**ИСР-1.1**

**Функции IntelliJ IDEA Community Edition**

**AI-функции**

1. **Основные возможности AI Assistant (доступные в Community Edition)**

* **Бесплатные базовые функции:** JetBrains предоставляет бесплатный доступ к AI-автодополнению кода (code completion) и мощным локальным рабочим процессам. Некоторые облачные функции могут быть ограничены.
* **Локальные модели:** Можно использовать локальные модели для AI-автодополнения, что позволяет работать офлайн, например, в полете или в закрытой корпоративной среде. Это включает предложения блоков кода для Java в автономном режиме и возможность выбора предпочтительной локальной модели.
* **Мультиязычное автодополнение:** AI Assistant теперь предоставляет AI-автодополнение для SQL, YAML, JSON и Markdown.

1. **Уровни AI-помощи**

AI-функции в IntelliJ IDEA реализованы на разных уровнях

* **Уровень 0: Минимальная помощь**: Локальные inline-автодополнения с использованием легковесных моделей. Помогает с повторяющимися задачами, когда разработчик точно знает, что делать. Быстрые и контекстно-aware, но ограниченные по охвату.
* **Уровень 1: Многострочные автодополнения**: Многострочные облачные автодополнения на основе комментариев или частичного кода. Полезны для решения известных проблем или генерации кода на незнакомых языках. Например, могут реализовать служебные функции просто из описательного комментария.

1. **Расширенные возможности (требующие подписки или недоступные в Community Edition)**

Некоторые функции могут потребовать подписку JetBrains AI Assistant или IntelliJ IDEA Ultimate:

* **Junie:** AI-агент для автоматизации сложных рабочих процессов (например, сканирование проекта, генерация планов разработки, выполнение задач across multiple files) не входит в базовую поставку Community Edition.
* **Поддержка MCP (Model Context Protocol):** Позволяет IntelliJ IDEA выступать в качестве MCP-сервера для сторонних AI-агентов, предоставляя более 30 встроенных инструментов. Это требует соответствующей конфигурации и, вероятно, подписки.
* **Расширенные контекстные возможности:** Например, прикрепление конкретных таблиц или представлений к AI-чату для точной помощи, основанной на актуальном состоянии структур данных, или прикрепление изображений в AI-чате для обмена ошибками и диаграммами. Эти функции связаны с облачными моделями (Anthropic, OpenAI) и могут требовать подписки.

1. **Работа с чатом и контекстом**

* **Добавление контекста:** Можно добавлять папки в контекст чата, чтобы улучшить многфайловый контекст для AI-разговоров, используя символы @ или #.
* **Project Rules:** Файл Project Rules позволяет определить проекто-специфичные инструкции (например, regarding coding style или framework constraints) для руководства AI-ответов. Это помогает кастомизировать AI-настройку под конкретные нужды проекта.

1. **Важные ограничения в Community Edition**

* **Отсутствие Spring Debugger и Spring Modulith**: Эти мощные инструменты для отладки и работы с Spring-приложениями являются частью IntelliJ IDEA Ultimate и не доступны в Community Edition.
* **Базовый функционал AI Assistant:** Полный набор функций AI Assistant, включая расширенный чат, рефакторинг и агентные возможности Junie, требует активации подписки JetBrains AI Assistant.

**Тестирование**

1. **Юнит-тесты (Unit Tests)**

**Цель:** Проверить корректность работы отдельного, изолированного модуля (функции, класса, метода) в коде.

**Ключевые характеристики:**

* **Изоляция:** Тестируемый модуль изолирован от всех внешних зависимостей (база данных, файловая система, сетевые запросы, другие классы).
* **Скорость:** Выполняются очень быстро (миллисекунды на тест).
* **Фокус:** Проверяют логику одного конкретного модуля.
* **Детерминированность**: Результат теста (успех/провал) должен всегда быть одинаковым для одних и тех же входных данных.

**Как добиться изоляции? Использование Test Doubles ( «двойников» )**

* **Mock:** Объект-заглушка, который регистрирует вызовы своих методов. Тест проверяет, был ли вызван нужный метод с правильными аргументами.
  + ***Пример:*** Проверить, что после вызова метода userService.register() был вызван метод emailService.sendWelcomeEmail().
* **Stub:** Объект-заглушка, который возвращает заранее заданные, фиктивные данные в ответ на вызовы методов.
  + ***Пример:*** Заглушка для DatabaseRepository всегда возвращает готовый объект User, чтобы протестировать UserService без реальной БД.
* **Fake:** Упрощенная, но рабочая реализация зависимости. Обычно не подходит для production, но работает для тестов.
  + ***Пример:*** FakeDatabaseRepository, который хранит данные в памяти (в HashMap), а не в реальной БД.

**Поддержка плагинов и расширений**

IntelliJ IDEA отличается мощной экосистемой плагинов и расширений, которые значительно расширяют её функциональность. Вот основные аспекты поддержки плагинов в этой среде разработки:

1. **Установка и управление плагинами**

Плагины в IntelliJ IDEA устанавливаются через встроенный менеджер плагинов, доступный в разделе Preferences/Settings → Plugins или на вкладке Plugins начального экрана. Это позволяет легко находить, устанавливать и обновлять расширения непосредственно из IDE.

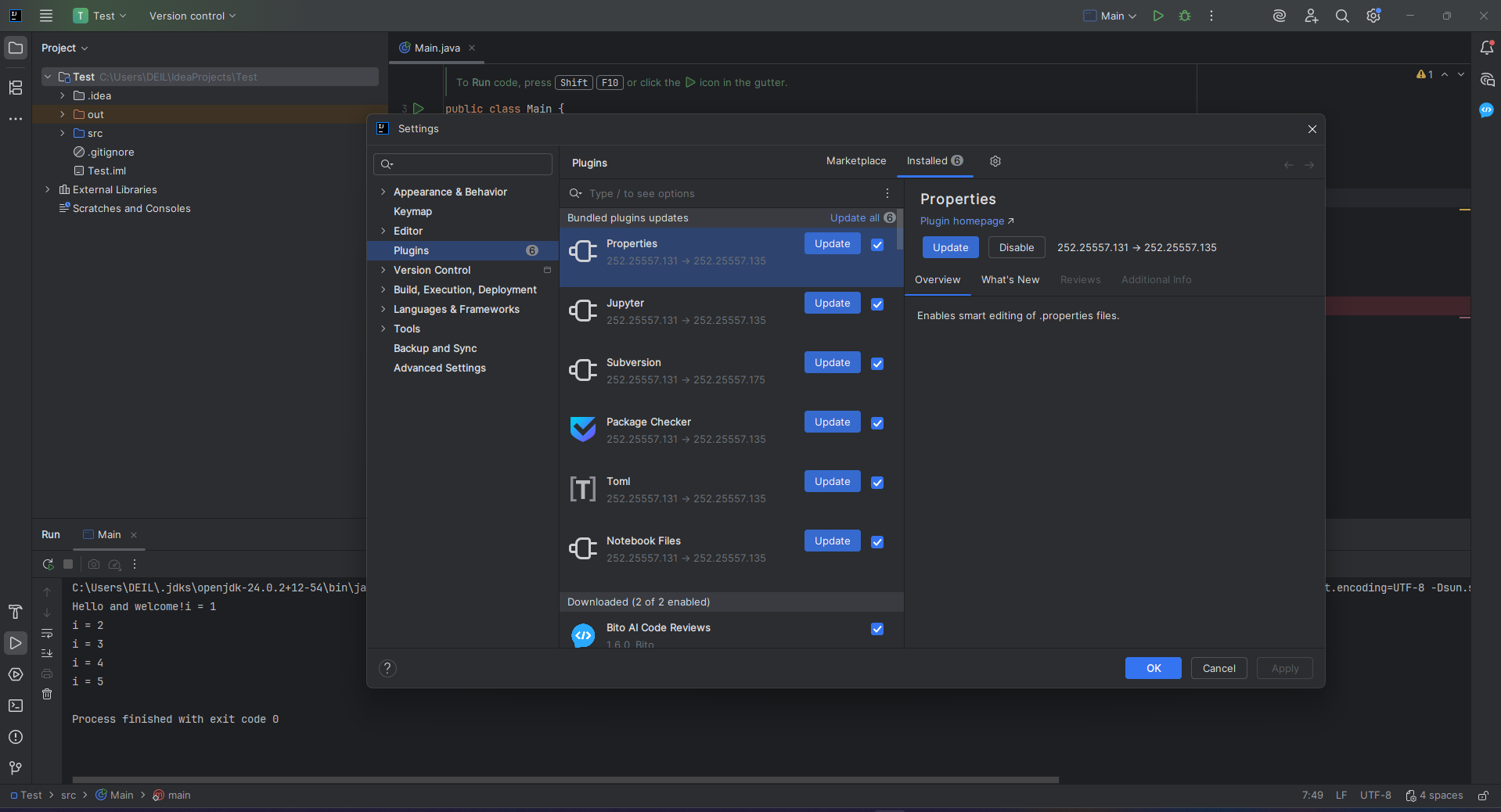


Рисунок 1 - Плагины

1. **Популярные и полезные плагины**

Некоторые из наиболее востребованных плагинов для IntelliJ IDEA:

* **Key Promoter X:** Помогает запоминать сочетания клавиш, показывая подсказки при использовании мыши для элементов IDE. Если для действия нет сочетания клавиш, плагин предложит его создать.
* **Rainbow Brackets**: Окрашивает парные скобки в разные цвета, что упрощает визуальное определение границ блоков кода.
* **Maven Helper**: Предоставляет расширенные возможности для работы с Maven-проектами, включая анализ и разрешение конфликтующих зависимостей.
* **EduTools**: Полезен как для обучения, так и для преподавания различных языков программирования (Kotlin, Java, Python и др.) прямо в IDE. Позволяет создавать упражнения и делиться ими.
* **GitToolBox**: Расширяет встроенную интеграцию с Git, добавляя такие функции, как просмотр истории строки, отображение состояния ветки и автоматическая загрузка изменений.
* **Database Navigator**: Предоставляет возможности для работы с базами данных (просмотр структуры, выполнение запросов) в IntelliJ IDEA Community Edition, где нет встроенных инструментов для БД.
* **SonarLint**: Помогает находить и исправлять проблемы в коде прямо во время написания, обеспечивая обратную связь по качеству и безопасности кода.
* **IdeaVim**: Добавляет поддержку Vim-привязок и функциональности для любителей этого текстового редактора.
* **CodeGlance**: Добавляет мини-карту кода для удобной навигации по файлу.

**Интеграция с фреймворками**

IntelliJ IDEA предоставляет мощные инструменты для интеграции с различными фреймворками, хотя возможности отличаются между Community и Ultimate редакциями. Ultimate-версия предлагает встроенную поддержку множества серверных и фронтенд-фреймворков, инструментов интеграции с базами данных и профилирования, mientras que Community Edition требует дополнительной настройки.

**Настройка фреймворков в проекте**

Для добавления поддержки фреймворка в IntelliJ IDEA необходимо:

1. Использовать диалог Add Framework Support через контекстное меню корневой директории модуля в проекте.
2. Выбрать нужный фреймворк из списка доступных.
3. Указать параметры библиотеки или разрешить IDE загрузить её автоматически.

**Работа с базами данных**

IntelliJ IDEA предоставляет мощные инструменты для работы с базами данных, но их доступность и функциональность значительно отличаются между Community и Ultimate редакциями.

**Подключение баз данных**

Для работы с БД в Community Edition требуется установка сторонних плагинов:

**Подключение MySQL к Community** **версии Intellij IDEA**

* **Установка**: File → Settings → Plugins → Marketplace → “Database Navigator»

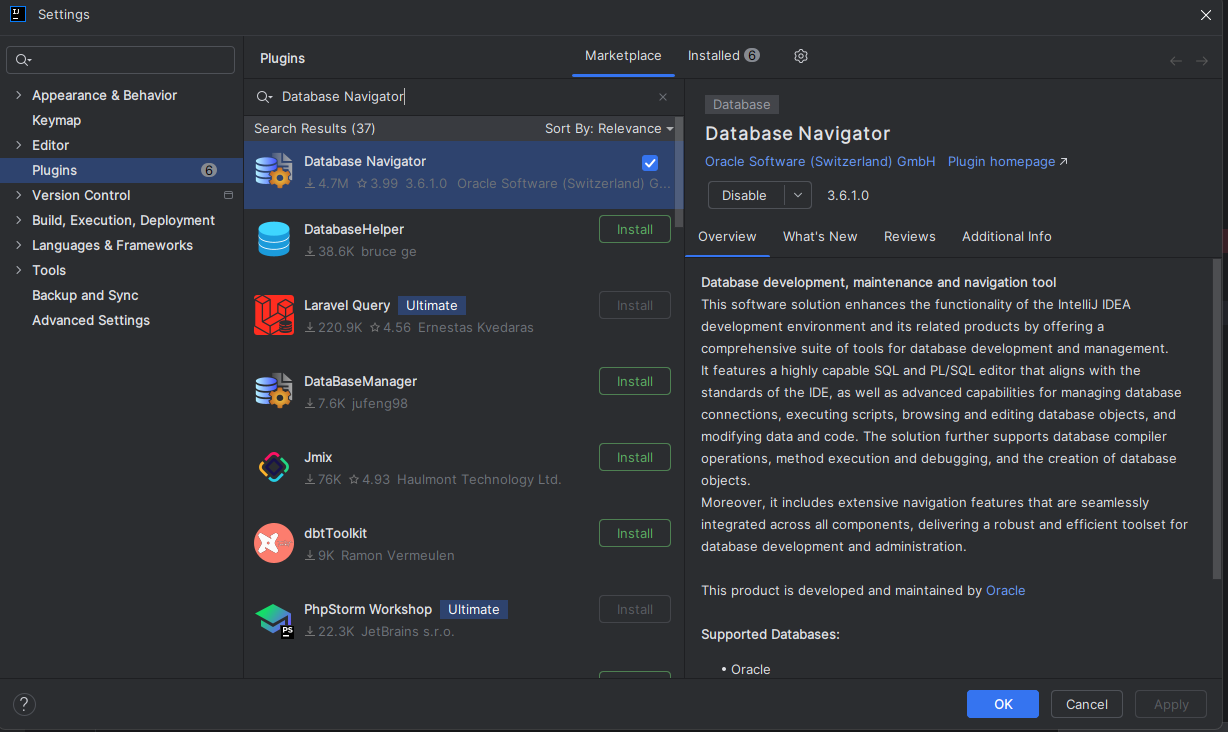


Рисунок 2 - Database Navigator

* Заходим в новое окно(DB Navigator), нажимаем плюсик и из предложенного списка выбираем MySQL

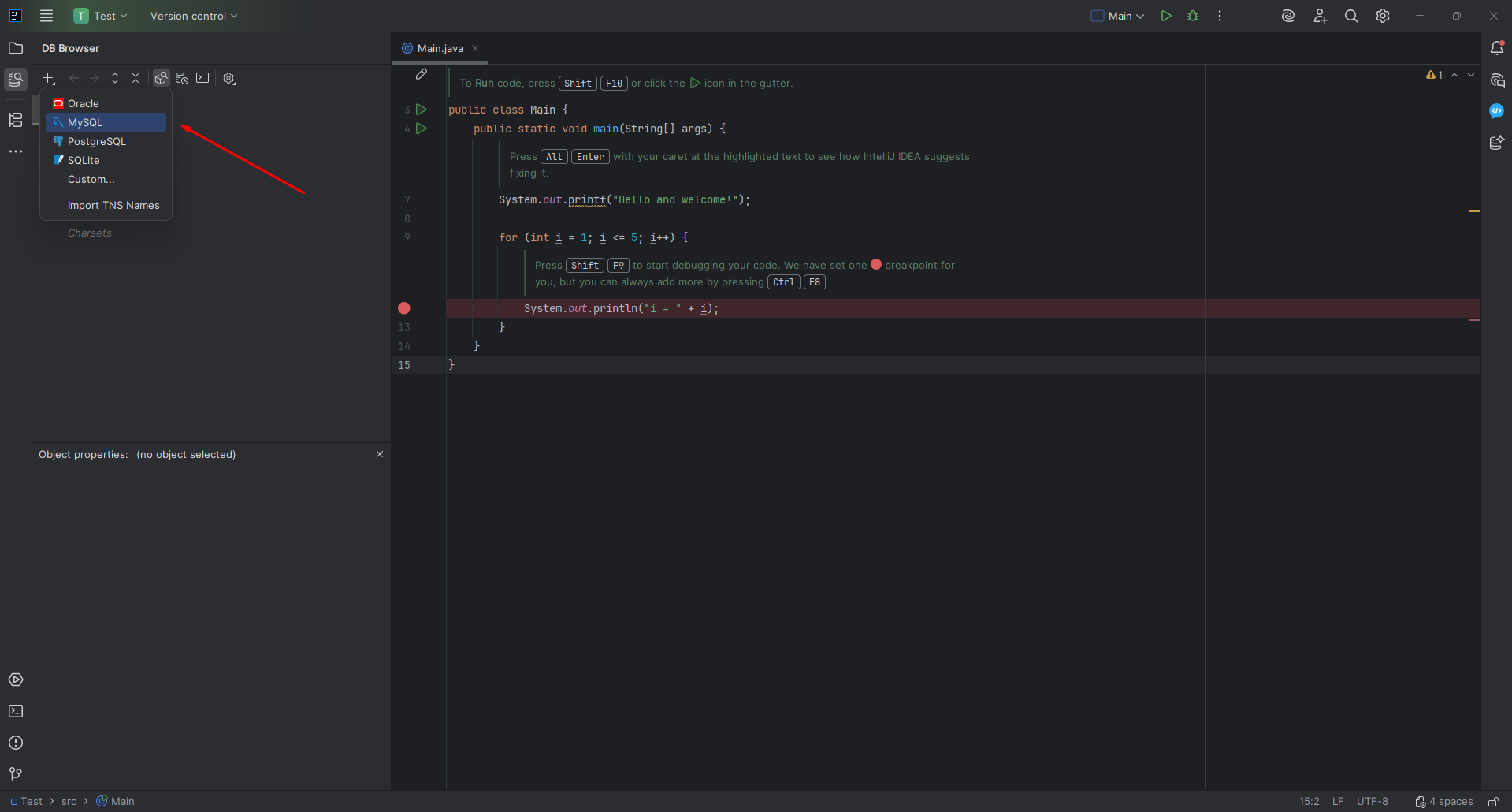


Рисунок 3 - MySQL

* В открывшемся окне заполните необходимые поля:

**Name**: Укажите желаемое название для вашей базы данных.

**Description**: Это поле является необязательным, его можно оставить пустым.

**Host** и **Port**: Оставьте значения по умолчанию без изменений.

**Database**: Убедитесь, что в этом поле указано значение mysql.

**User** и **Password**: Введите учетные данные для доступа к СУБД (обычно это root и соответствующий пароль).

После заполнения полей нажмите кнопку **Test Connection**, чтобы проверить подключение.

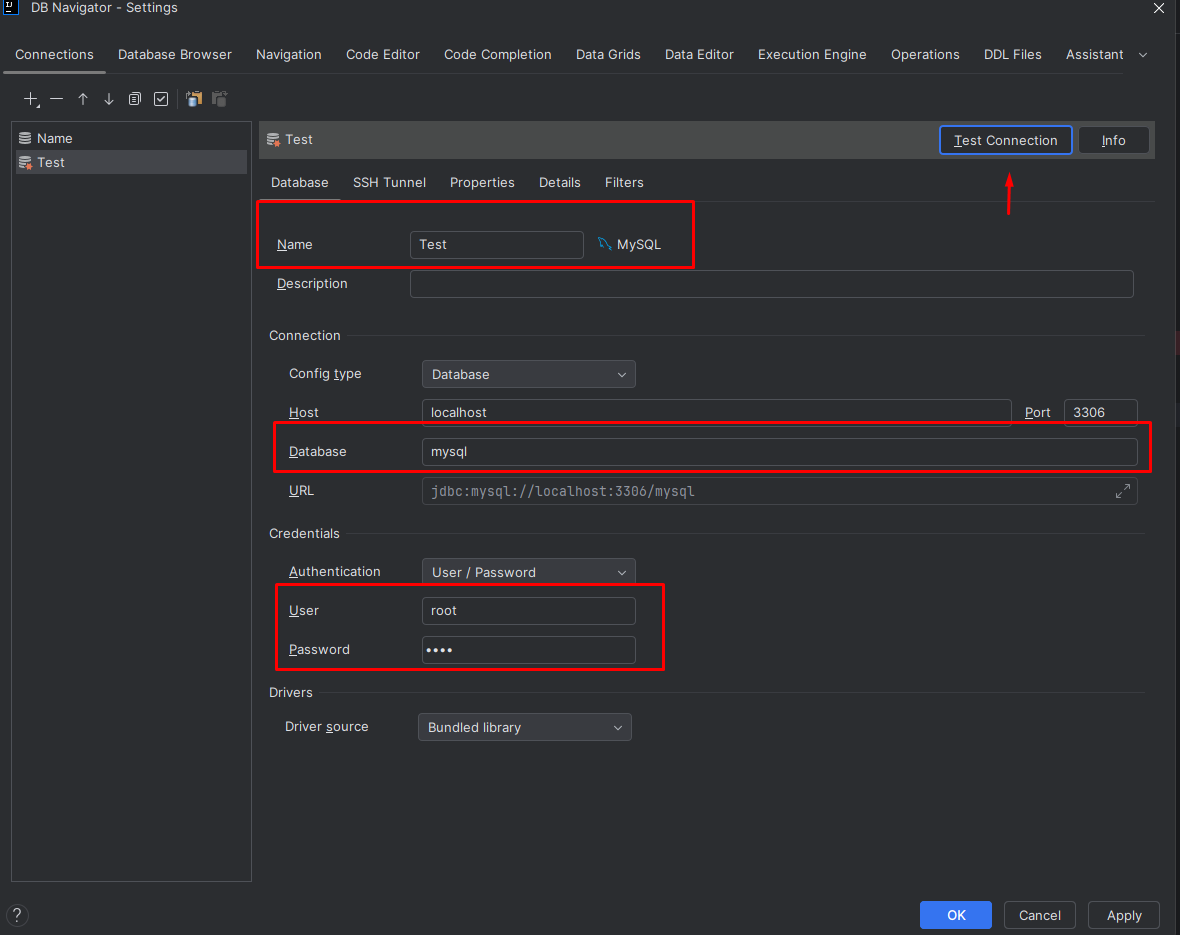


Рисунок 4 – DB Navigator - Settings

* При тестировании соединения может возникнуть ошибка временной зоны. Для её исправления открываем **MySQL Command Line Client**
* В появившемся консоле вводим пароль БД, И вводим команду “set global time\_zone = '+3:00';” (+3 это часовой пояс)

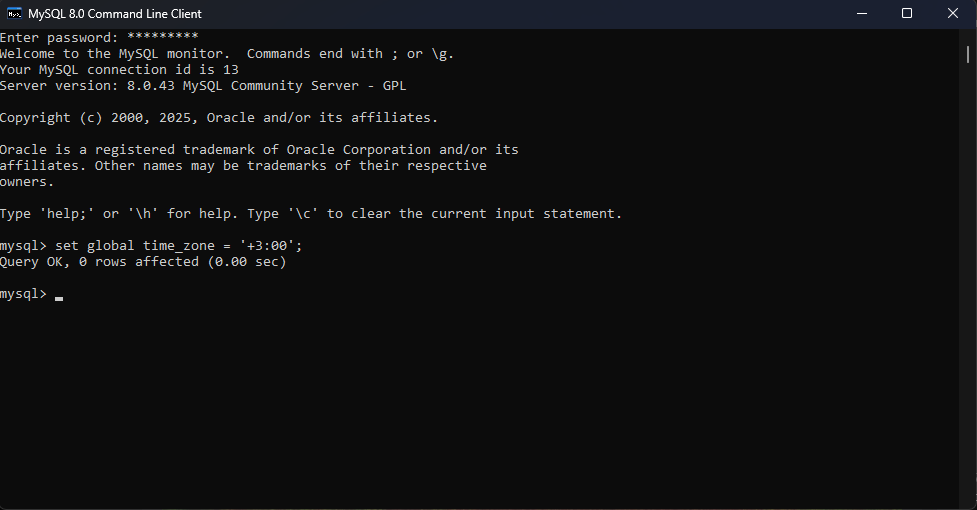


Рисунок 5 – MySQL Command Line Client

* После устранения ошибки нажмите **Apply**, а затем **OK**. В окне DB Navigator отобразится структура базы данных, после чего вы сможете просматривать её таблицы и другие объекты.

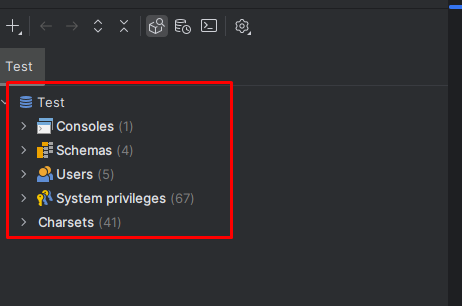


Рисунок 6 – Структура базы данных

* Рекомендуется при просматривании таблиц нажимать на кнопку **No filters**.

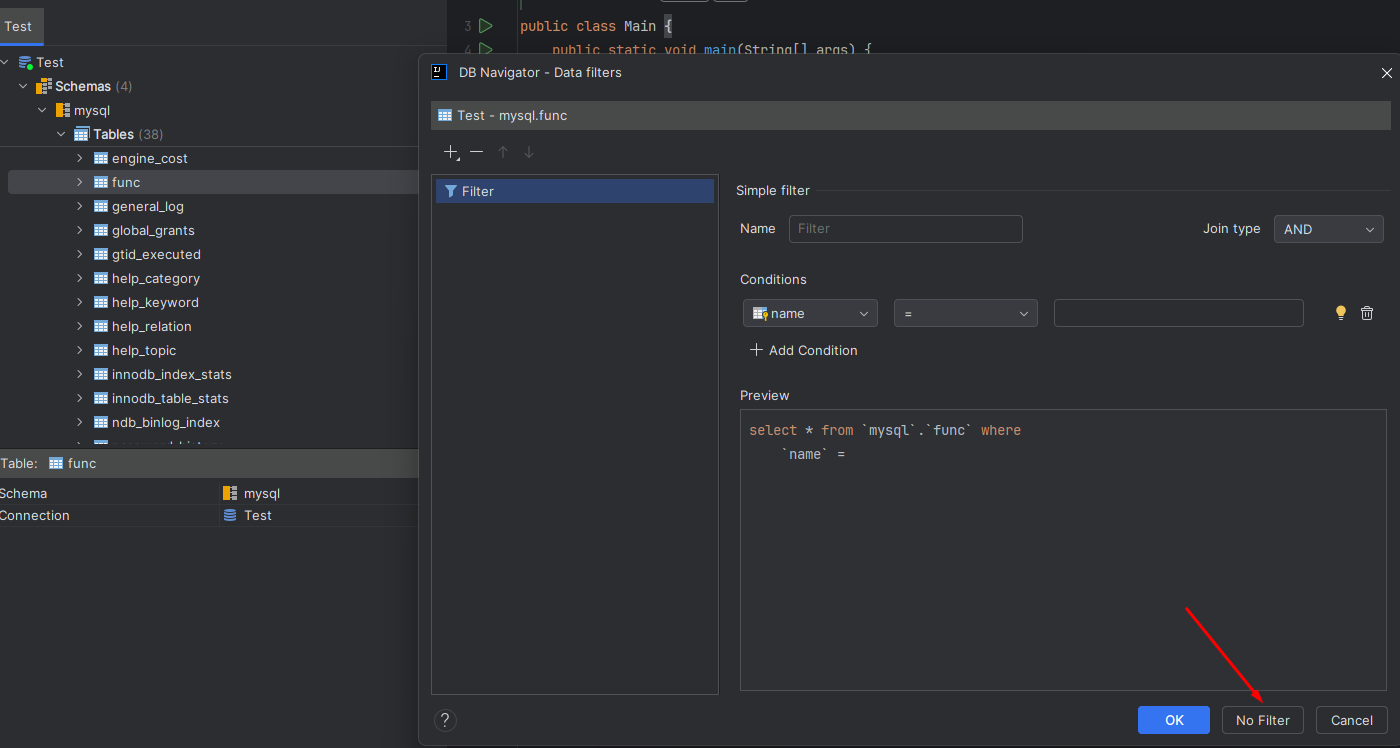


Рисунок 7 – Кнопка “No filters”

* Следующим шагом необходимо установить драйвер для работы с вашей базой данных. Для этого:

1. Перейдите на официальный сайт Oracle по адресу: dev.mysql.com/downloads/connector/j.
2. В списке доступных версий выберите вариант Platform Independent.
3. Загрузите предложенный архивный файл.

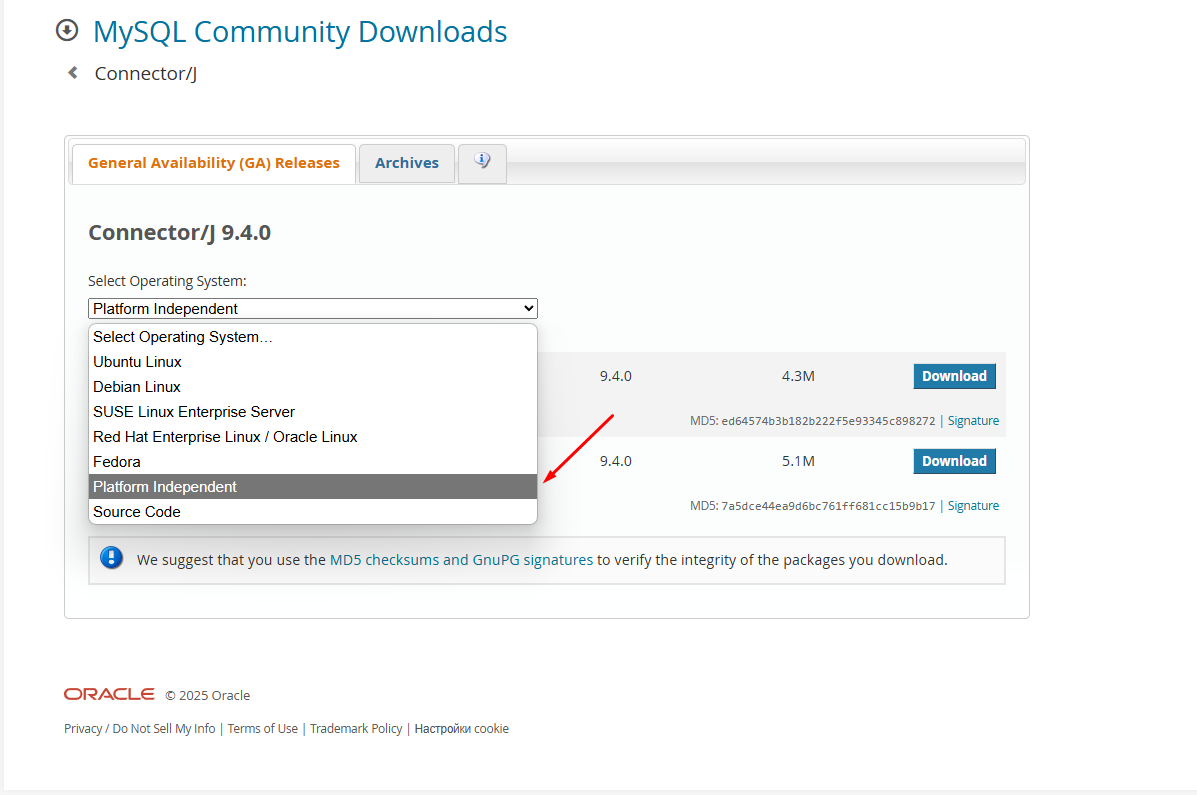


Рисунок 8 – Platform Independent

* Когда архив загрузился, открываем его и извлекаем файл “mysql-connector-j-9.4.0.jar” в папку (путь к папке нужно запомнить).
* Переходим в Intellij IDEA, там ищем **File->Project Structure ->SDK's -> плюсик который отмечен стрелкой -> ищем файл который только что скачали -> жмём ОК**.

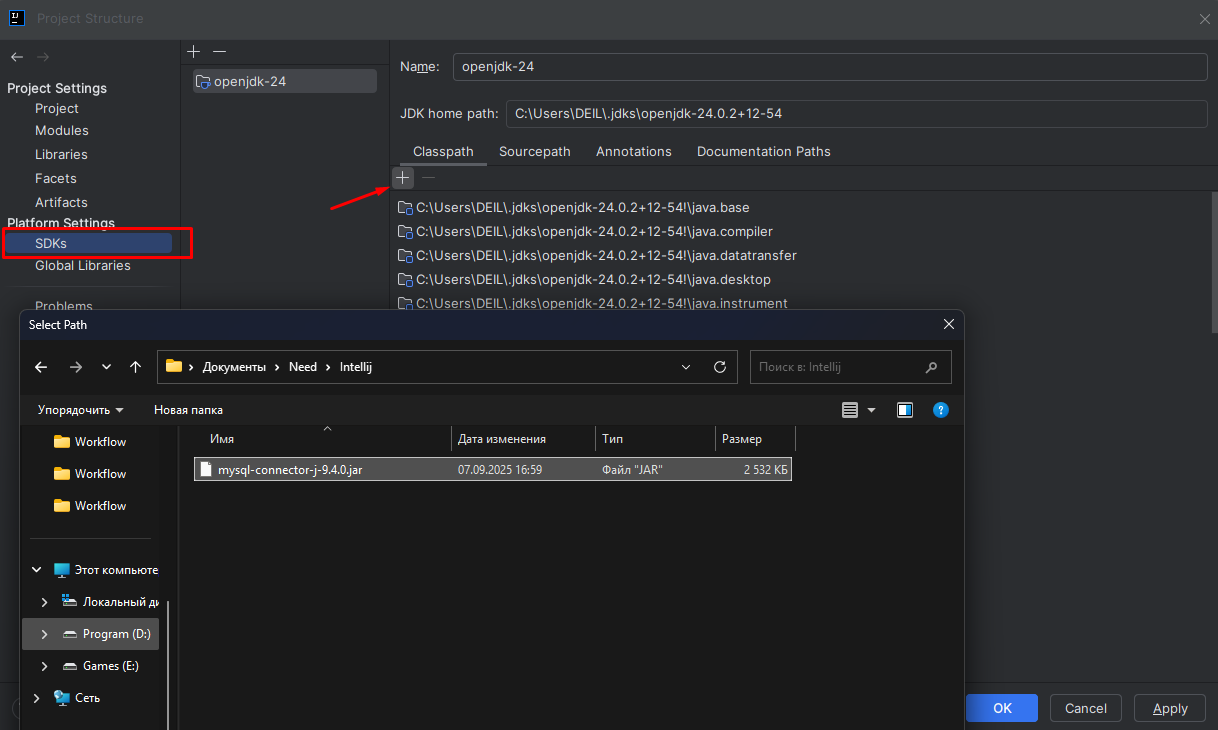


Рисунок 9 – Mysql-connector

**Возможности**:

* Подключение основных SQL-СУБД: MySQL, PostgreSQL, SQLite, Oracle
* Просмотр структуры БД и данных таблиц
* Выполнение SQL-запросов

**Ограничения**:

* Нет поддержки NoSQL-баз
* Отсутствуют ER-диаграммы

**Альтернативные решения**

* **DBeaver:** Внешний универсальный клиент с поддержкой множества СУБД.
* **СУБД-специфичные инструменты**: pgAdmin (PostgreSQL), phpMyAdmin (MySQL).

**Просмотр и редактирование баз**

* **Таблицы:** Двойной клик по таблице открывает данные в редакторе.
* **Редактирование:** Нажмите F2 для редактирования ячейки.
* **Сортировка и фильтрация:** Клик по заголовку столбца + использование панели фильтров.
* **Транзакции:** Управление коммитами через кнопки Commit/Rollback.

**Выполнение запросов**

* Открыть **Query Console** (правый клик на БД → New → Query Console).
* Написать SQL-запрос.
* Выполнить сочетанием **Ctrl+Enter** (или кнопкой Execute).
* Просмотреть результаты в нижней панели.

**Пример запроса с автодополнением**

SELECT first\_name, last\_name

FROM contact

WHERE birth\_date > '1990-01-01'

ORDER BY last\_name;

**Сравнение IntelliJ IDEA с аналогами**

**Сравнительная таблица IntelliJ IDEA, VS Code, Eclipse и NetBeans**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Критерий** | **IntelliJ IDEA** | **VS Code** | **Eclipse** | **NetBeans** |
| **Тип лицензии и стоимость** | Community Edition: бесплатная  Ultimate Edition: платная подписка (от $599/год)  С 2025.3: единый дистрибутив с бесплатным базовым функционалом | Полностью бесплатный (Open Source) | Полностью бесплатный (Open Source) | Полностью бесплатный (Open Source) |
| **Популярность среди Java-разработчиков (2025)** | 84% | 31% | 28% | <1% (входит в «другие») |
| **Основные сильные стороны** | Высокая производительность и стабильность  Глубокая интеграция с экосистемой JetBrains и мощными инструментами (БД, Spring, профайлинг)  Превосходный интеллект кода (умное дополнение, навигация, рефакторинг)  Обширная экосистема плагинов  Встроенные мощные инструменты (базы данных, Docker, HTTP-клиент)  AI Assistant (умное дополнение, рефакторинг, чат) | Невероятно легковесный и быстрый  Огромный marketplace расширений для любых языков и технологий Интеграция с GitHub Copilot  Идеален для фронтенда и мультиязычной разработки  Полностью бесплатный | Бесплатный для коммерческого использования  Высокая расширяемость  Сильное сообщество и история  Мощные инструменты для разработки под OSGi и плагины Eclipse | Прост в освоении, хорош для новичков  Нативная интеграция с Apache Maven  Хорошая поддержка Java EE |
| **Основные слабые стороны** | Ресурсоемкость (требует много RAM)  Сложность для новичков (обилие функций)  Полная функциональность Spring, БД и др. - только в Ultimate | Требует настройки под Java (установка Extension Pack)  Конфликты расширений  Меньшая «глубина» анализа кода для Java по сравнению с IDEA  Славная интеграция с отладчиком для сложных сценариев | Более высокая сложность освоения  Устаревший UI/UX (по сравнению с современными IDE)  Менее интеллектуальное дополнение кода  Медленная работа с крупными проектами | Меньше сообщество и поддержка  Медленная разработка новых функций  Слабые возможности рефакторинга по сравнению с аналогами |
| **Поддержка языков и технологий** | Java, Kotlin - наилучшая поддержка. Scala, Python, Go, SQL, JavaScript/TypeScript и др. - через плагины Глубокая интеграция с Spring, Jakarta EE, Quarkus, Micronaut (в Ultimate) | Мультиязычность - главное преимущество. Отличная поддержка JavaScript/TypeScript, Python, Go, Rust, C++ и многих других через расширения. Для Java требуется установка пакета расширений | Java - основная специализация. Хорошая поддержка C/C++, Python, PHP через плагины. Мощные инструменты для моделирования (UML) и отчетности. | Java - основная специализация. Хорошая поддержка Java EE, PHP, C/C++ через плагины. |
| **Производительность и потребление ресурсов** | Высокое потребление памяти, но отличная производительность на мощных машинах. | **Очень низкое потребление ресурсов**, быстрый запуск | Высокое потребление памяти, может быть медленным с крупными проектами | Среднее потребление ресурсов. |
| **Экосистема и плагины** | Огромный магазин плагинов JetBrains Marketplace. Высокое качество и глубокая интеграция многих плагинов. | Огромный marketplace расширений. Некоторые расширения могут конфликтовать или быть низкого качества | Большое количество плагинов, но часто с устаревшим интерфейсом. | Меньшее количество плагинов по сравнению с другими. |
| **Идеальный сценарий использования** | Корпоративная Java/Kotlin-разработка  Большие и сложные проекты  Разработка на Spring  Проекты, требующие глубокого анализа кода и рефакторинга | Фронтенд-разработка  Мультиязычные проекты  Начинающие разработчики  Сценарии, где важна легкость и скорость | Разработка под OSGi  Легаси-проекты, исторически связанные с Eclipse  Бюджетные проекты, требующие бесплатного инструмента | Обучение Java  Небольшие проекты  Разработчики, сильно зависящие от Maven |

**Ключевые выводы и рекомендации**

* + 1. **IntelliJ IDEA Ultimate** - бесспорный лидер для профессиональной enterprise-разработки на Java и Kotlin, особенно в связке с фреймворками Spring и другими технологиями JetBrains. Это выбор тех, кто ценит глубину анализа, интеграцию и производительность.
    2. **VS Code** - идеальный легковесный и универсальный редактор для мультиязычных проектов, фронтенда и сценариев, где важны скорость и простота. Его популярность растет благодаря бесплатности, легкости и обширной экосистеме расширений.
    3. **Eclipse** остается надежным бесплатным выбором для специфических задач (например, разработка под OSGi) и в компаниях с устоявшимися процессами на основе этой IDE. Однако его доля на рынке постепенно снижается.
    4. **NetBeans** проигрывает в конкурентной борьбе и выбирается в основном для обучения или для работы с legacy-проектами, сильно завязанными на его специфические возможности.
    5. **Тренд на использование нескольких IDE**: Почти половина (42%) разработчиков используют более одной IDE, часто комбинируя IntelliJ IDEA для бэкенда и VS Code для фронтенда или быстрых правок.

**Список источников**

1. Как подключить MySQL к бесплатной версии Intellij IDEA (community). — Текст : электронный // habr : [сайт]. — URL: https://habr.com/ru/sandbox/146588/ (дата обращения: 07.09.2025).
2. What’s New in IntelliJ IDEA 2025.2. — Текст : электронный // jetbrains : [сайт]. — URL: https://www.jetbrains.com/idea/whatsnew/ (дата обращения: 07.09.2025).
3. Harnessing the Power of AI in IntelliJ IDEA. — Текст : электронный // jetbrains : [сайт]. — URL: https://www.jetbrains.com/guide/ai/links/harnessing-power-ai/ (дата обращения: 07.09.2025).
4. Топ-10 плагинов для IntelliJ IDEA. — Текст : электронный // blog.jetbrains : [сайт]. — URL: https://blog.jetbrains.com/ru/idea/2021/06/top-10-plugins-for-intellij-idea/ (дата обращения: 07.09.2025).
5. Как превратить IntelliJ IDEA Community Edition в Ultimate за 3 простых шага. — Текст : электронный // amplicode : [сайт]. — URL: https://amplicode.ru/blog/kak-prevratit-intelliJ-idea-community-edition-v-ultimate/ (дата обращения: 07.09.2025).
6. IntelliJ IDEA overview. — Текст : электронный // jetbrains : [сайт]. — URL: https://www.jetbrains.com/help/idea/discover-intellij-idea.html (дата обращения: 07.09.2025).