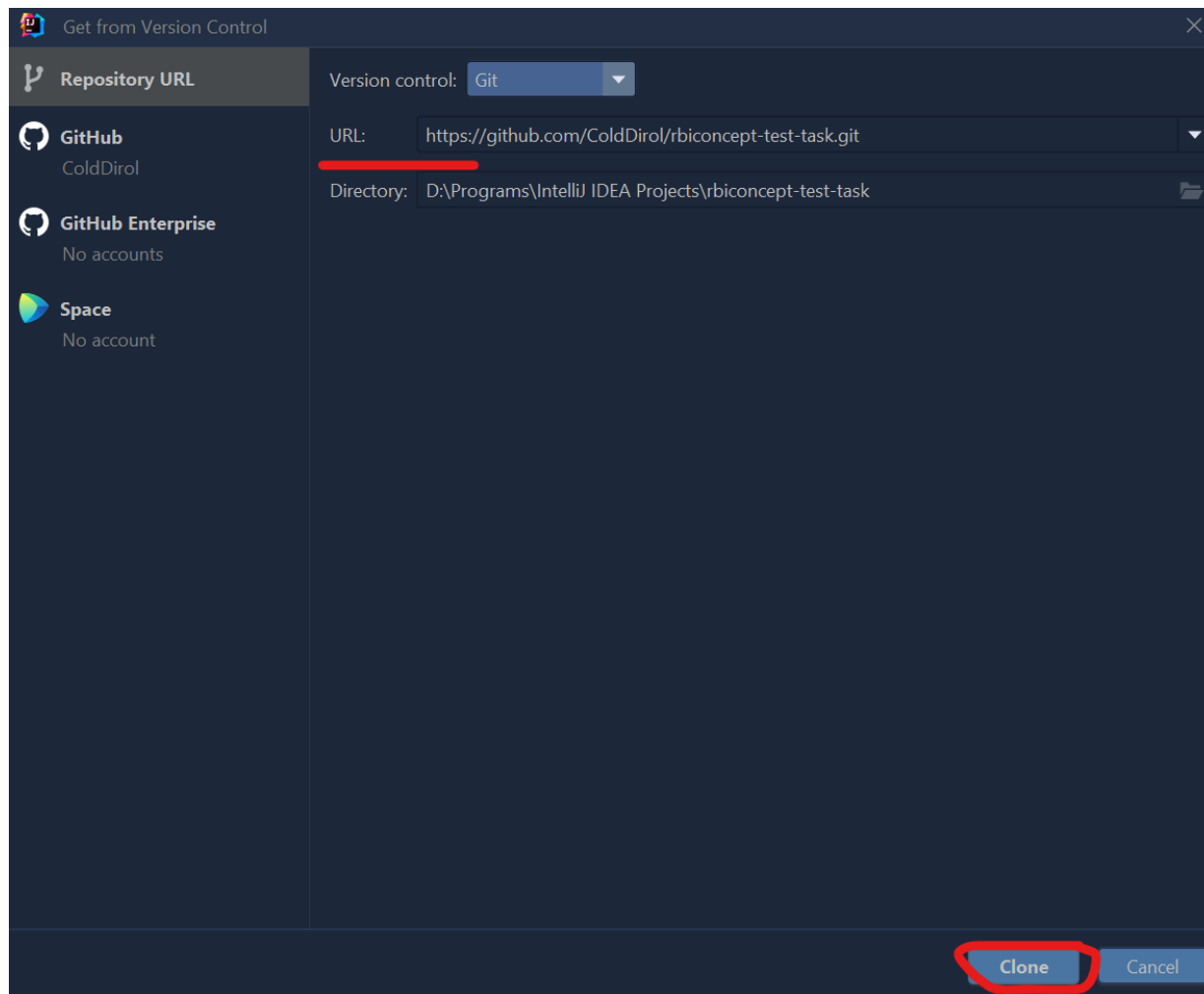


3. Открываем стартовую страницу IntelliJ IDEA
4. Нажимаем кнопку **Get from VCS**



5. В поле URL вставляем ранее скопированную ссылку с GitHub

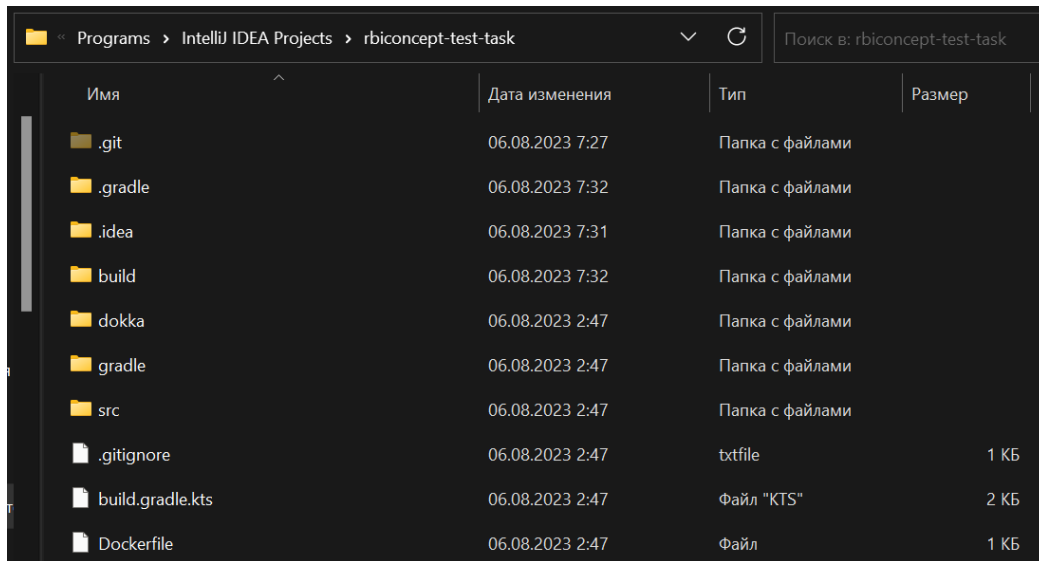
6. Нажимаем Clone



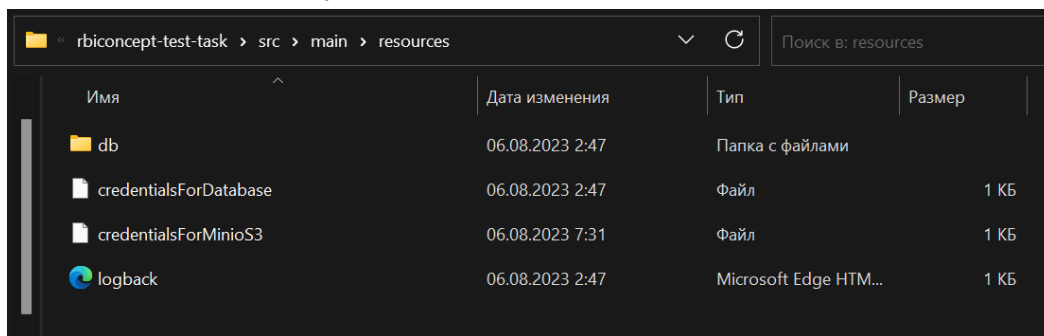
7. Проект открылся.

IntelliJ IDEA - настройка ресурсов

1. Открываем папку проекта в файловом-менеджере

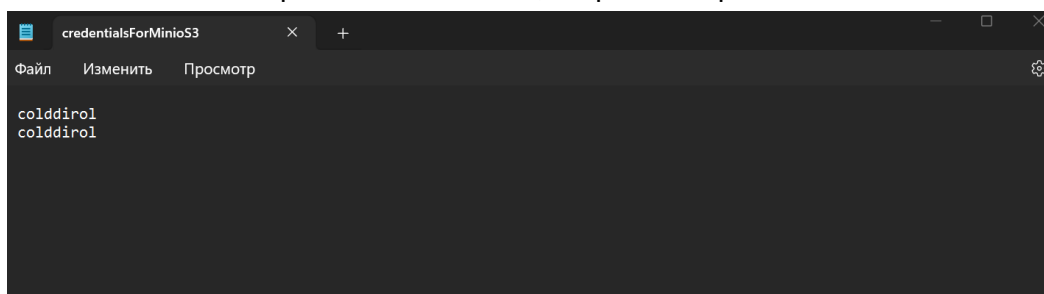


2. Открываем папку src -> main -> resources

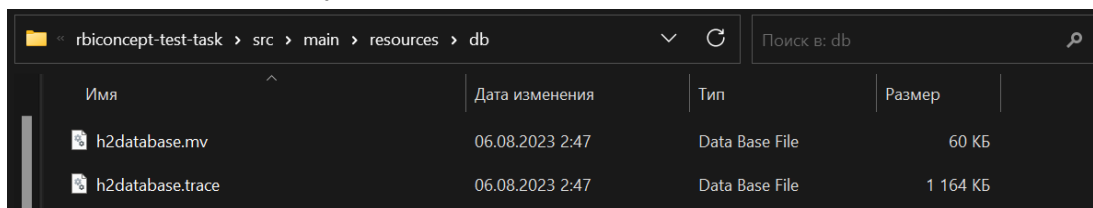


3. *Если вдруг при запуске образа Minio S3 вы указали свои данные для входа:

1. Открываем файл **credentialsForMinioS3** через блокнот;
2. Меняем первую строку на Ваш логин (указано в ключе MINIO_ACCESS_KEY);
3. Вторую строку меняем на Ваш пароль (указано в ключе MINIO_SECRET_KEY);
4. Сохраняем изменения, закрываем файл

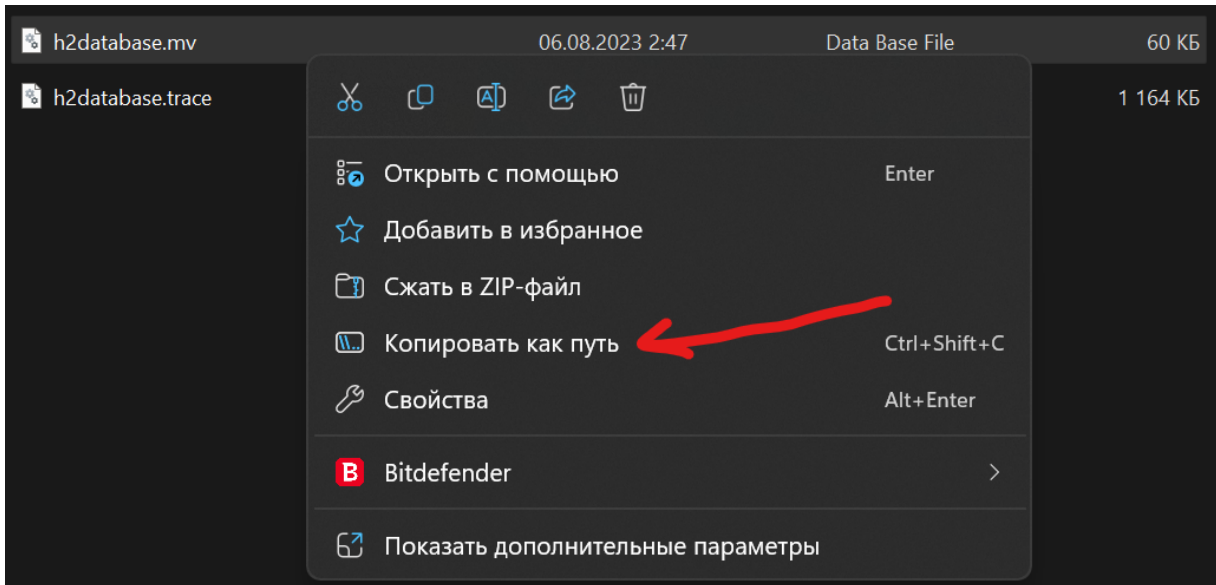


4. Открываем папку db



5. Нажимаем правой кнопкой мыши на файл базы данных h2database.mv

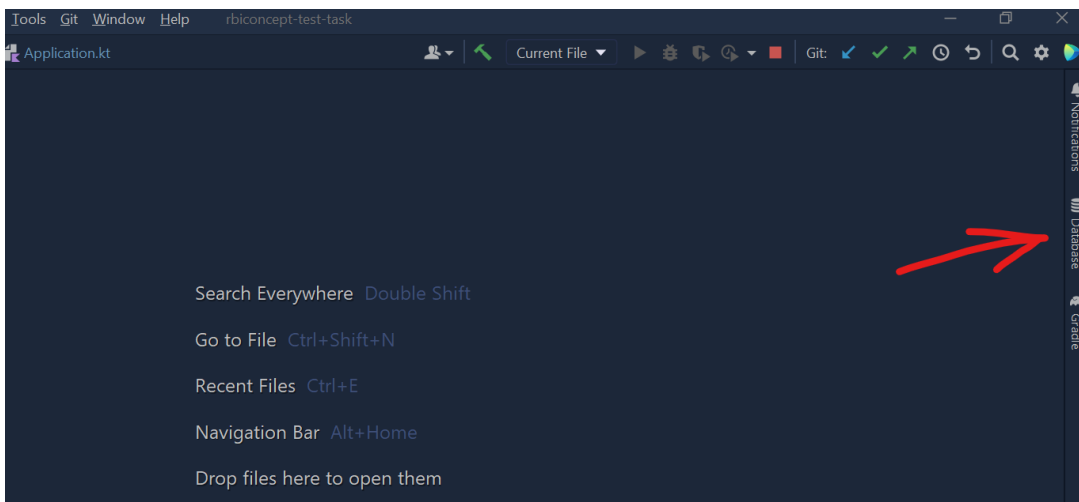
6. Нажимаем “Копировать как путь”



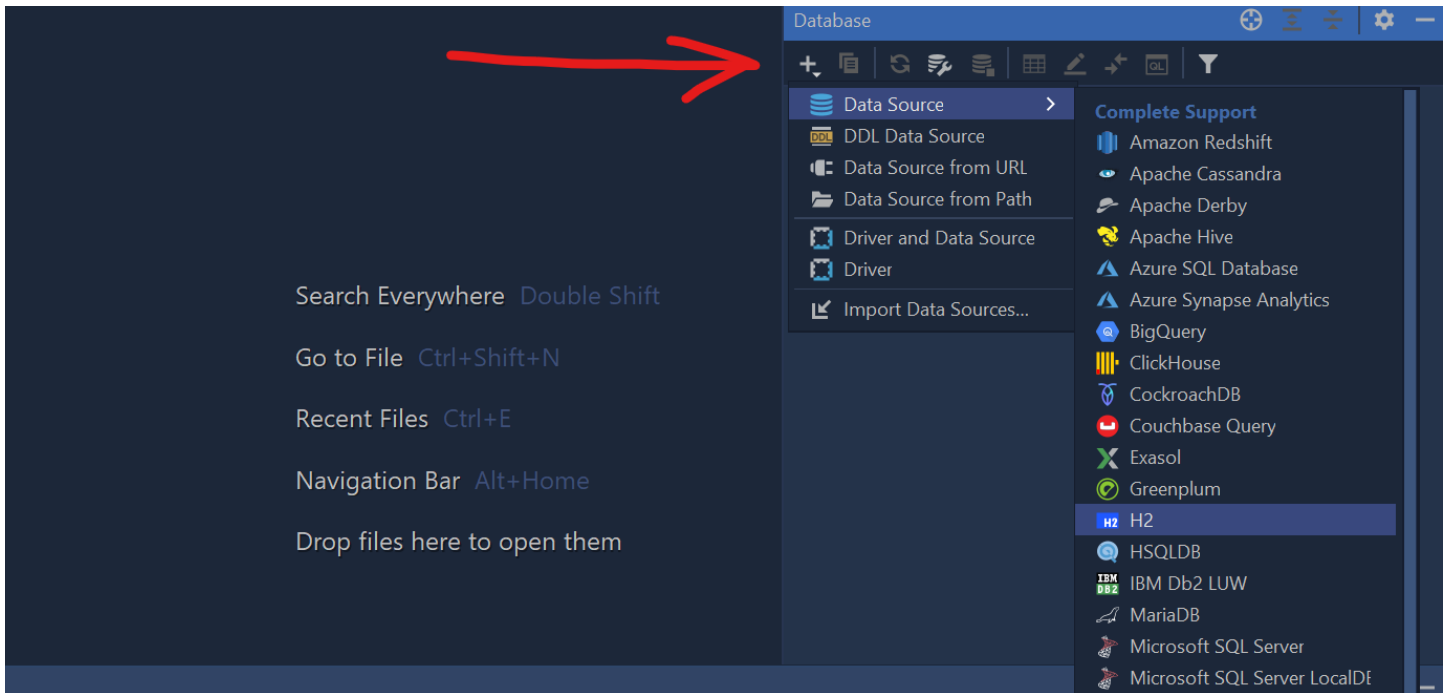
7. Возвращаемся в IntelliJ IDEA

IntelliJ IDEA - настройка базы данных

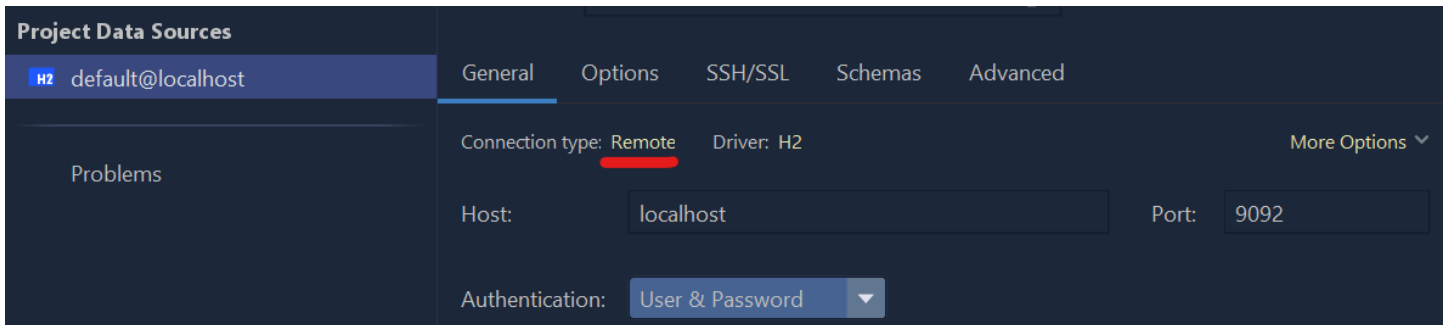
1. В правой части экрана нажимаем на окно Database



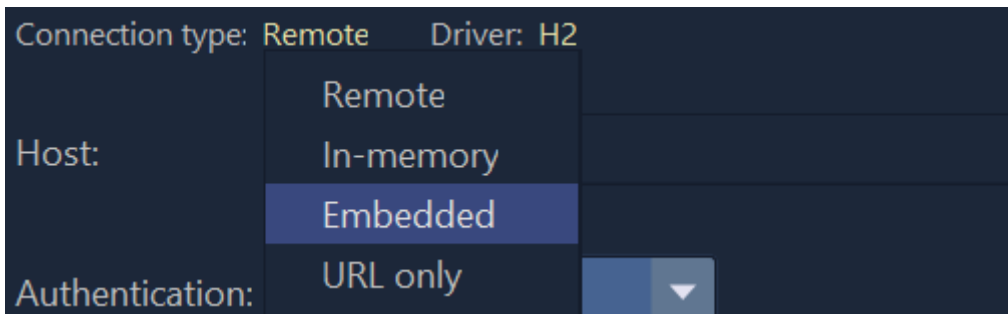
2. В открывшемся окне нажимаем на + -> Data Source -> H2



3. В открывшемся окне нажимаем на Remote в поле Connection type:



4. В открывшемся списке выберем Embedded:



5. Далее в поле Path нам необходимо вставить скопированный путь нашей базы данных (скопировали в предыдущем параграфе)

!!! важно убедиться, чтобы он был без кавычек. Если кавычки все-таки есть - удаляем.

Connection type: Embedded Driver: H2 More Options ▾

Path: `IDEA Projects\rbiconcept-test-task\src\main\resources\db\h2database.mv.db` ...

Authentication: User & Password ▾

User:

Password: Save: Forever ▾

URL: `IntelliJ IDEA Projects\rbiconcept-test-task\src\main\resources\db\h2database` ↗
Overrides settings above

6. В **user** вписываем **colddirol**
В **password** вписываем **colddirol**
Нажимаем **Test connection**

Problems

Connection type: Embedded Driver: H2 More Options ▾

Path: `IDEA Projects\rbiconcept-test-task\src\main\resources\db\h2database.mv.db` ...

Authentication: User & Password ▾

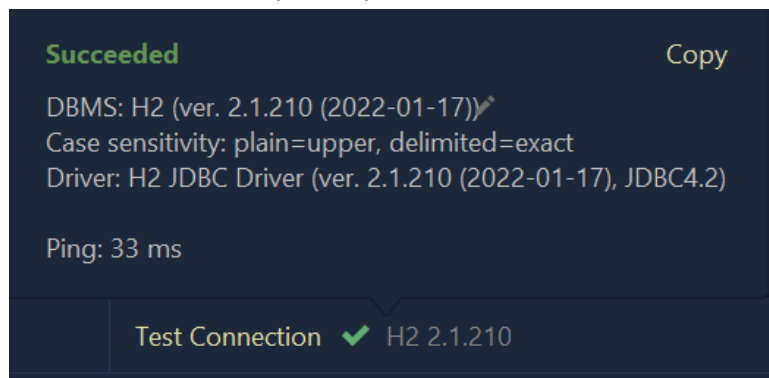
User: `colddirol`

Password: Save: Forever ▾

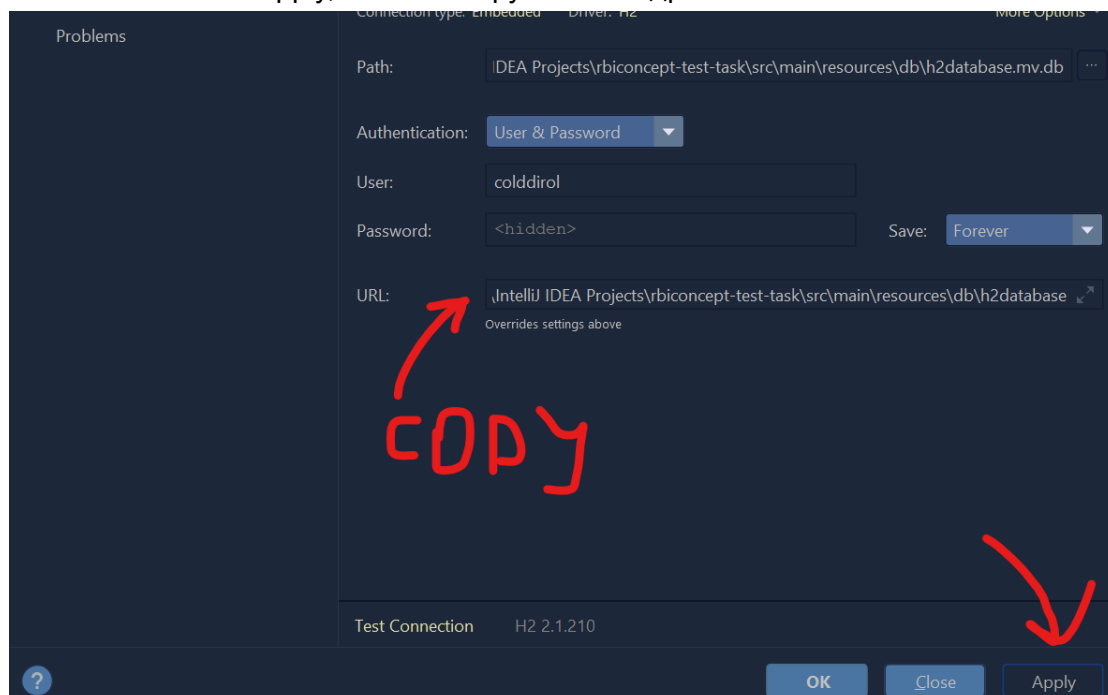
URL: `IntelliJ IDEA Projects\rbiconcept-test-task\src\main\resources\db\h2database` ↗
Overrides settings above

Test Connection H2 ↗

7. Если мы получили уведомление о том, что подключение выполнено успешно - идем дальше

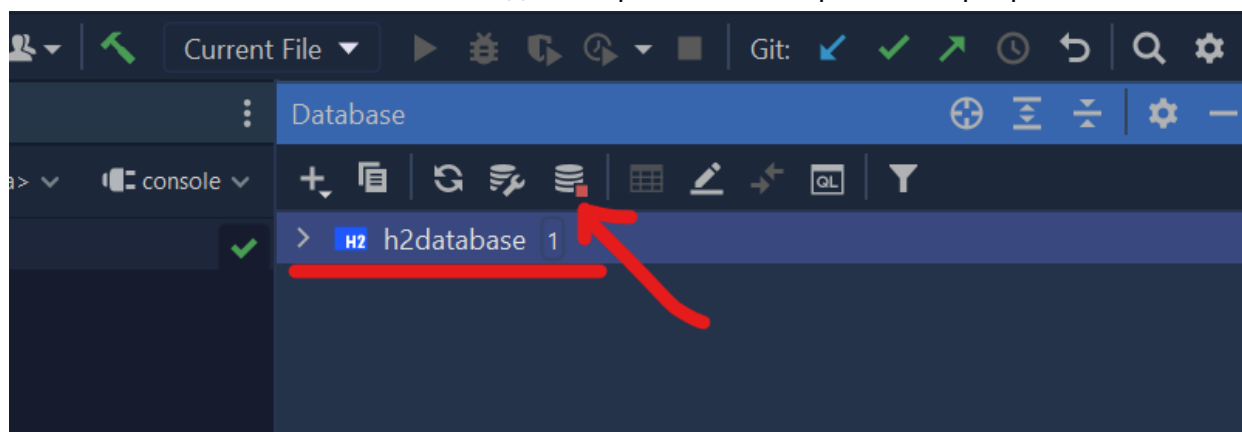


8. Нажимаем Apply, затем копируем URL адрес



9. Закрываем окно, нажав на кнопку OK.

10. Обязательно отключаемся от нашей базы данных выбрав h2database и нажав кнопку Deactivate, так как наша база данных работает не в режиме сервера



11. Переходим к проекту:
12. Открываем Application.kt:
src -> main -> kotlin -> com.colddirol -> **Application.kt**
13. В функции **main()** заменяем значение поля **URL** на наш скопированный путь

```
27 fun main() {  
28     try {  
29         CredentialsService.readCredentials()  
30  
31         val URL = "your-URL"  
32  
33         Database.connect(  
34             url: "$URL",  
35             driver = "org.h2.Driver",  
36             user = DatabaseCredentials.username,  
37             password = DatabaseCredentials.password  
38         )  
39     }
```

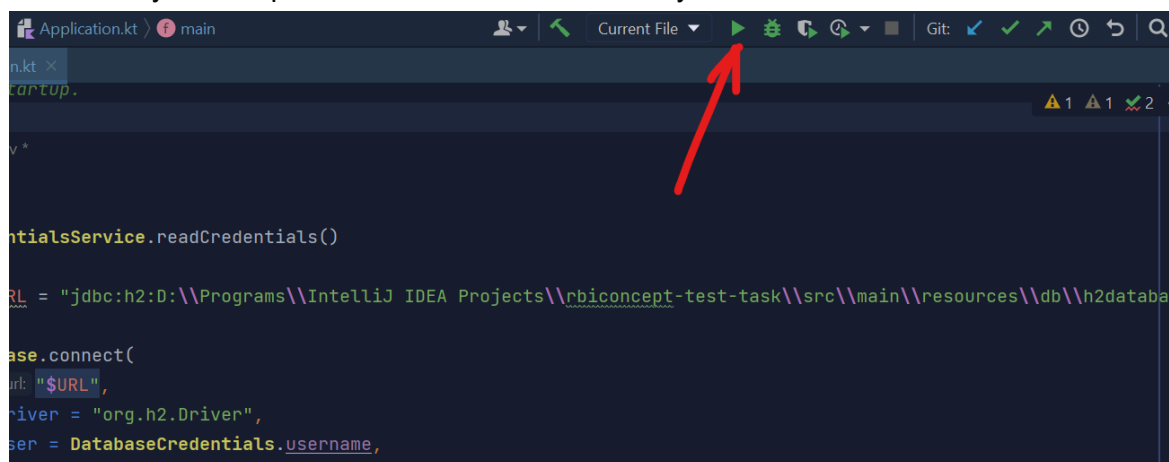
14. Должно получиться так:

```
Vladimir Kartashev *  
fun main() {  
    try {  
        CredentialsService.readCredentials()  
  
        val URL = "jdbc:h2:D:\\Programs\\IntelliJ IDEA Projects\\rbiconcept-test-task\\src\\main\\resources\\db\\h2database"  
  
        Database.connect(  
            url: "$URL",  
            driver = "org.h2.Driver",  
            user = DatabaseCredentials.username,  
            password = DatabaseCredentials.password  
        )  
    }
```

15. Переходим к следующему шагу

IntelliJ IDEA - настройка базы данных

1. Запускаем приложение, нажав на кнопку Run

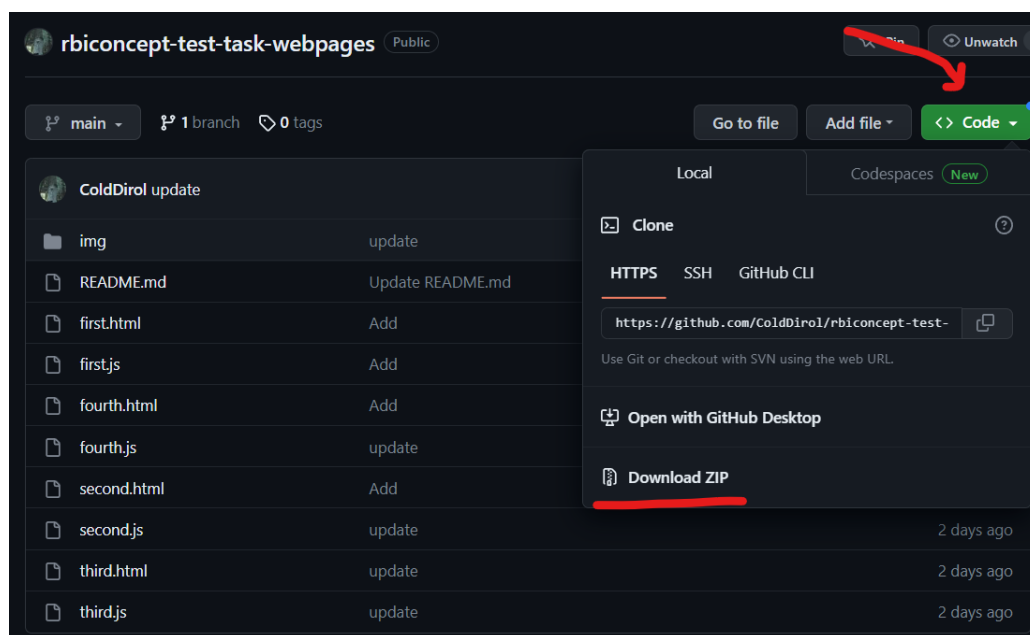


2. Если в консоли Run Вы видите сообщения без ошибок и исключений - все получилось.

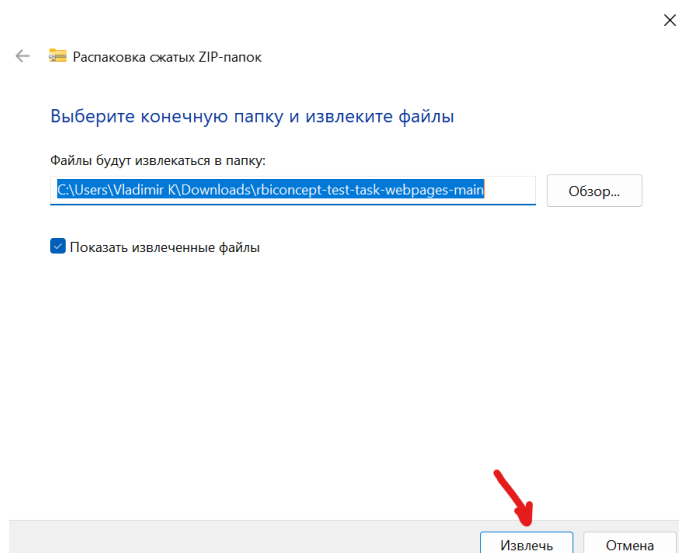
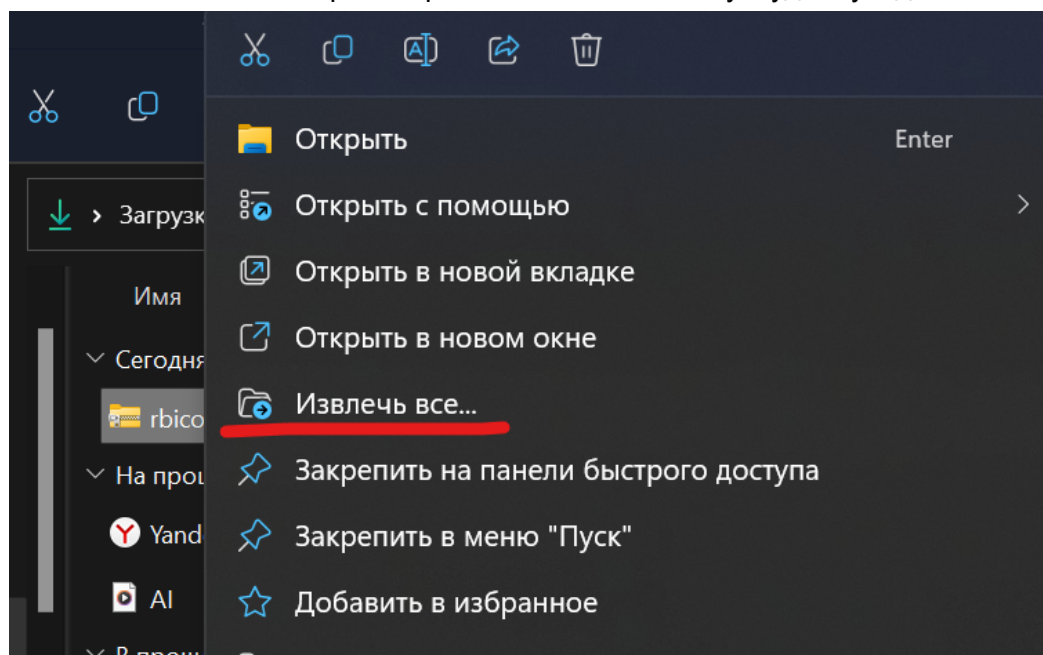
```
"C:\Program Files\Java\jdk-17.0.4.1\bin\java.exe" ...
Credentials for Database has been set
Credentials for Minio S3 has been set
2023-08-06 08:23:10.339 [DefaultDispatcher-worker-1] INFO ktor.application - Autoreload is disabled because the development mode is off.
2023-08-06 08:23:10.715 [DefaultDispatcher-worker-1] INFO ktor.application - Application started in 0.437 seconds.
2023-08-06 08:23:10.761 [DefaultDispatcher-worker-5] INFO ktor.application - Responding at http://127.0.0.1:8080
```

Загрузка HTML веб-страниц

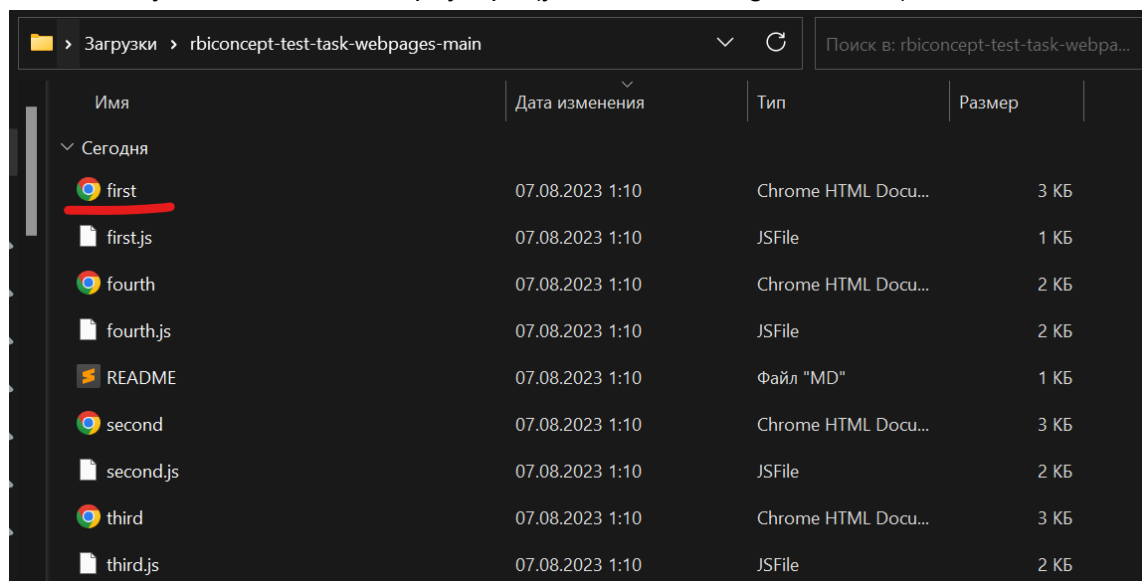
1. Переходим в репозиторий [rbiconcept-test-task-webpages](https://github.com/ColdDirol/rbiconcept-test-task-webpages)
2. Нажимаем **Code** -> **Download ZIP**



3. Скачиваем ZIP архив и распаковываем в любую удобную для вас папку

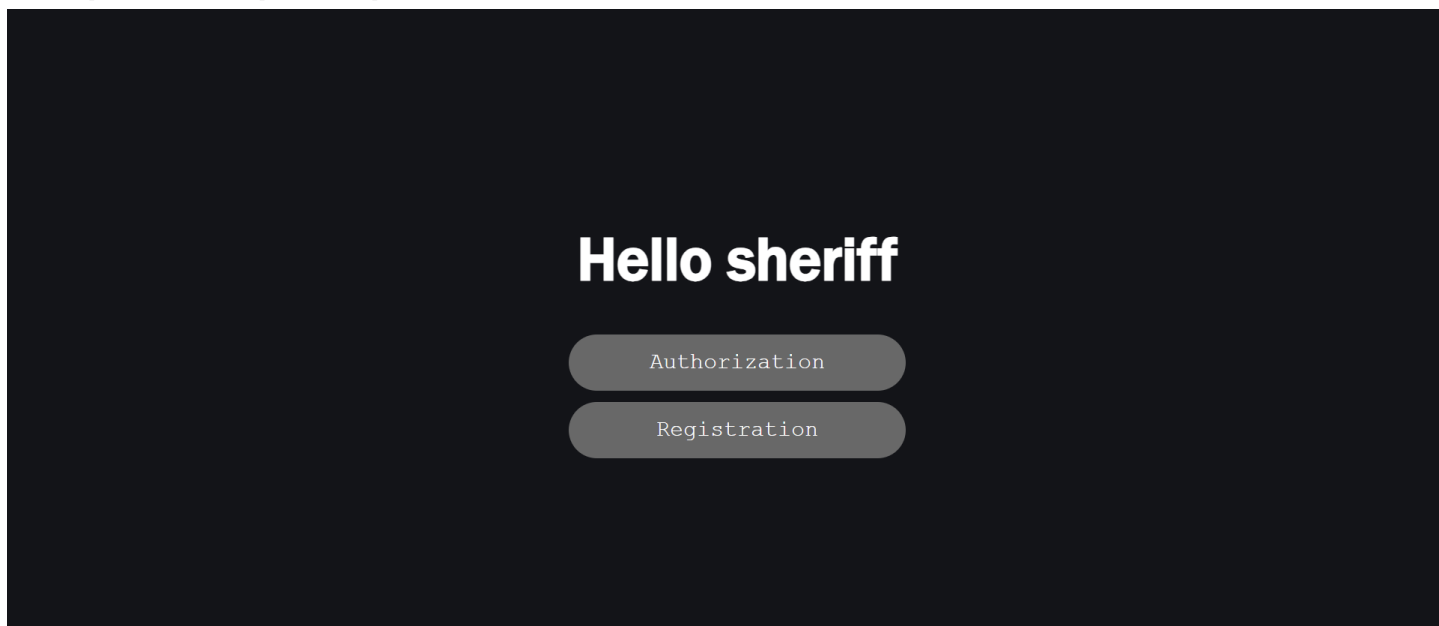


4. Открываем папку, в которую извлекли наш ZIP архив
5. Убеждаемся, что наш Minio S3 контейнер запущен локально в Docker на порту 9000
6. Убеждаемся, что наш Ktor Kotlin сервер запущен локально на порту 8080
7. Запускаем first.html в браузере (у меня это Google Chrome)



Работа веб-приложения

Авторизация / регистрация

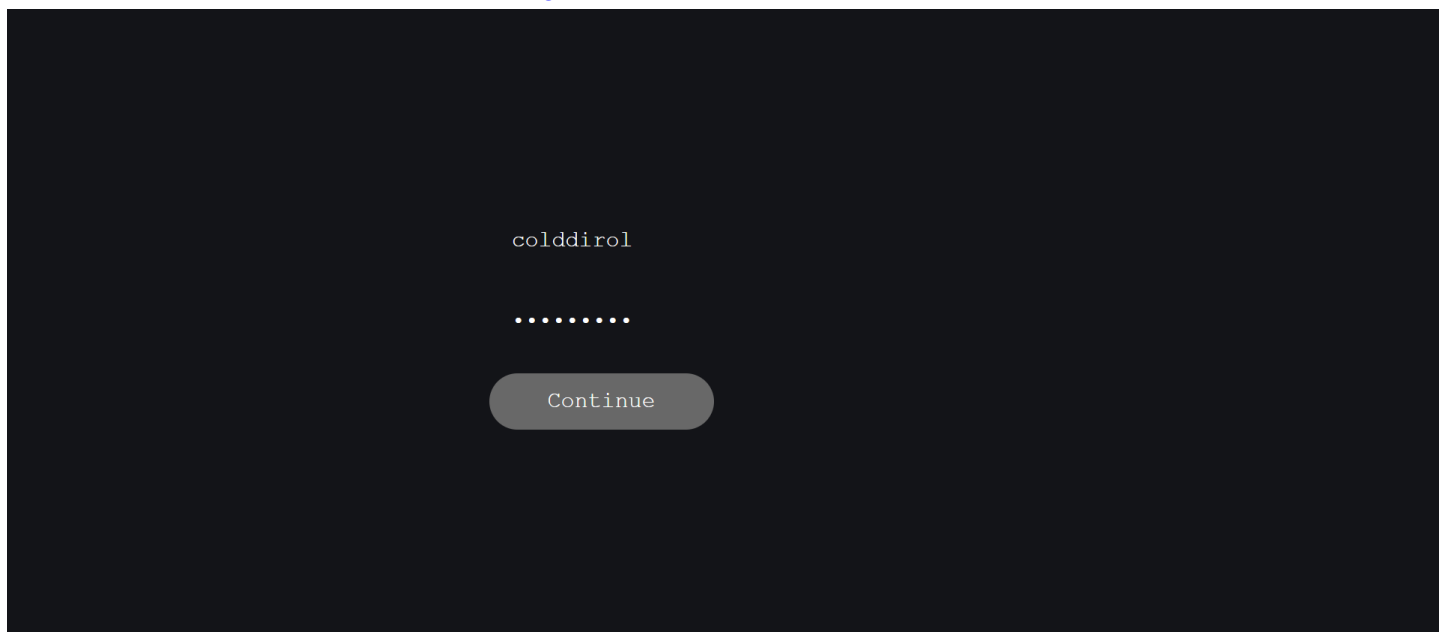


При нажатии **Authorization** мы отправляемся на страницу **second.html**, передавая ей мод **login**:

file:///YOUR-PATH/second.html?mode=login

При нажатии **Registration** мы отправляемся на страницу **second.html**, передавая ей мод **registration**:

file:///YOUR-PATH/second.html?mode=registration

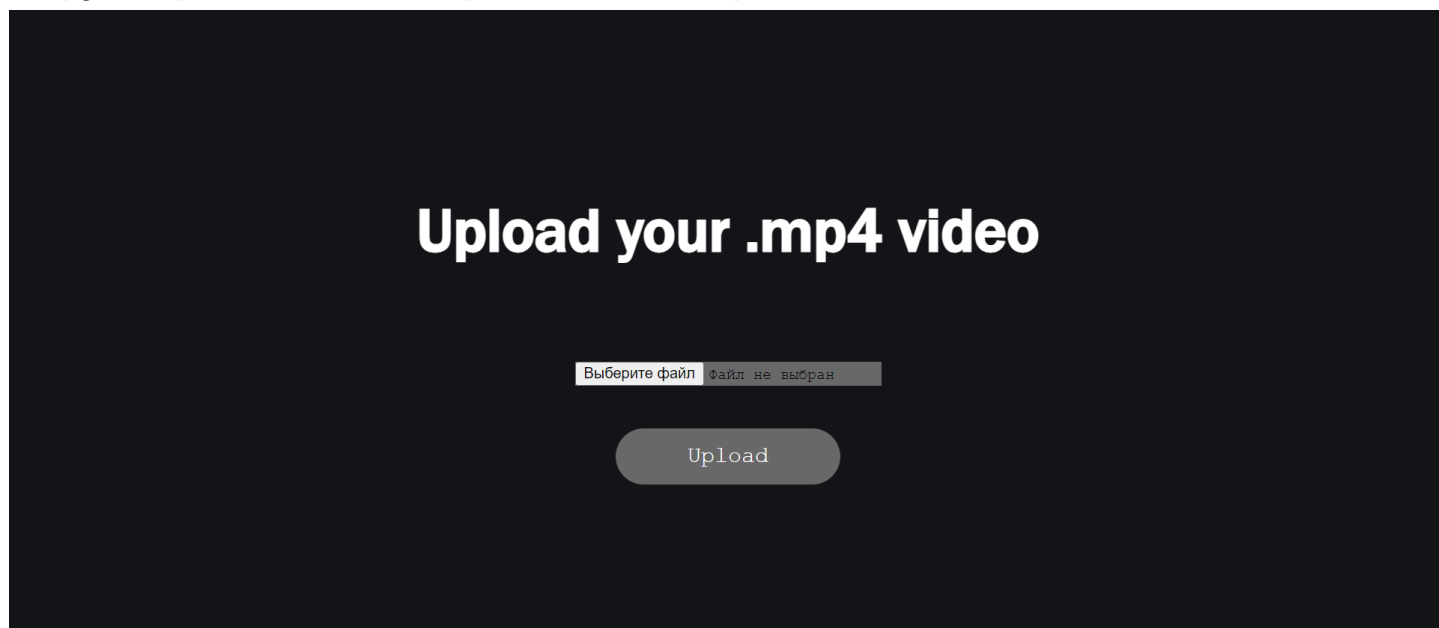


Далее вводим наши креды и нажимаем continue.

Если мы перешли на следующую страницу - Поздравляю, все сделано верно

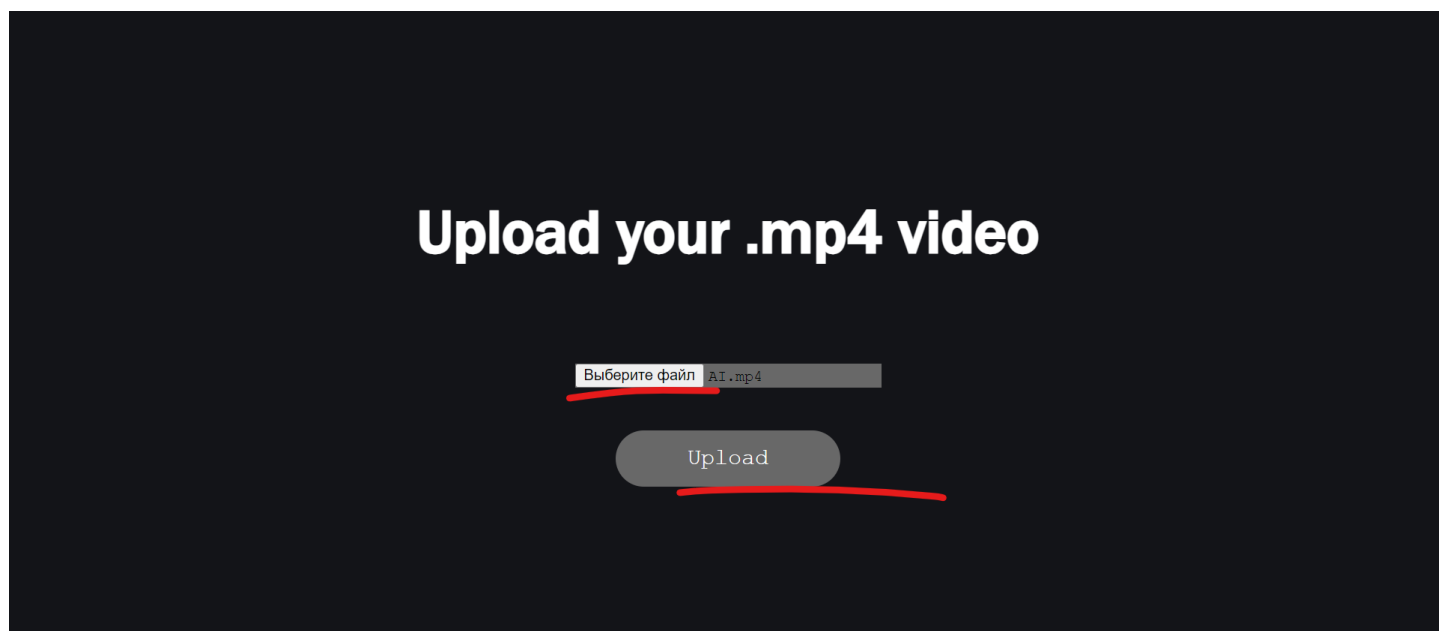
Если же нет - смотрим на исключения в консоли IntelliJ IDEA (если при запуске исключений не возникло - скорее всего, база данных была подключена неправильно)

Загрузка файла и его воспроизведение стримом

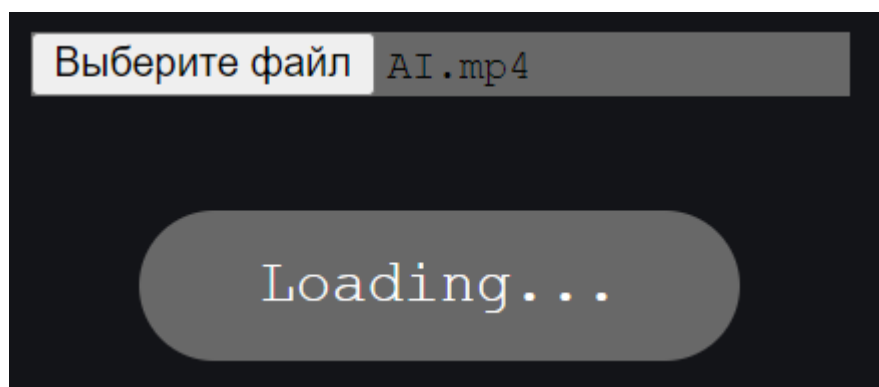


Загружаем наш большой mp4 файл (на страницу можно загрузить только mp4 файл. Если загрузить файл другого формата - сервер обработает его и выбросит исключение).

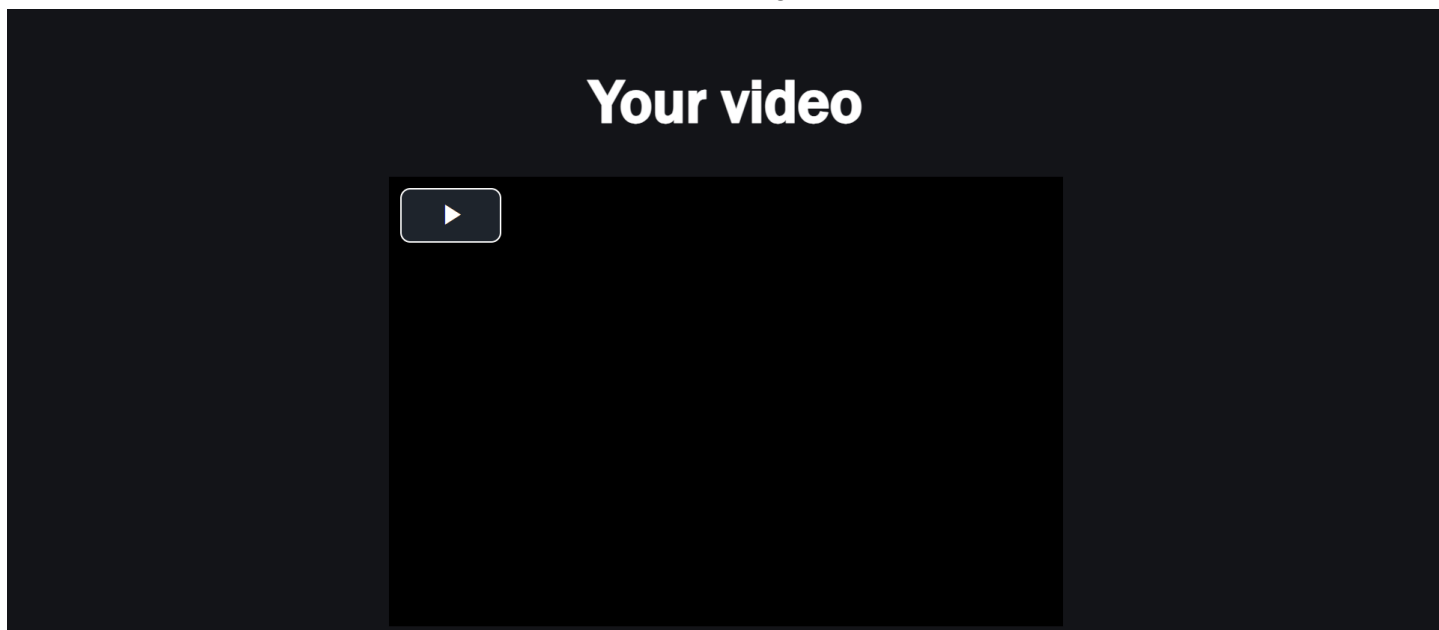
Я буду загружать фильм Искусственный Разум 2001 года на 1.02 Гб (для удобства создал папку с большими видеофайлами - [ссылка](#))



Выбираем файл и нажимаем **Upload**.

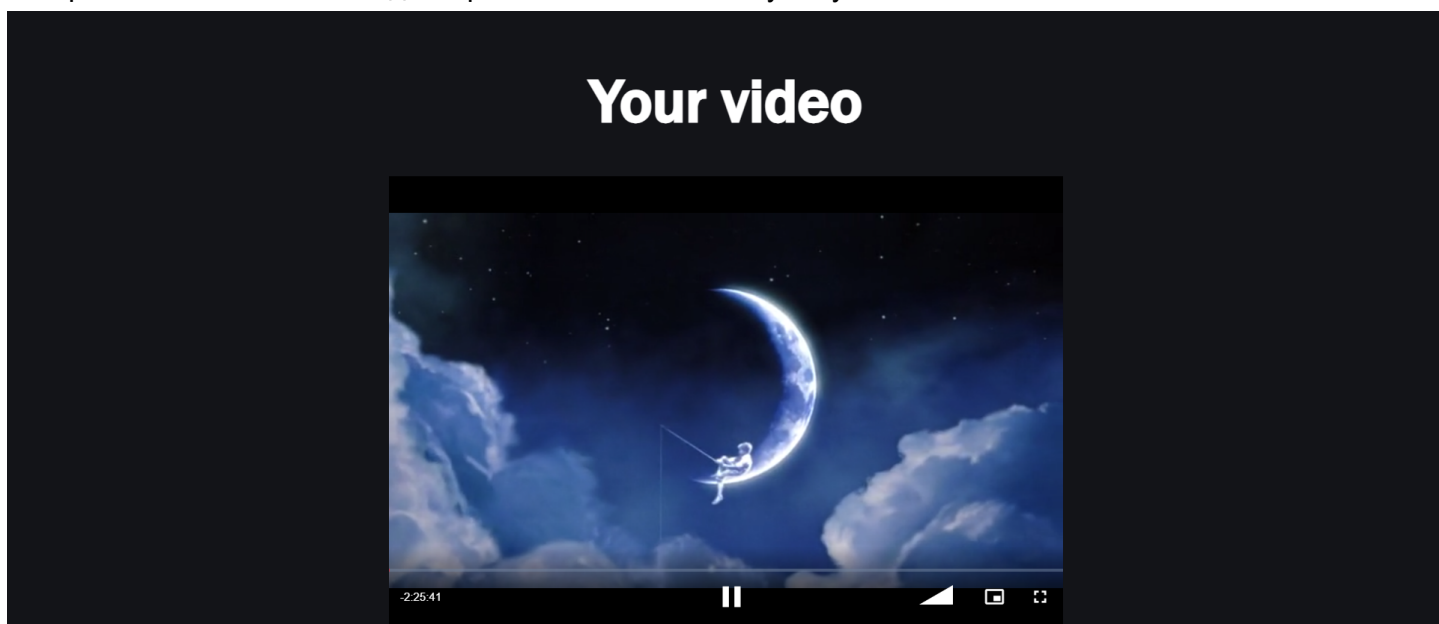


После нажатия кнопка меняет надпись Upload на Loading...



Через несколько секунд видео загрузилось

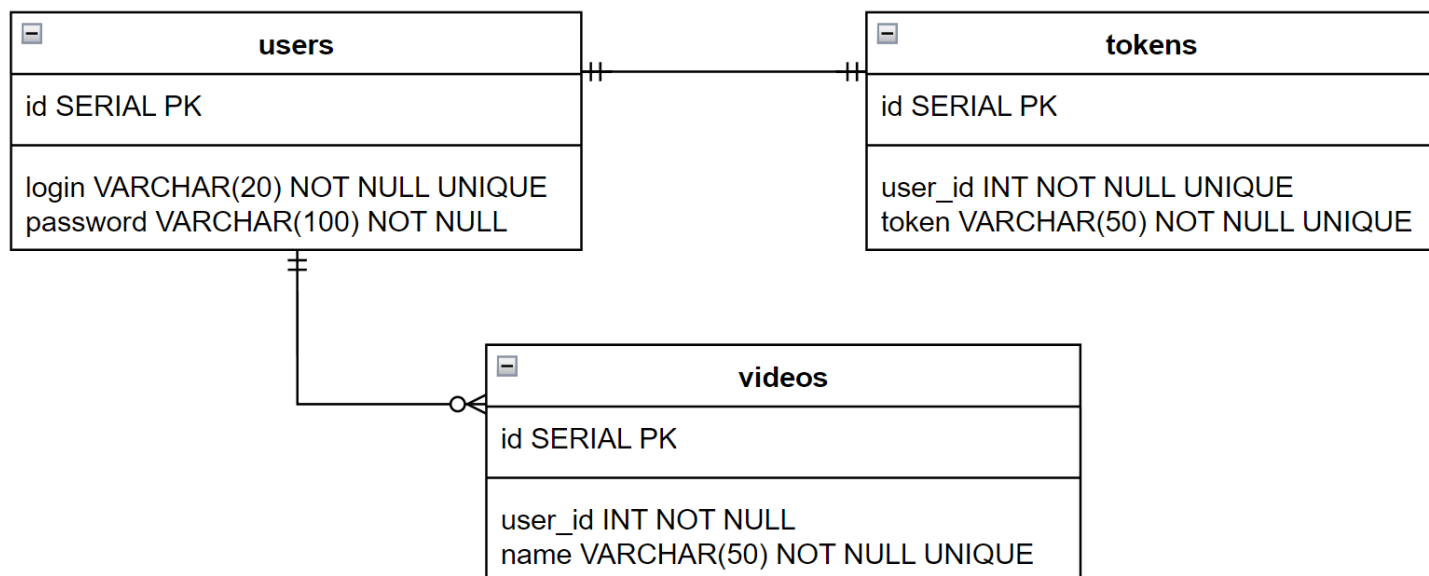
Теперь мы можем начать видеострим нажав на кнопку Play.



Устройство базы данных

Даталогическая модель базы данных

(Синтаксис PostgreSQL)



Скрипт для сброса всех данных Базы Данных

(Синтаксис H2)

```

DROP TABLE users;
DROP TABLE tokens;
DROP TABLE videos;

CREATE TABLE IF NOT EXISTS users
(
    id INT AUTO_INCREMENT PRIMARY KEY,
    login VARCHAR(20) NOT NULL UNIQUE,
    password VARCHAR(100) NOT NULL
);

CREATE TABLE IF NOT EXISTS tokens
(
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    token VARCHAR(50) NOT NULL
);

CREATE TABLE IF NOT EXISTS videos
(
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    name VARCHAR NOT NULL
);
  
```