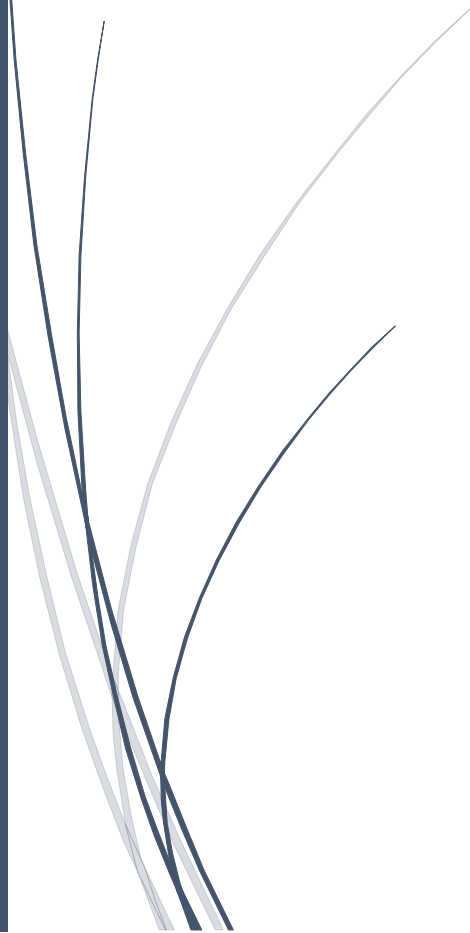


A dark blue vertical bar on the left side of the page. A blue arrow points to the right from this bar, containing the date.

11.12.2017

Настройка среды разработки

Система бронирования
отпусков РТИ

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

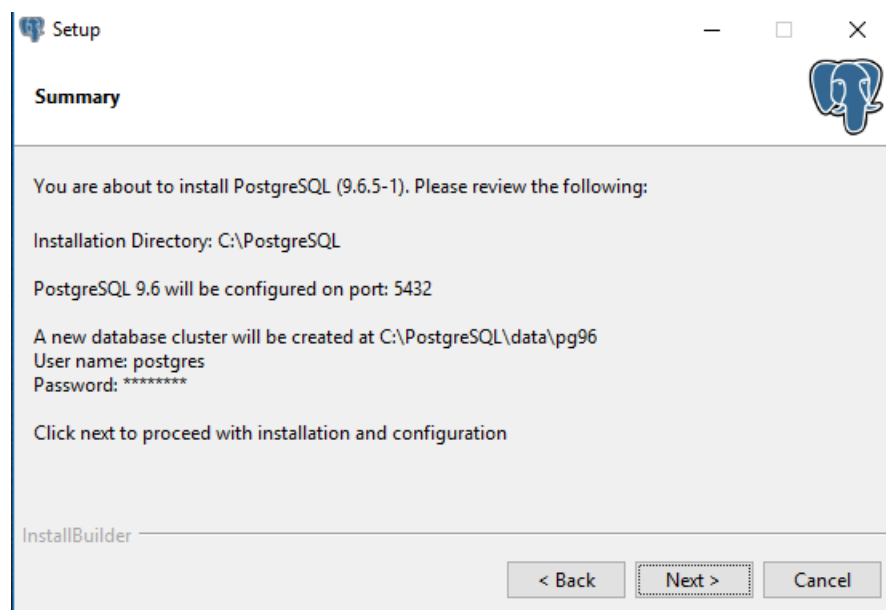
Абрамкин Максим Андреевич
ООО «РОСТЕЛЕКОМ-ИНТЕГРАЦИЯ»

Оглавление

Установка необходимого ПО и настройка окружения	2
Создание локального репозитория Git	3
Настройка подключения к локальной БД PostgreSQL в DBeaver	5
Создание новой роли 'holidays' и новой базы данных 'holidays'	13
Запуск приложения из IntelliJ IDEA в режиме отладки кода (Debug)	15
Прекращение/останов отладки приложения	19
Запуск приложения из IntelliJ IDEA без режима отладки (Run)	20
Описание доступных конфигураций запуска и отладки	22

Установка необходимого ПО и настройка окружения

1. Устанавливаем **JDK 1.8.0_152** (не ниже) с официального сайта Oracle – <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>. **Обращаем внимание на разрядность дистрибутива – он должен соответствовать версии Вашей ОС (x86 или x64).**
2. Устанавливаем инструменты для работы с Git – **git-gui**, **gitk** с сайта <https://git-scm.com/downloads>
3. Устанавливаем **TortoiseGit** с сайта <https://tortoisegit.org/>
4. Устанавливаем локально **PostgreSQL** с сайта <https://www.openscg.com/bigsql/postgresql/installers.jsp/>. При установке PostgreSQL необходимо будет задать пользователя для базы данных, в данном случае оставляем настройки и пользователя по умолчанию (**postgres**). Порт также оставляем по умолчанию равным **5432**. Перед началом процесса установки параметры будут следующими:



5. Устанавливаем **Maven** версии не ниже 3.5.0 с сайта <https://maven.apache.org/download.cgi>. В целом, шаг не обязательный, т.к. у IDEA, которую будем ставить дальше, есть встроенный плагин, поэтому шаги 5-6 выполняем в случае желания отдельной установки Maven.
6. Устанавливаем бесплатный клиент для БД **DBeaver**, в нашем случае для подключения к локальной базе данных PostgreSQL – <https://dbeaver.jkiss.org/download/>. Программа написана на Java и требует для запуска JRE. Дистрибутив представлен в двух вариантах: портативный (ZIP-архив, не включающий в себя JRE) и установщик Windows. Можно выбрать любой удобный для Вас вариант. **Обращаем внимание на разрядность дистрибутива – он должен соответствовать версии Вашей ОС (x86 или x64).**
7. Распаковываем zip-архив с Maven, скачанный в п.5 и прописываем следующие значения переменные окружения:

```
M2_HOME=C:\Program Files\Apache\Maven3.5.0
MAVEN_HOME=C:\Program Files\Apache\Maven3.5.0
```

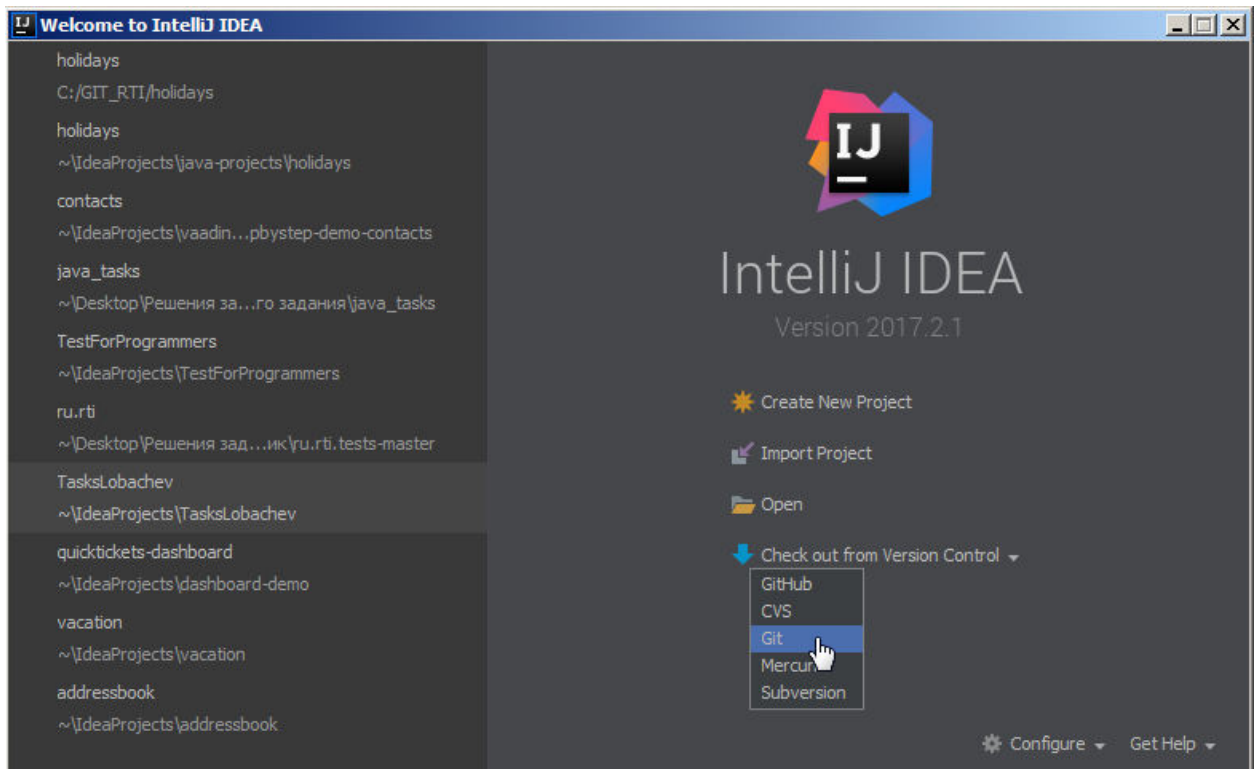
8. Устанавливаем **IntelliJ IDEA Community Edition** версии не ниже 2017.2.1 (Build #IC-172.3544.35) – официальный сайт: <https://www.jetbrains.com/idea/>:



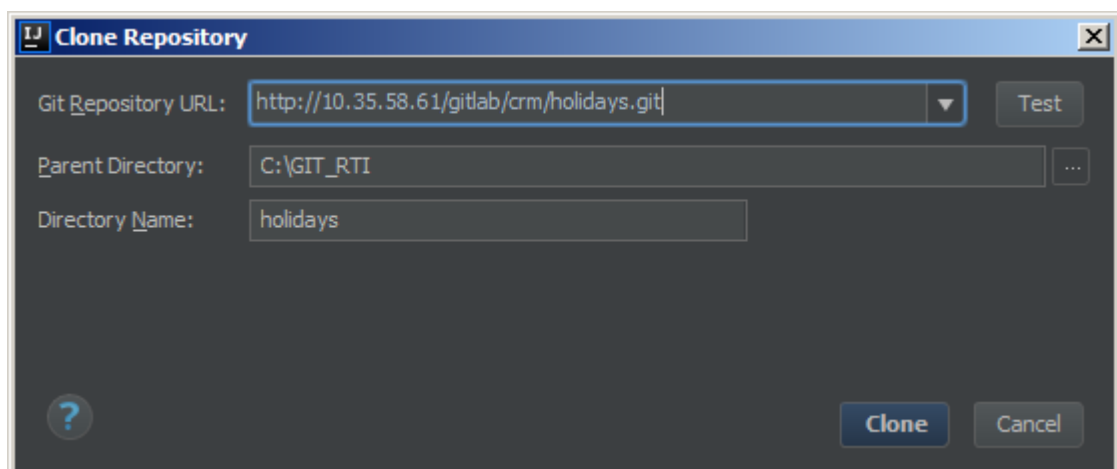
Обращаем внимание на разрядность дистрибутива – он должен соответствовать версии Вашей ОС (x86 или x64).

Создание локального репозитория Git

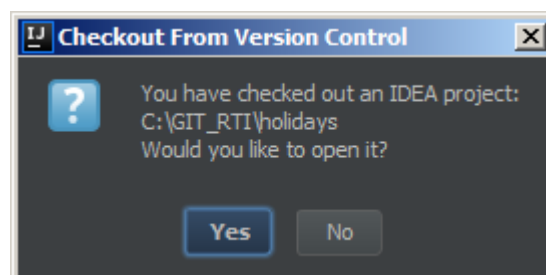
1. Создаём на локальном диске каталог для хранения локального репозитория. В данной инструкции предполагается, что локальный репозиторий располагается в **C:\GIT_RTI**. В качестве альтернативного варианта, Вы можете использовать стандартное местоположение Workspace IntelliJ IDEA - **C:\Users\<Ваш_доменный_логин_Windows>\IdeaProjects**, например, в моём случае это каталог **C:\Users\Maksim.Abramkin\IdeaProjects**. В любом из случаев далее в инструкции мы будем ссылаться на этот каталог, именуя его *каталогом локального репозитория*.
2. Запускаем IntelliJ IDEA и в стартовом окне выбираем **"Check out from Version Control" -> Git**



3. В открывшемся окне указываем путь к репозиторию Git, где хранятся исходные коды проекта системы бронирования отпусков РТИ: <http://10.35.58.61/gitlab/crm/holidays.git>, после чего нажимаем кнопку **“Clone”**:

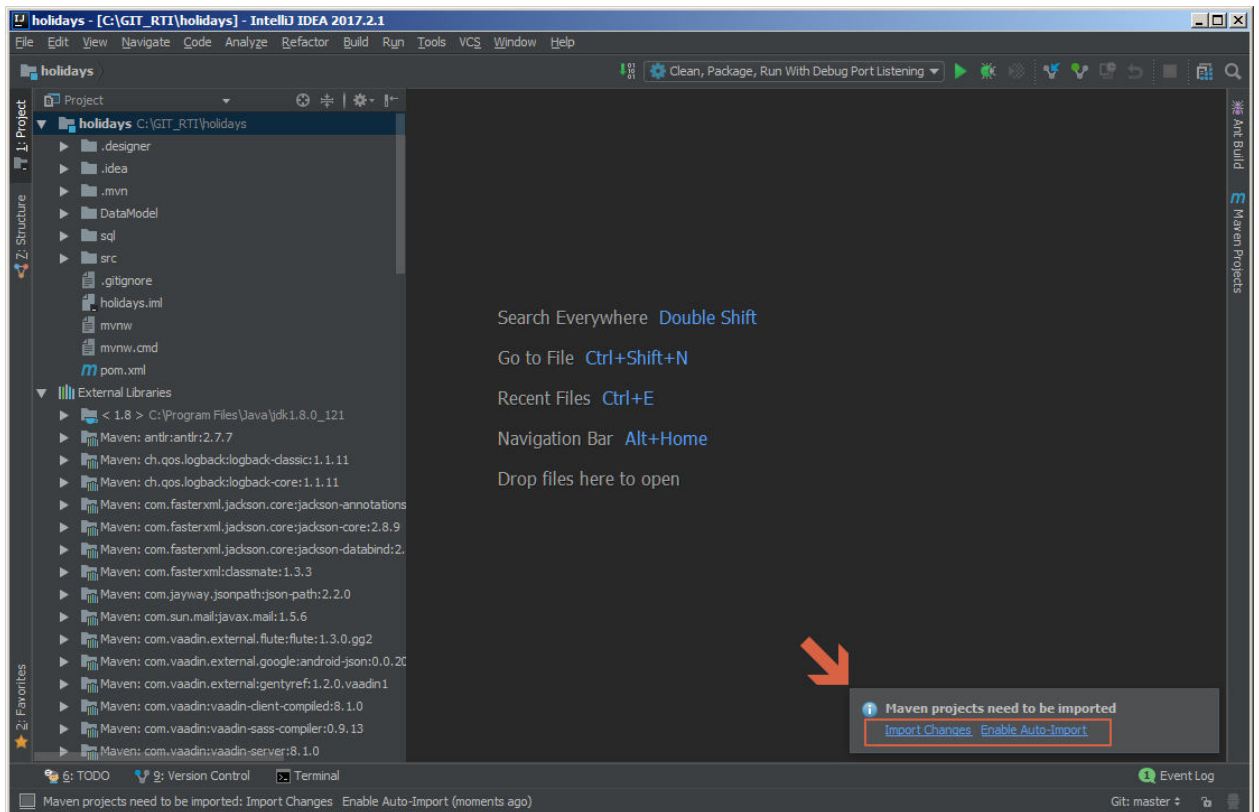


4. При условии того, что Вы подключены к корпоративной сети передачи данных (КСПД), начнётся загрузка файлов проекта в каталог локального репозитория. После завершения загрузки появится диалоговое окно:



5. Нажимаем **“Yes”** для того, чтобы сразу открыть проект в среде разработки.

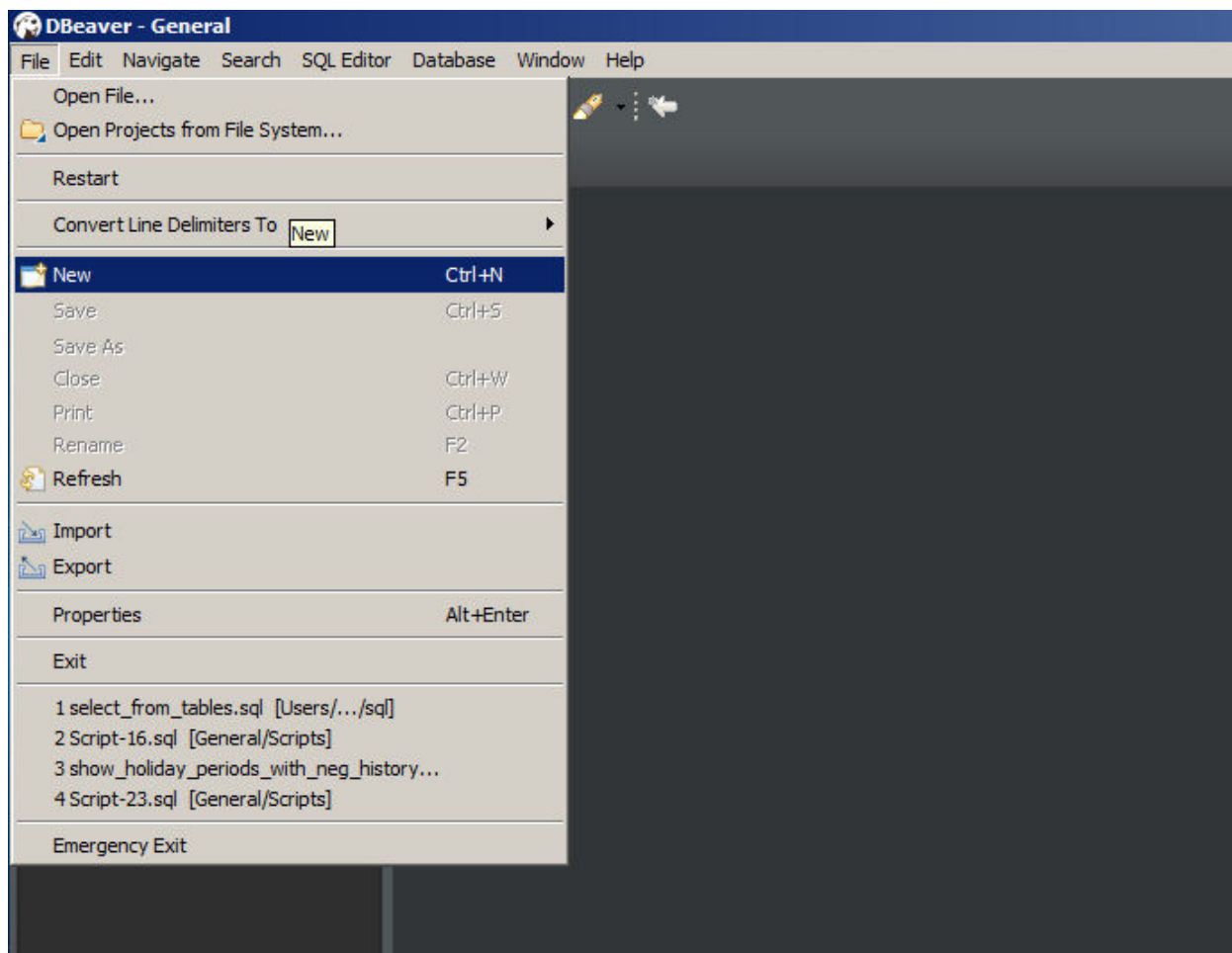
- После открытия проекта в IDEA может появиться сообщение в правом нижнем углу экрана – **“Maven projects need to be imported”** с опциями – **“Import Changes”**, **“Enable Auto-Import”**. Можно выбрать любую из них – первая делает разовый импорт изменений для репозитория Maven, вторая включит режим автоматического импорта изменений в случае их наличия.



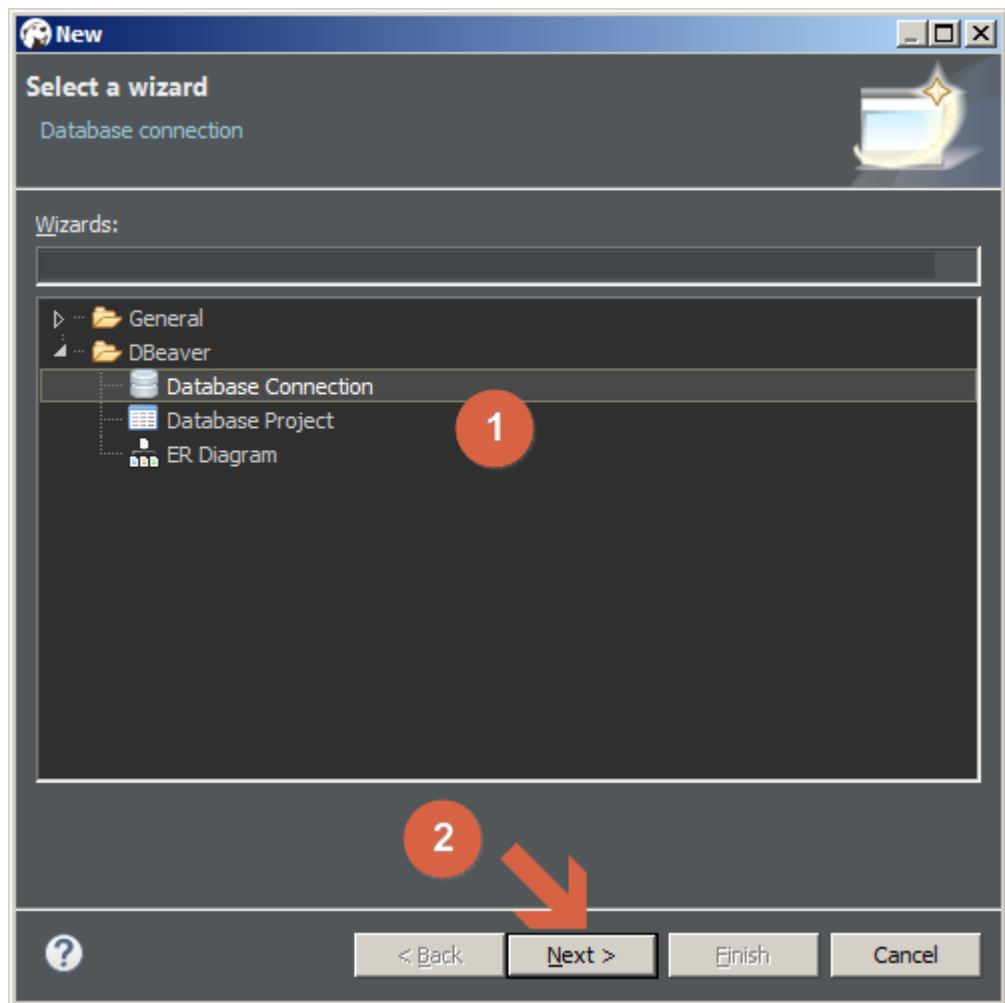
- Нажимаем **“Import Changes”**, дожидаемся, когда зависимости для Maven будут разрешены, а изменения импортированы.

Настройка подключения к локальной БД PostgreSQL в DBeaver

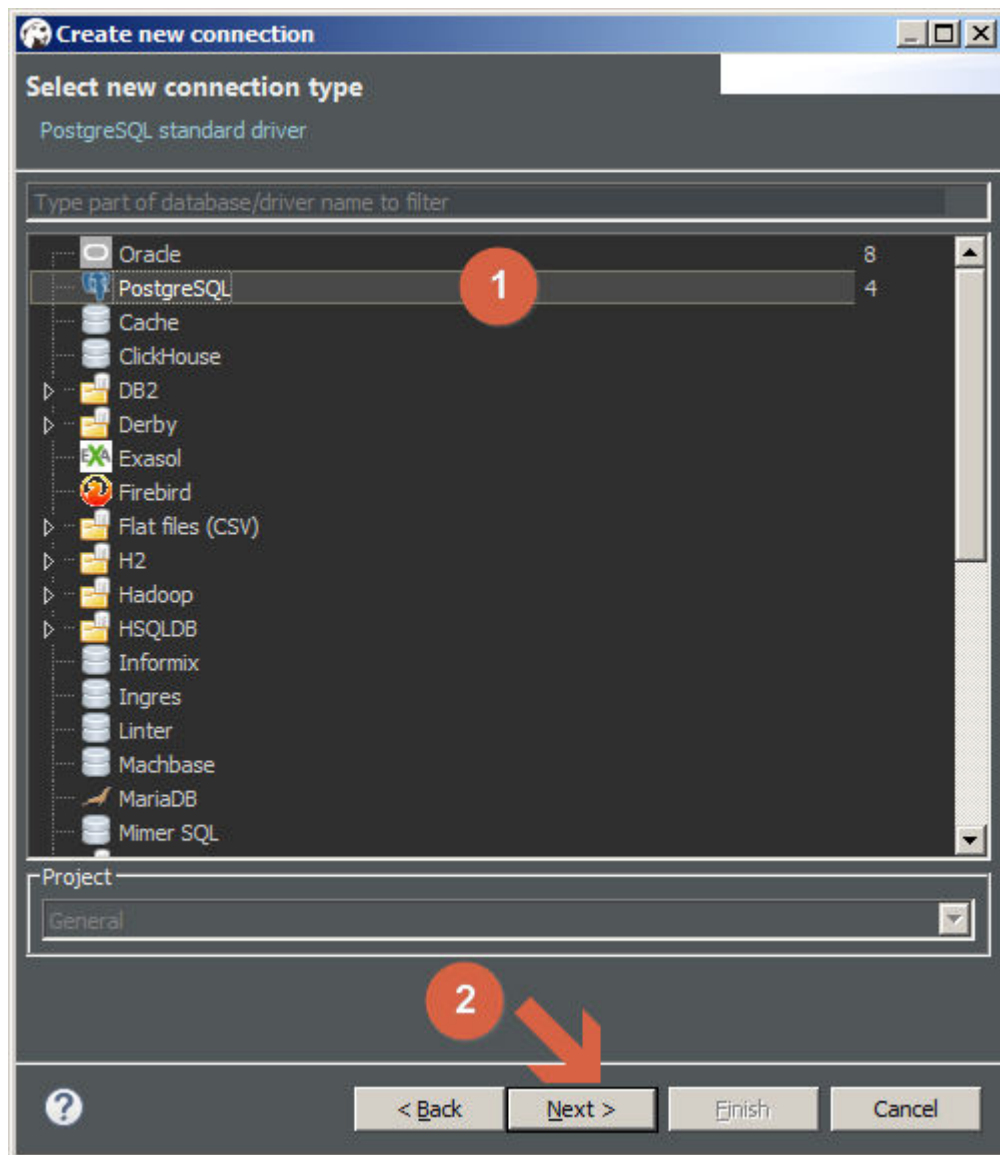
- Запускаем **DBeaver** и в главном окне приложения выбираем **“File”** -> **“New”**:



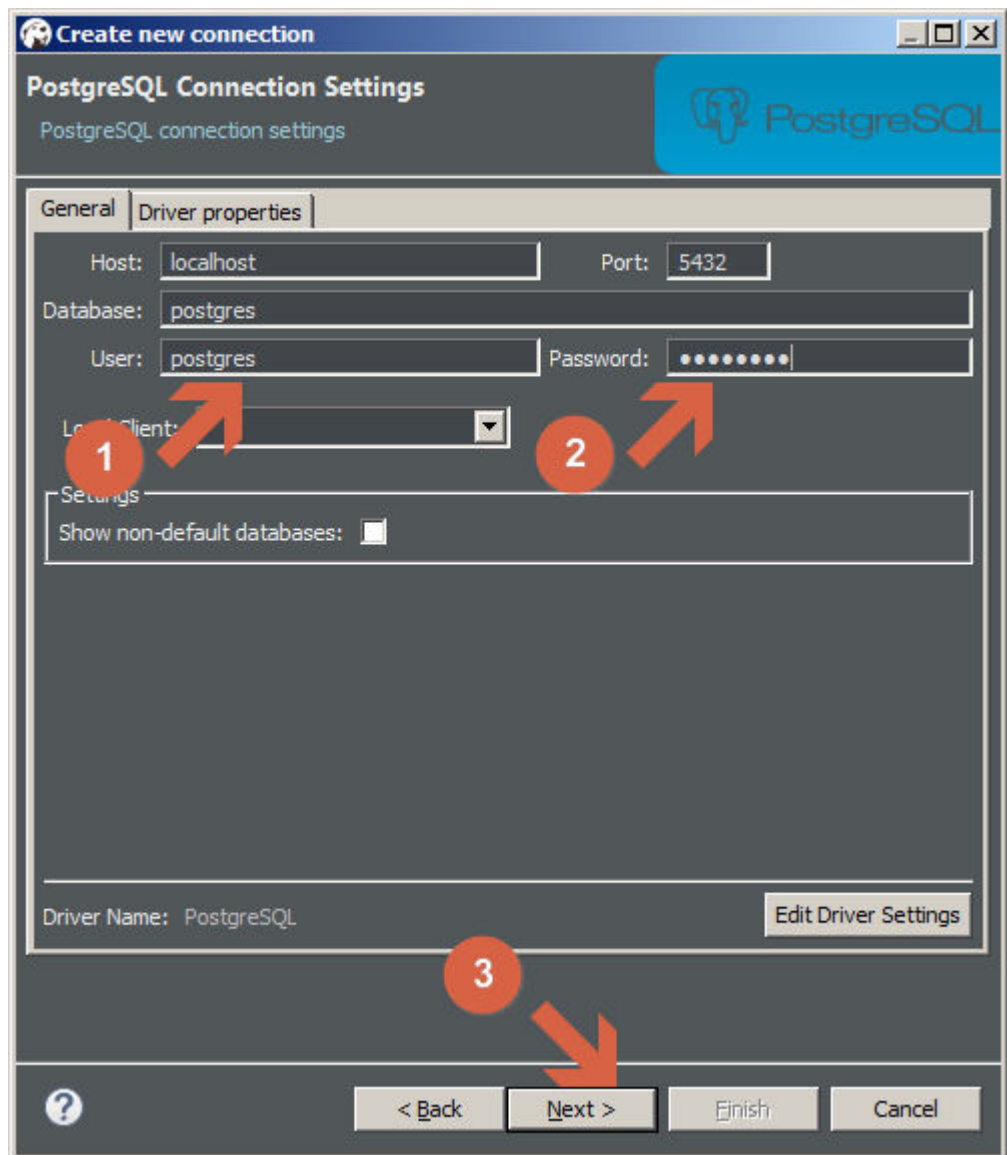
2. В открывшемся окне выбираем **"Database Connection"** и нажимаем **"Next >"**:



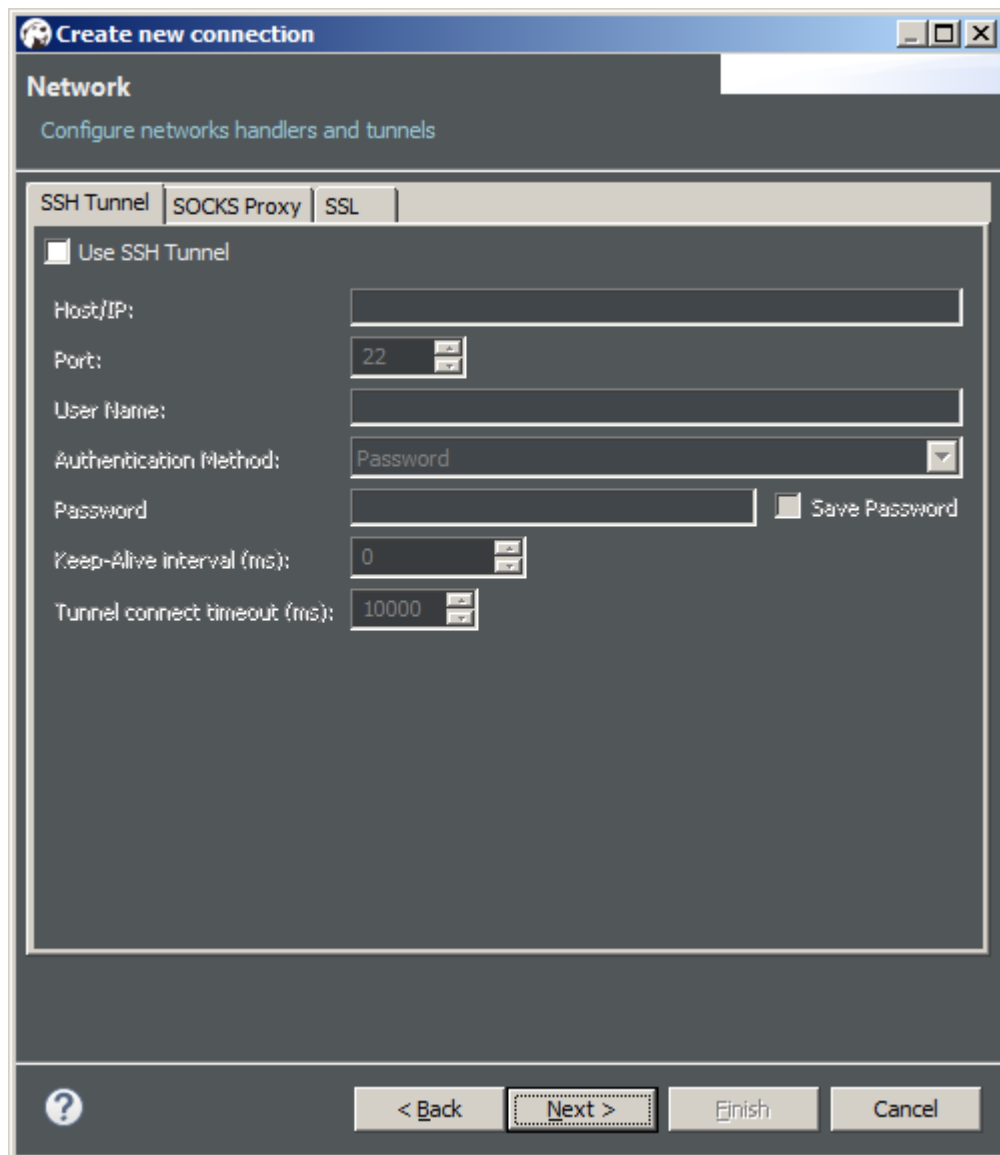
3. Выбираем **"PostgreSQL"** на шаге **"Select new connection type"** и нажимаем **"Next >":**



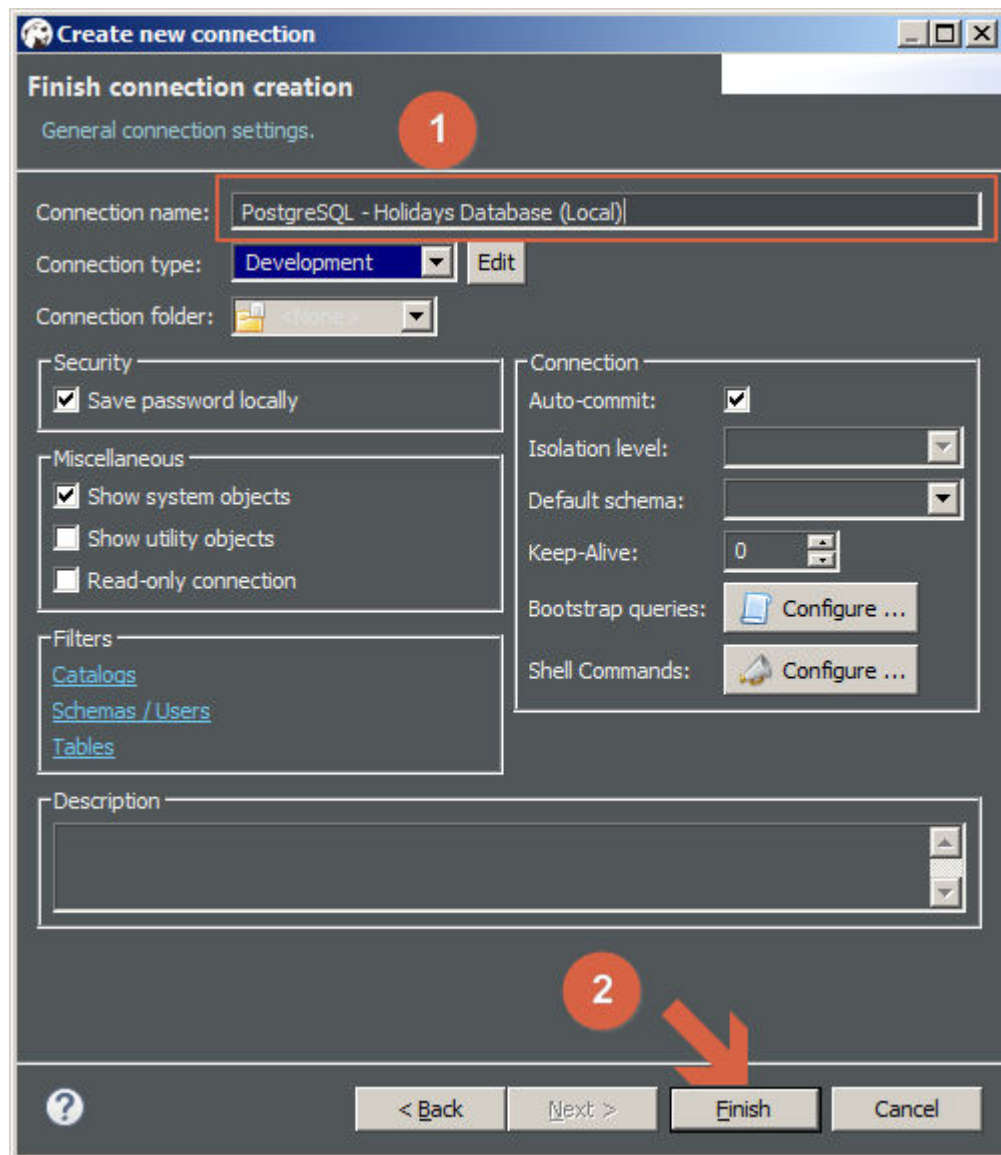
4. На шаге **"PostgreSQL Connection Settings"** вводим имя пользователя и пароль этого пользователя, которые Вы указали в процессе установки СУБД PostgreSQL:



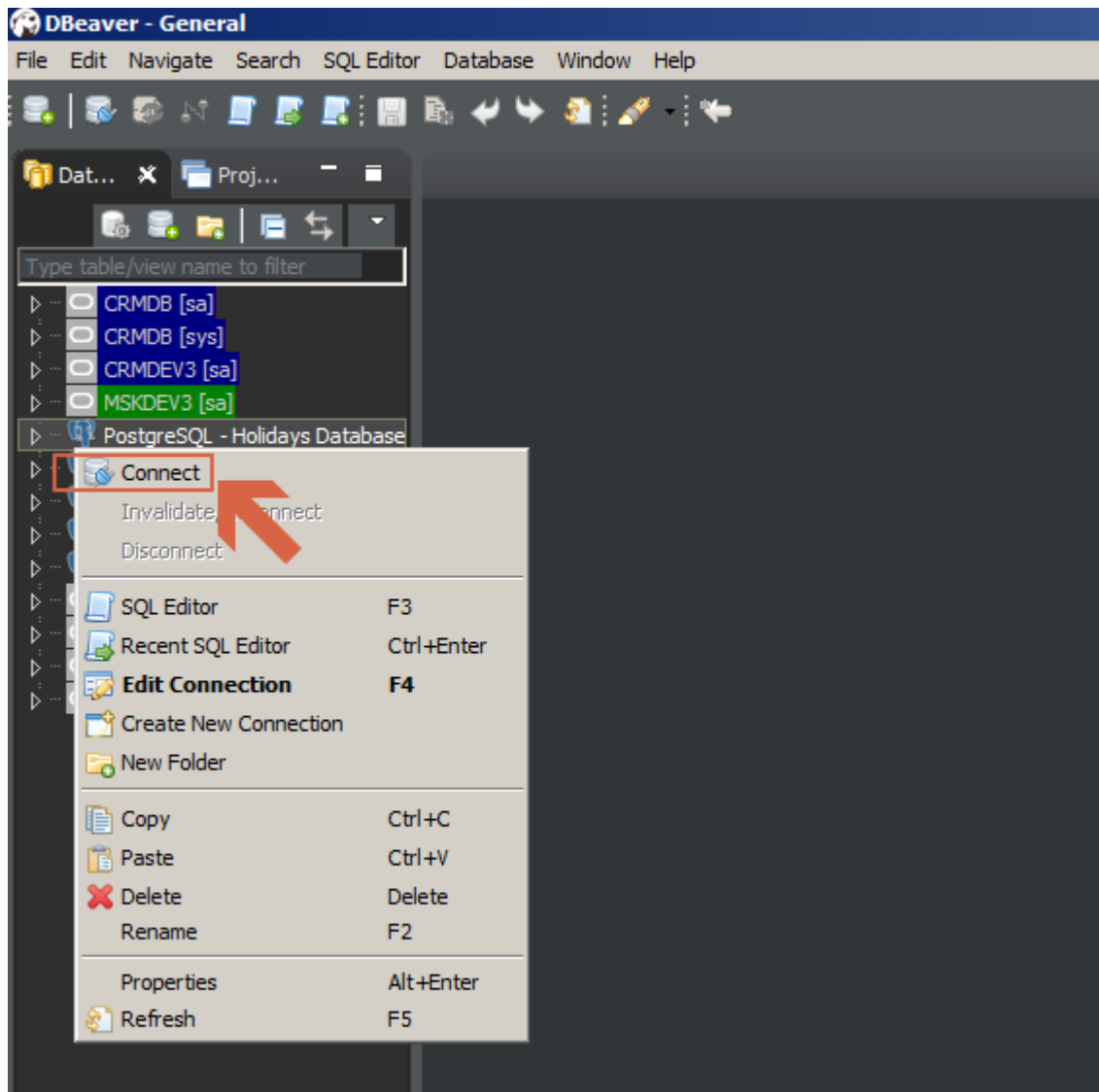
5. На шаге **“Network”** оставляем всё, как есть, и нажимаем **“Next >”**:



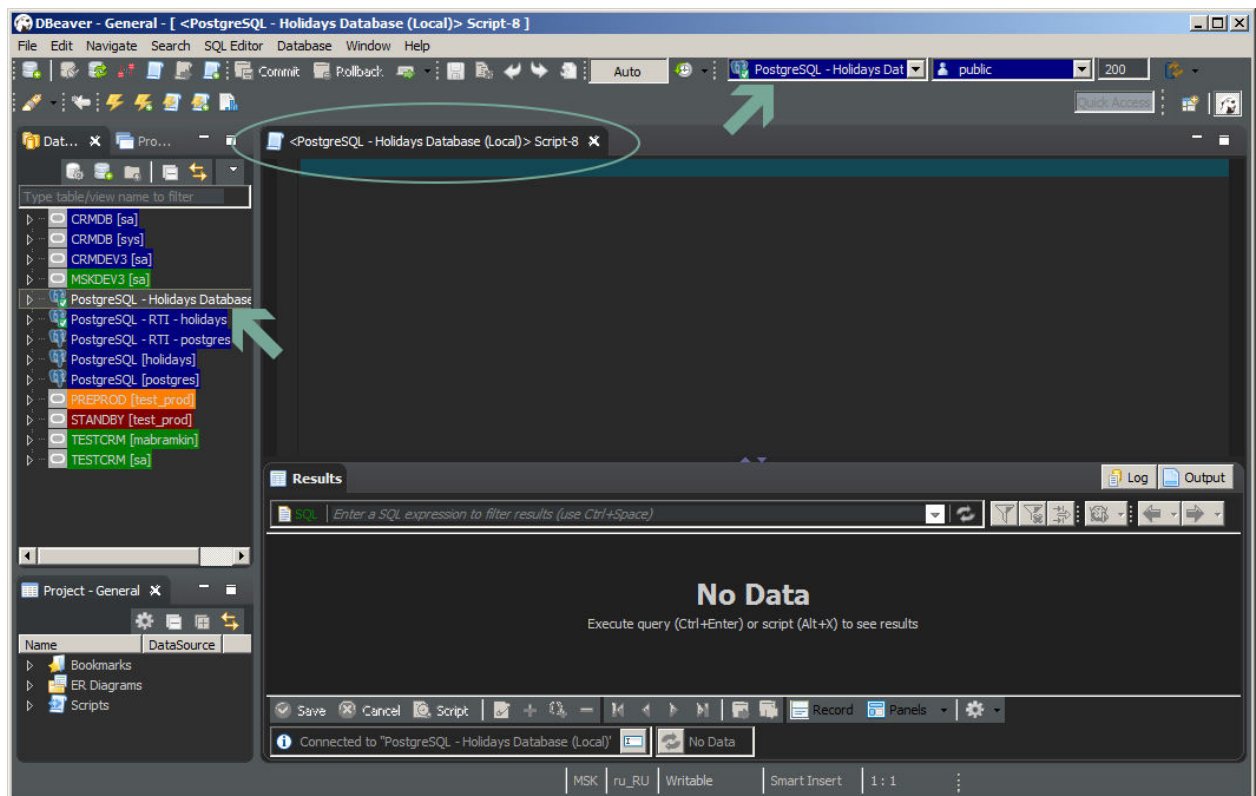
6. На шаге **"Finish connection creation"** в поле **"Connection name"** указываем имя подключения, например, **"PostgreSQL - Holidays Database (Local)"**, после чего нажимаем кнопку **"Finish"**:



7. После этого в дереве соединений слева, в окне **“Database Navigator”** появится созданное подключение с именем **“PostgreSQL – Holidays Database (Local)”**. Нажимаем на него правой кнопкой мыши и в контекстном меню выбираем **“Connect”**, либо делаем просто двойной щелчок по подключению:



8. После этого должно произойти успешное подключение к локальной БД PostgreSQL, о чём свидетельствуют зелёные иконки на имени подключения в окне навигатора слева и сверху. Также будет создан скрипт по умолчанию и откроется в новом вложенном окне:



9. На этом этапе подключение к локальной БД готово и настроено.

Создание новой роли 'holidays' и новой базы данных 'holidays'

После настройки подключения к локальной БД PostgreSQL и перед началом запуска основного приложения из среды разработки IntelliJ IDEA предварительно необходимо создать отдельную роль (пользователя) с именем **holidays**, а также создать для этого пользователя новую базу данных с таким же именем **holidays**. Для этого необходимо выполнить следующие шаги:

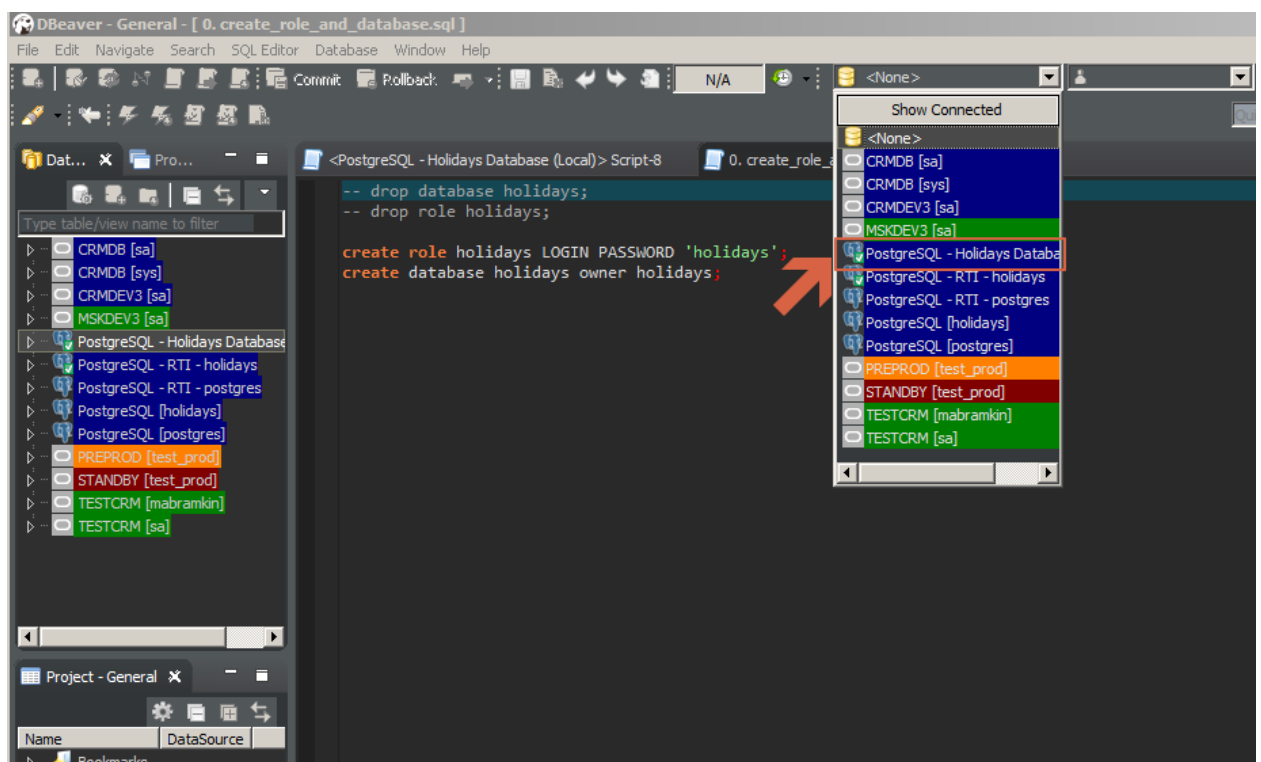
1. Подключиться к локальной БД (см. предыдущий раздел) из DBeaver
2. Перейти в каталог локального репозитория и в корневом каталоге с файлами приложения найти каталог **sql** и открыть его:

Имя ^	Дата изменения	Тип	Размер
.designer	11.12.2017 19:16	Папка с файлами	
.git	11.12.2017 20:54	Папка с файлами	
.idea	11.12.2017 21:14	Папка с файлами	
.mvn	11.12.2017 19:16	Папка с файлами	
DataModel	11.12.2017 19:16	Папка с файлами	
sql	11.12.2017 19:16	Папка с файлами	
src	11.12.2017 19:16	Папка с файлами	
target	11.12.2017 19:34	Папка с файлами	
.gitignore	11.12.2017 19:16	Текстовый документ	1 КБ
holidays.iml	11.12.2017 19:16	Файл "IML"	11 КБ
mvnw	11.12.2017 19:16	Файл	7 КБ
mvnw.cmd	11.12.2017 19:16	Сценарий Windows	6 КБ
pom.xml	11.12.2017 19:28	Документ XML	6 КБ

3. Найти скрипт с именем **0. create_role_and_database.sql**:

Имя ^	Дата изменения	Тип	Размер
utility_queries	11.12.2017 19:16	Папка с файлами	
!README.txt	11.12.2017 19:16	Текстовый документ	1 КБ
0. create_role_and_database.sql	11.12.2017 19:16	Файл "SQL"	1 КБ
1. insert_departments.sql	11.12.2017 19:16	Файл "SQL"	1 КБ
2. insert_teams.sql	11.12.2017 19:16	Файл "SQL"	2 КБ
3. insert_project_roles.sql	11.12.2017 19:16	Файл "SQL"	2 КБ
4. insert_employees.sql	11.12.2017 19:16	Файл "SQL"	31 КБ
5. insert_neg_statuses.sql	11.12.2017 19:16	Файл "SQL"	3 КБ

4. Перетащить этот SQL-скрипт в окно DBeaver с помощью drag & drop. После этого будет создана отдельная вкладка с содержимым скрипта, но данная вкладка не привязана к активному подключению. Для привязки к подключению с локальной БД PostgreSQL в верхнем выпадающем списке с иконкой БД выбираем настроенное ранее подключение с именем **"PostgreSQL - Holidays Database (Local)"**:



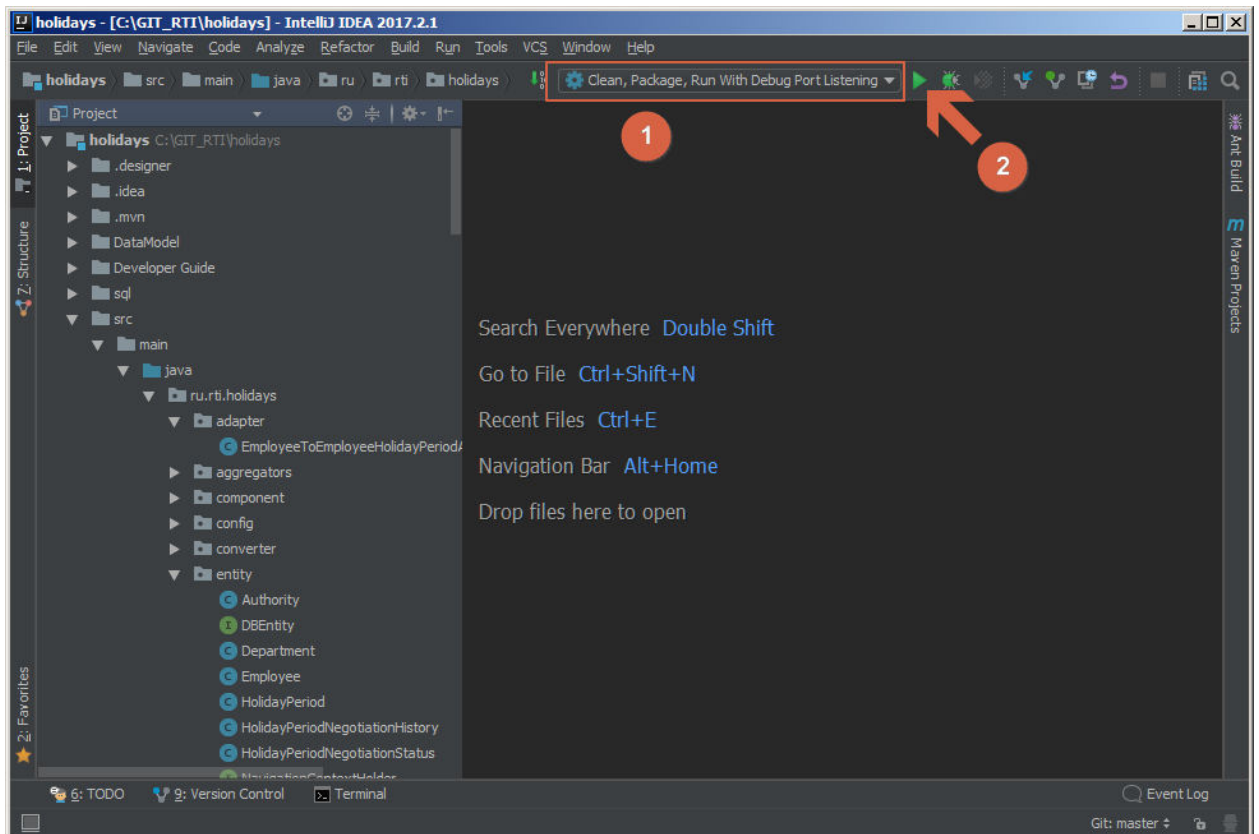
5. После того, как окно SQL-скрипта будет привязано к подключению в окне скрипта нажимаем комбинацию **Alt + X** для выполнения скрипта полностью.

В результате будет создан отдельный пользователь с именем **holidays** и для него создана новая пустая база данных с таким же именем (**holidays**).

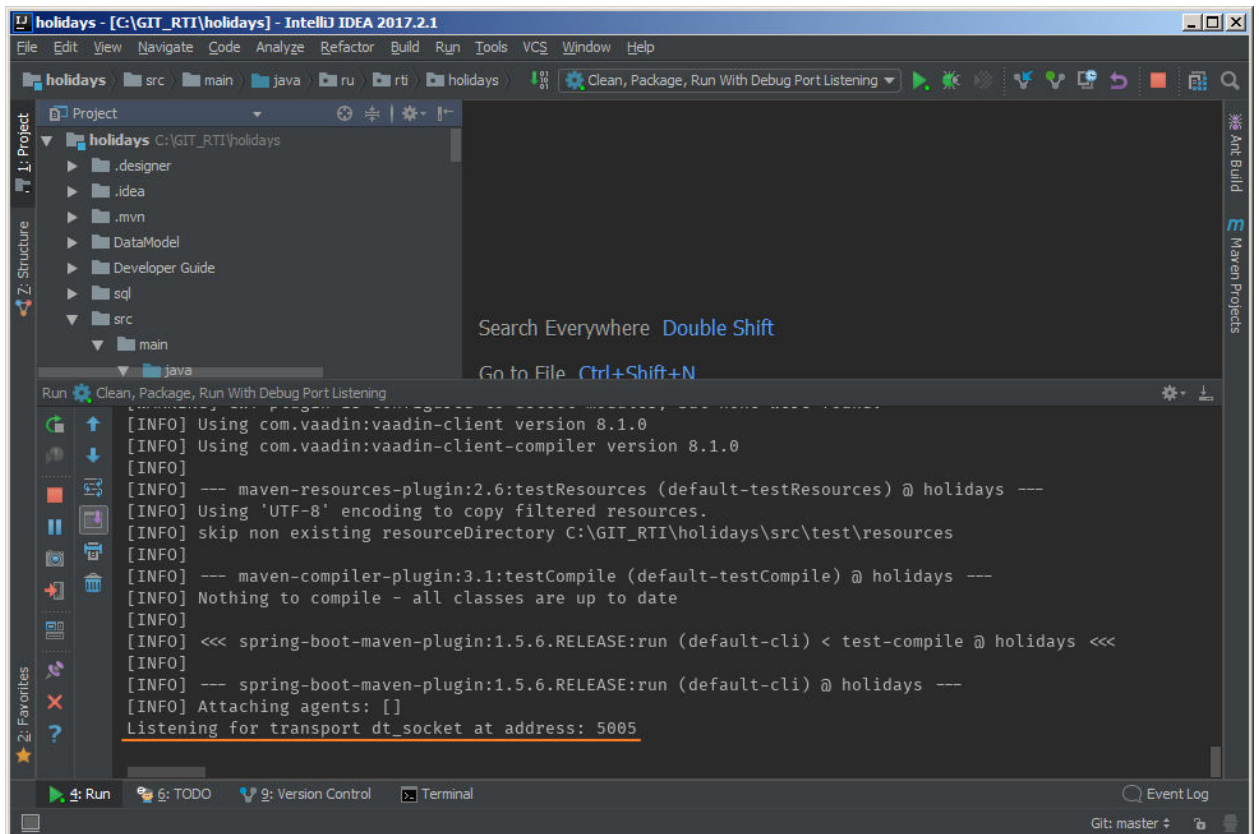
Запуск приложения из IntelliJ IDEA в режиме отладки кода (Debug)

Для запуска приложения в режиме отладки необходимо:

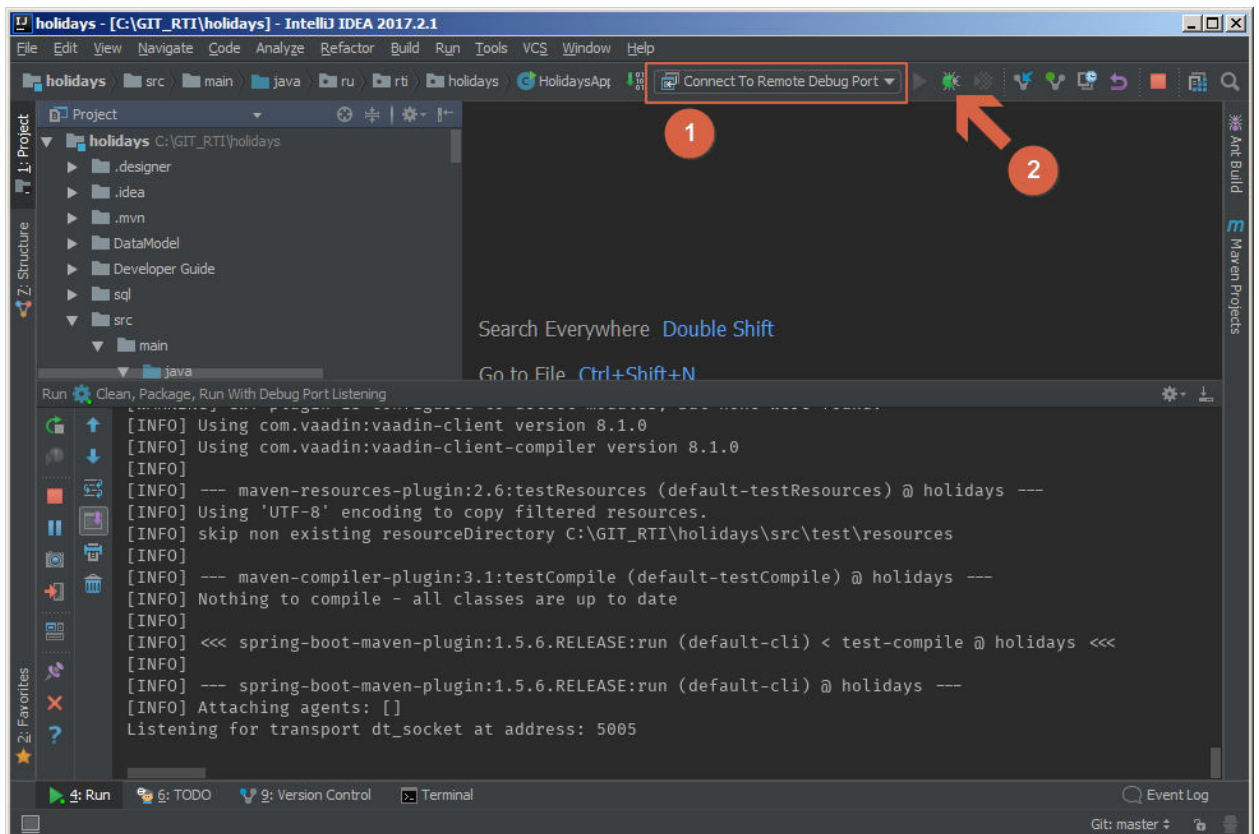
1. В выпадающем списке конфигураций, выбрать конфигурацию Maven с именем **“Clean, Package, Run With Debug Port Listening”**:, после чего нажать на иконку **Run** (или нажать **Shift + F10**):



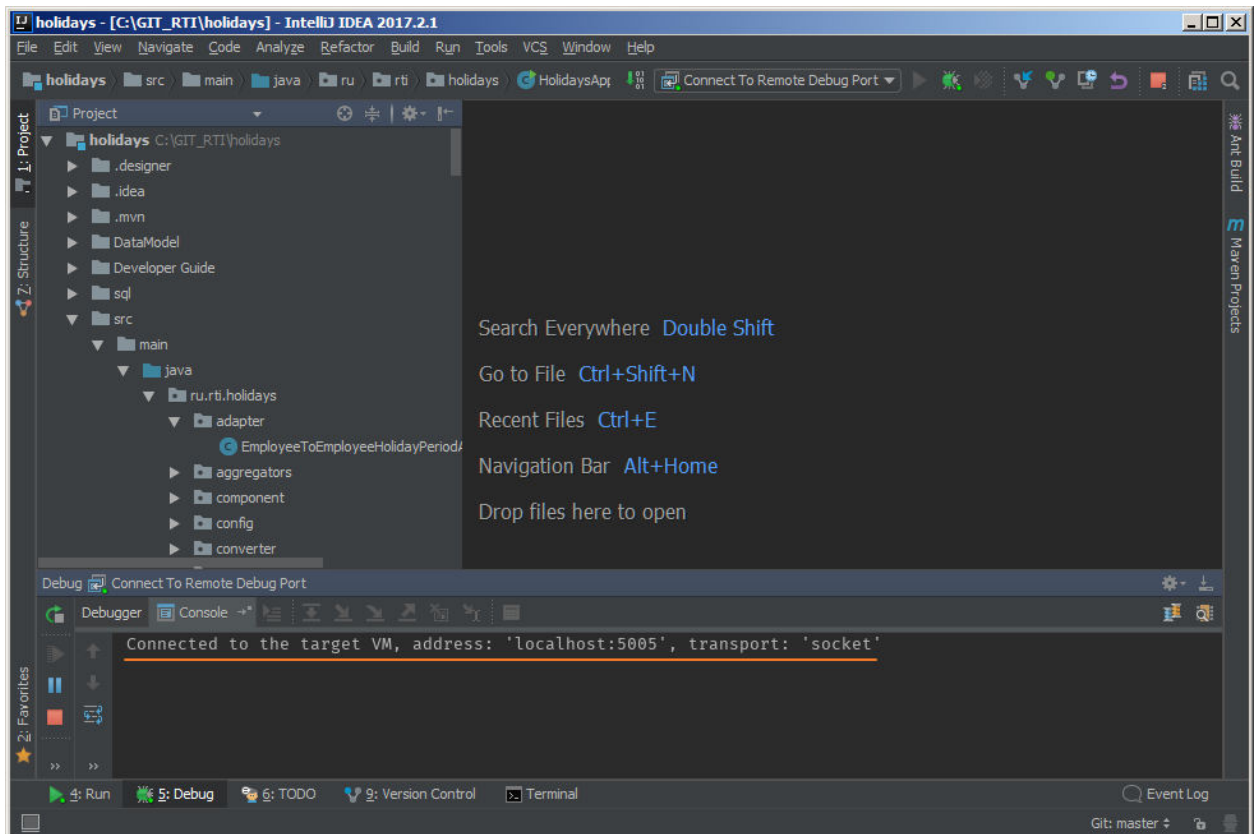
2. После этого приложение начнёт запускаться, а в нижнем окне **“Run”** консоли начнут появляться соответствующие сообщения инициализации приложения. Ожидать необходимо до момента появления сообщения **“Listening for transport dt_socket at address: 5005”**, как показано на рисунке ниже:



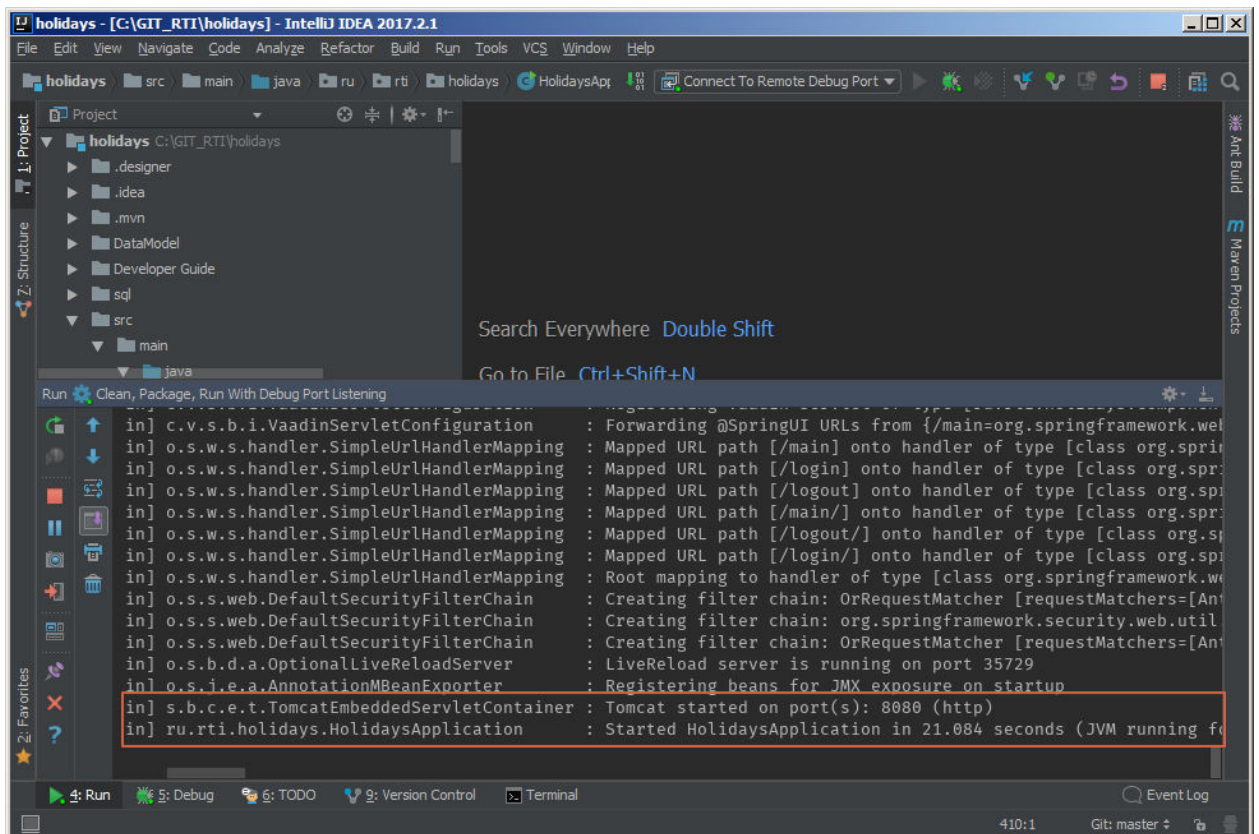
3. После этого необходимо в выпадающем списке конфигураций выбрать другую конфигурацию Maven с именем **“Connect To Remote Debug Port”**, после чего нажать на значок «жука» (Debug) или нажать комбинацию **Shift + F9**:



4. В результате этого действия в окне **“Console”** нижней вкладки **“Debug”** появится сообщение **“Connected to the target VM, address: ‘localhost:5005’, transport: ‘socket’”**:

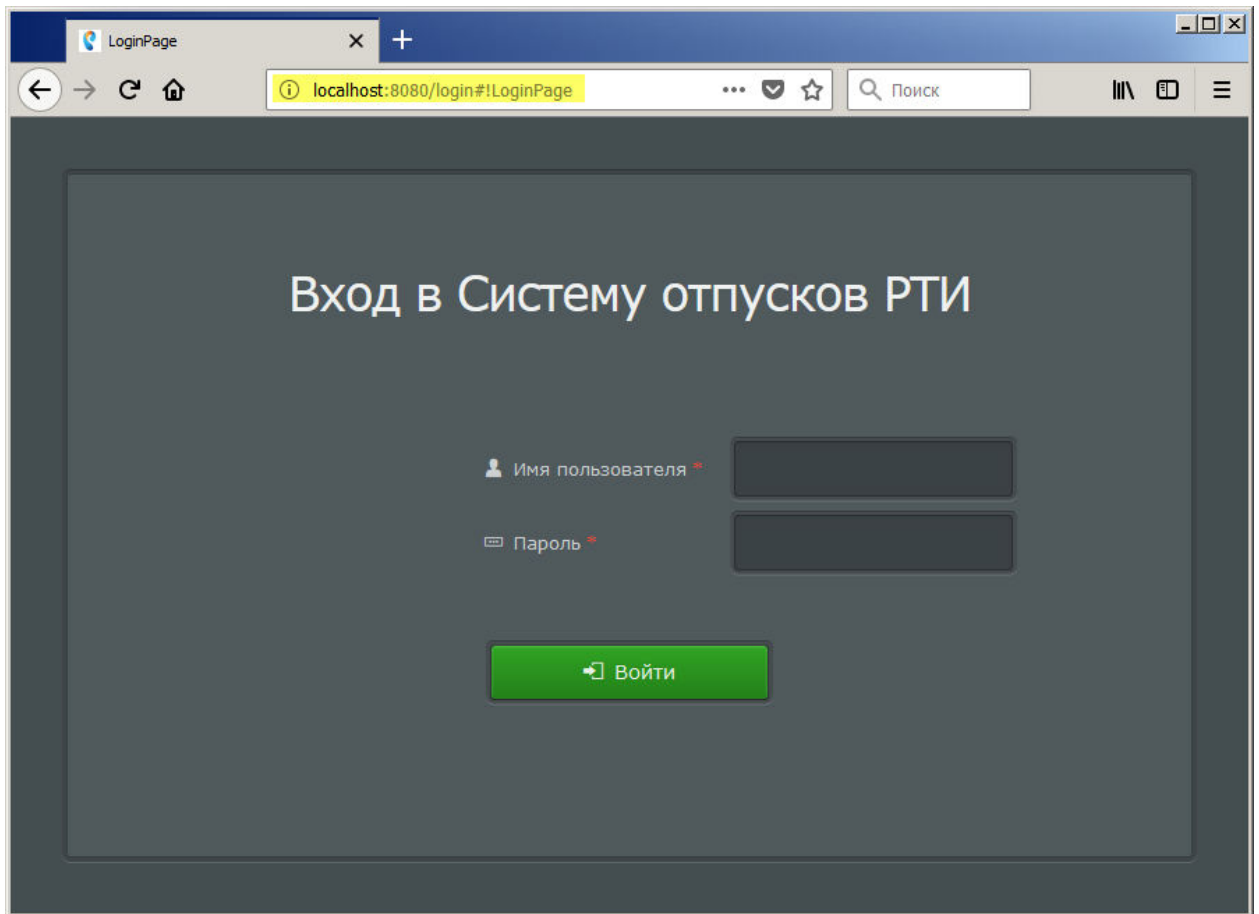


5. Переключитесь в нижней панели на вкладку "Run". Необходимо дождаться инициализации приложения до момента появления сообщения **"Started HolidaysApplication in NN.MMM seconds (JVM running for XXX.YYY)"**:



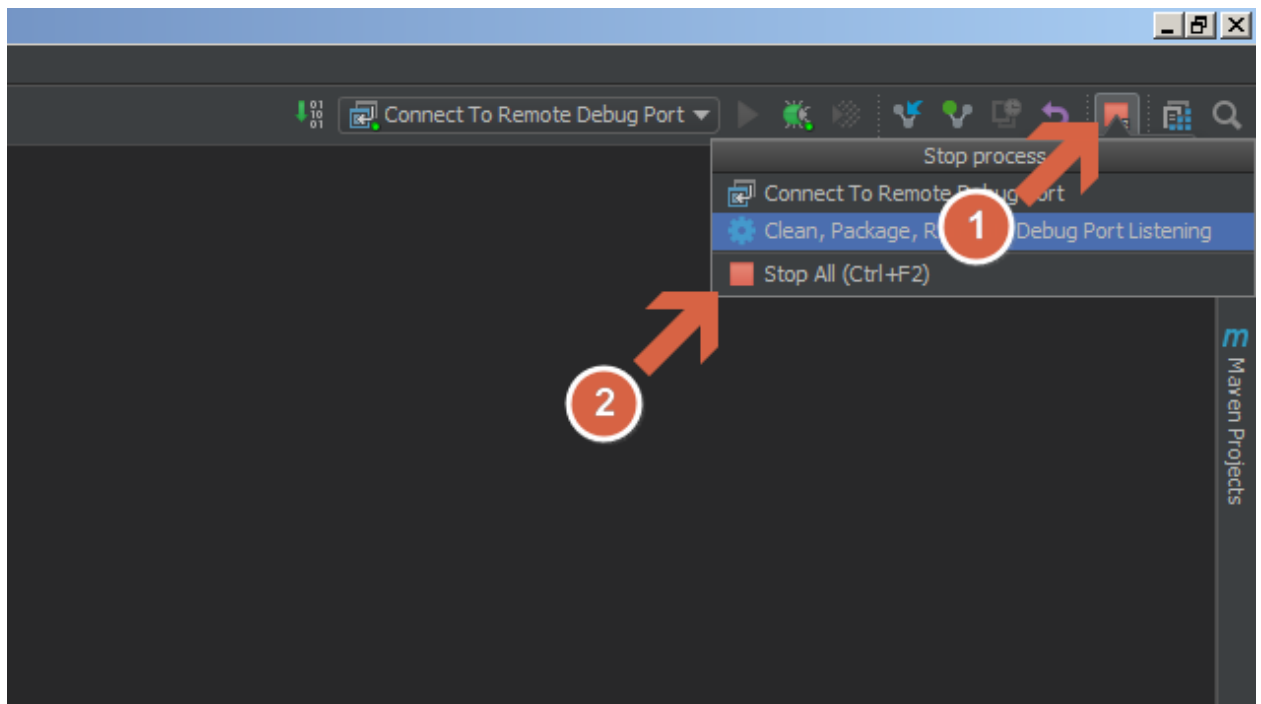
6. После этого режим отладки полностью подключен, приложение запущено. Откройте Ваш любимый браузер и наберите в адресной строке URL развёрнутого локально приложения: <http://localhost:8080>. После загрузки

Web-страницы Вы увидите интерфейс системы бронирования отпусков РТИ и система автоматически перенаправит запрос на адрес <http://localhost:8080/login#!LoginPage>:

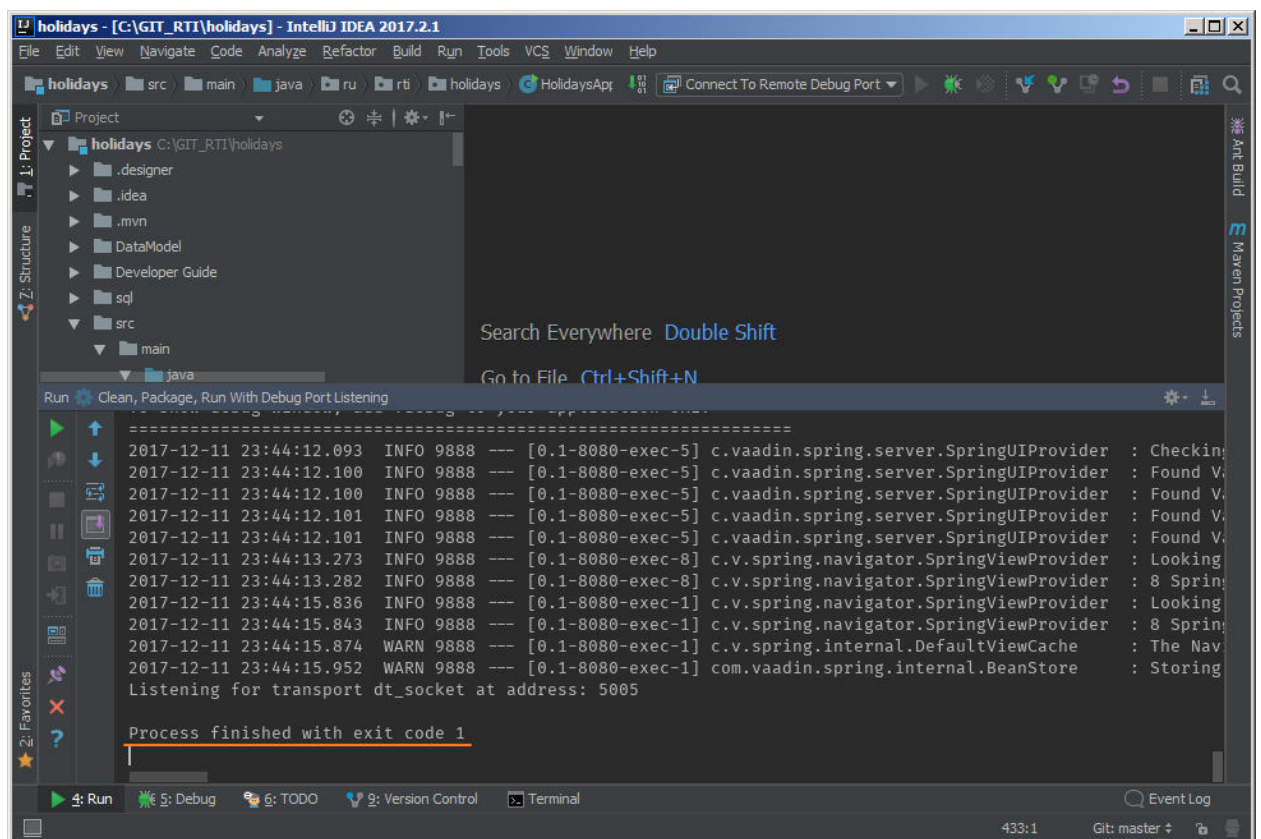


Прекращение/останов отладки приложения

Для прекращения отладки приложения **дважды** нажмите комбинацию **Ctrl + F2** или в верхней правой панели IDEA выберите иконку **Stop** и в выпадающем меню пункт **"Stop All"**:



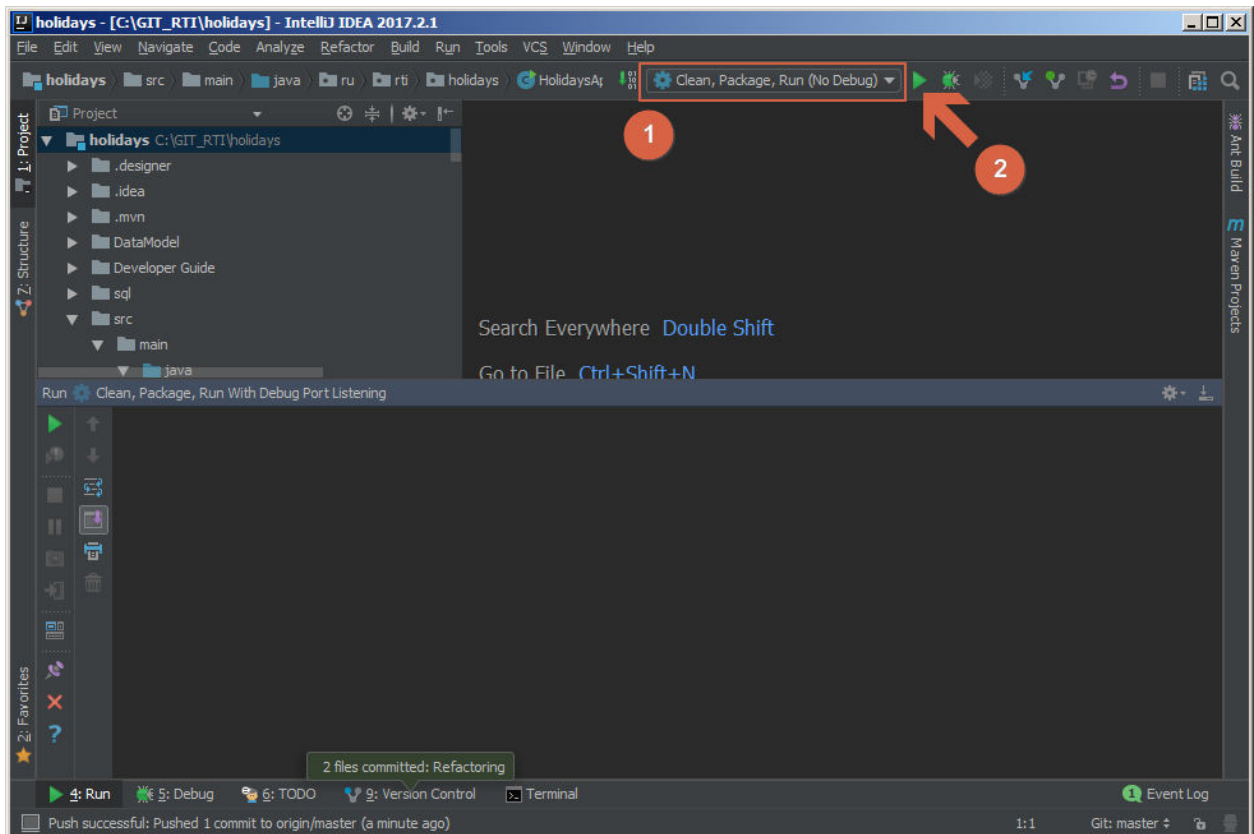
После этого в окне **“Run”** появится сообщение **“Process finished with exit code 1”**, что говорит о том, что отладка прекращена, приложение остановлено:



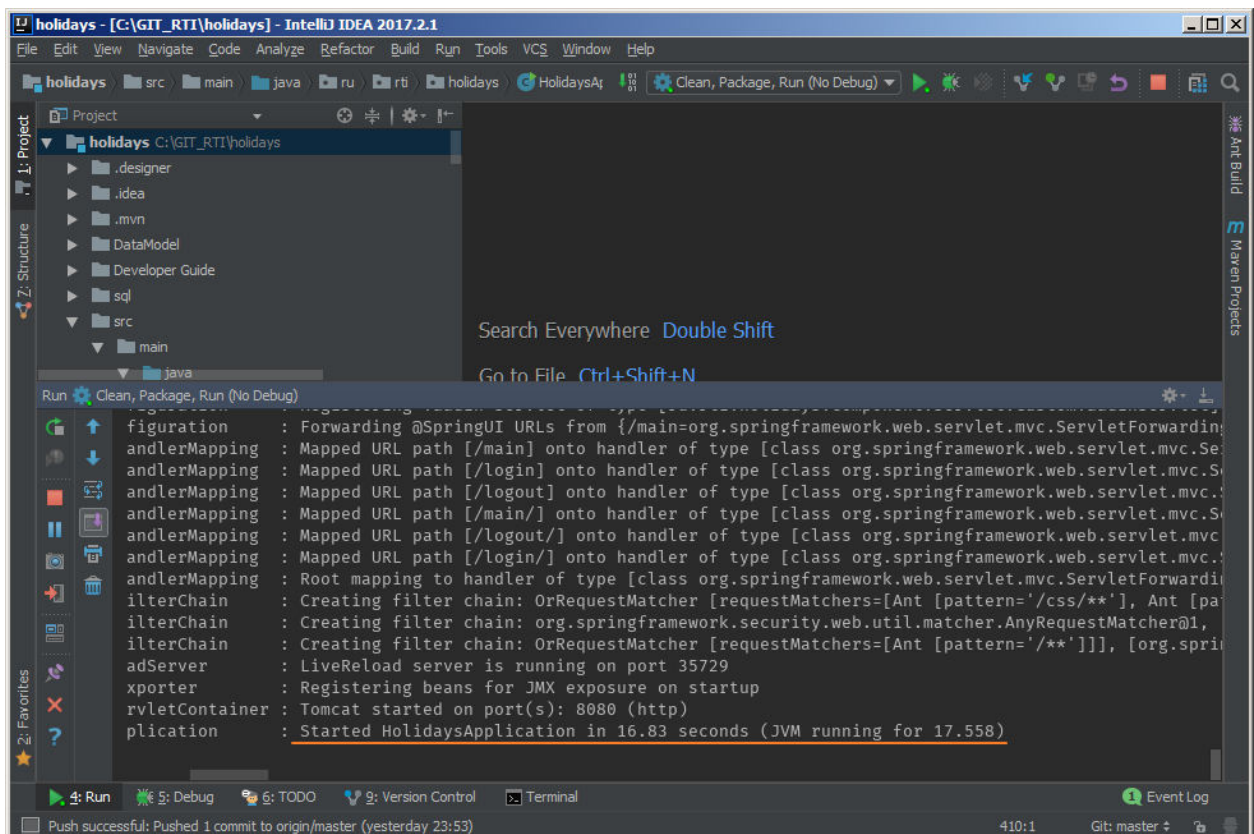
Запуск приложения из IntelliJ IDEA без режима отладки (Run)

Для запуска приложения без возможностей отладки необходимо:

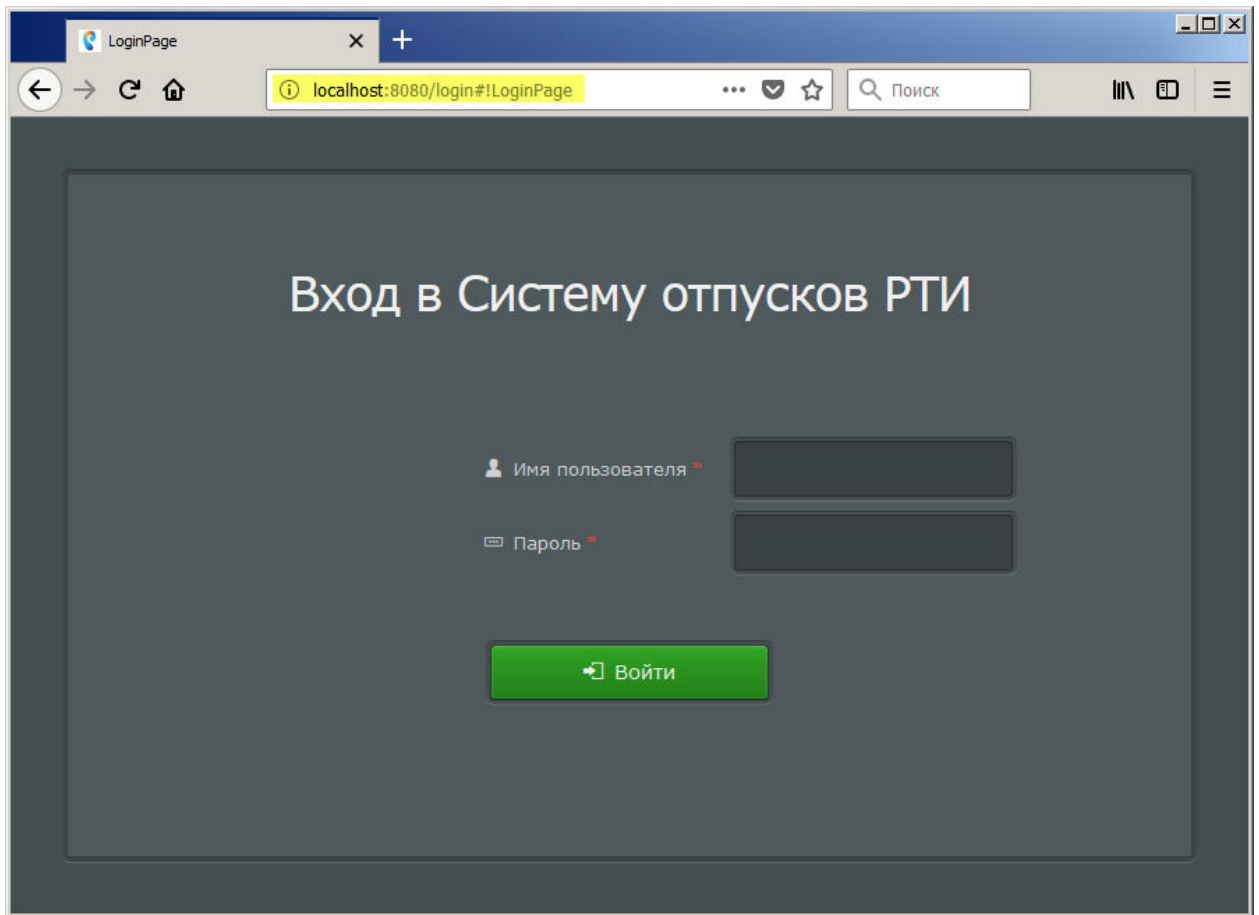
1. В выпадающем списке конфигураций, выбрать конфигурацию Maven с именем **"Clean, Package, Run (No Debug)"**:, после чего нажать на иконку **Run** (или нажать **Shift + F10**):



2. В окне **"Run"** нижней панели станут появляться сообщения. Необходимо дождаться надписи **"Started HolidaysApplication in NN.MMM seconds (JVM running for XXX.YYY)"**:

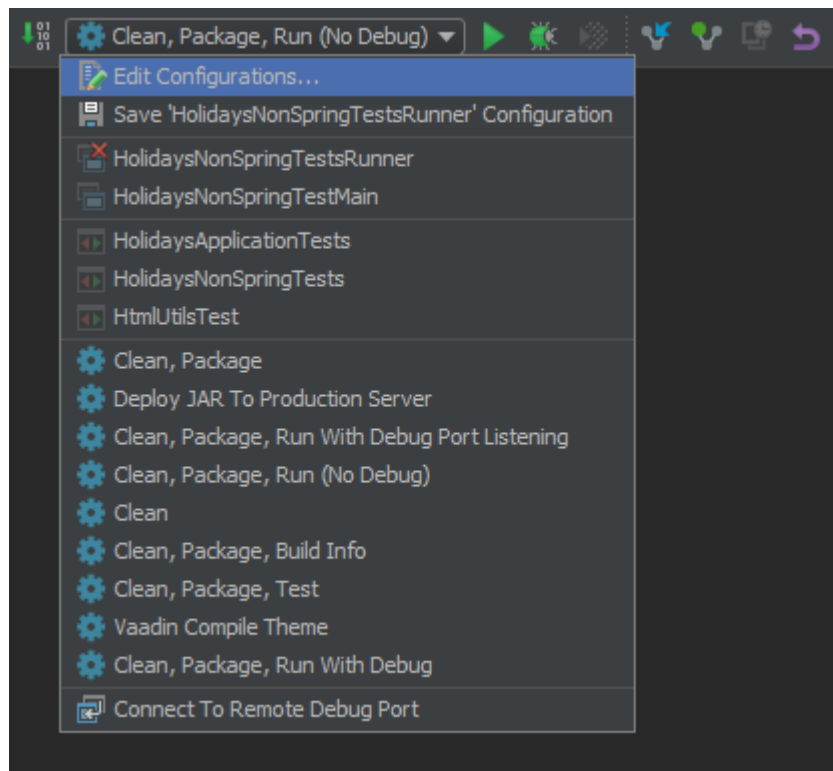


3. После этого приложение запущено. Откройте Ваш любимый браузер и наберите в адресной строке URL развёрнутого локально приложения: <http://localhost:8080>. После загрузки Web-страницы Вы увидите интерфейс системы бронирования отпусков РТИ и система автоматически перенаправит запрос на адрес <http://localhost:8080/login#!LoginPage>:



Описание доступных конфигураций запуска и отладки

В выпадающем списке верхней панели среды разработки IDEA можно увидеть ряд доступных конфигураций запуска/отладки для управления приложением:



Ниже приводится описание доступных конфигураций:

Название конфигурации	Тип конфигурации	Описание
Clean, Package	Maven	Производит очистку и упаковку приложения в *.JAR файл
Deploy JAR To Production Server	Maven	Производит копирование *.JAR файла на продуктивный сервер
Clean, Package, Run With Debug Port Listening	Maven	Производит очистку, упаковку приложения в *.JAR файл и запускает приложение в режиме отладки (Debug) на прослушиваемом порту 5005
Clean, Package, Run (No Debug)		Производит очистку, упаковку приложения в *.JAR файл и запускает приложение без режима отладки
Clean	Maven	Производит очистку приложения
Clean, Package, Build-Info	Maven	Производит очистку и упаковку приложения, а также выводит информацию о сборке
Clean, Package, Test	Maven	Производит очистку и упаковку приложения, а также запускает Junit-тесты
Vaadin Compile Theme		Производит компиляцию темы из SCSS-файлов стилей в CSS файлы темы
Clean, Package, Run With Debug	Maven	Производит очистку и упаковку приложения, а также

Connect To Remote Debug Port	Remote	Производит подключение к отладчику в случае, если он развёрнут на порту 5005 и позволяет отлаживать приложение
-------------------------------------	--------	--