

Основные пакеты и их назначение

1. **config** — конфигурационные файлы для приложения:
 - **SecurityConfig.java** — настраивает безопасность с использованием Spring Security.
 - Реализована базовая аутентификация с логином (**admin**) и паролем (**admin123**).
 - Эндпоинты защищены и требуют авторизации.
2. **controller** — контроллеры, обрабатывающие HTTP-запросы:
 - **TaskController.java** — REST-контроллер для управления задачами.
 - Поддерживает операции **CRUD** (создание, чтение, обновление и удаление).
3. **exception** — обработка ошибок:
 - **ResourceNotFoundException.java** — исключение для обработки случаев, когда задача не найдена (возвращает статус HTTP 404).
4. **model** — модель данных:
 - **Task.java** — сущность, представляющая задачу с полями:
 - **id** — идентификатор задачи.
 - **title** — название задачи.
 - **description** — описание задачи.
 - **status** — статус выполнения задачи (например, "Pending").
 - **createdAt** — время создания задачи.
5. **repository** — слой доступа к данным:
 - **TaskRepository.java** — интерфейс для работы с базой данных, использует **JpaRepository** для выполнения операций с задачами.

Конфигурационные файлы

1. **application.yml** (в папке **resources**):
 - Настройки базы данных (**H2 in-memory database**):
 - URL: **jdbc:h2:mem:taskdb**
 - Логин: **sa**, пароль отсутствует.

Включена консоль H2 по адресу:

<http://localhost:8080/h2-console>

- Логирование SQL-запросов.
- Автоматическое создание и обновление таблиц (DDL auto-update).
- Конфигурация безопасности с базовой аутентификацией (**admin/admin123**).

Контроллер поддерживает следующие CRUD-операции:

Получить все задачи (GET):

```
GET /api/tasks
```

1.

Создать задачу (POST):

```
POST /api/tasks
```

Body:

```
{
  "title": "Task 1",
  "description": "Complete the project",
  "status": "Pending"
}
```

2.

Обновить задачу (PUT):

```
PUT /api/tasks/1
```

Body:

```
{
  "title": "Updated Task",
  "description": "Submit the report",
  "status": "In Progress"
}
```

3.

4. Удалить задачу (DELETE):

```
DELETE /api/tasks/1
```

Безопасность

- Все эндпоинты защищены базовой аутентификацией.

Логин и пароль задаются в файле **application.yml**:

```
username: admin
```

- password: admin123

База данных (H2)

Используется встроенная база данных **H2** в оперативной памяти.
Доступ к консоли базы данных:

`http://localhost:8080/h2-console`

Параметры:

- **URL:** `jdbc:h2:mem:taskdb`
- **User:** `sa`
- **Password:** *(пусто)*

Тестирование

1. Для тестирования используется **JUnit 5** и **Spring Boot Test**.
2. Реализован базовый тест в `DemoApplicationTests.java`, проверяющий загрузку контекста.
3. Добавлена поддержка мок-пользователей для тестов безопасности.

Примечания

- Проект поддерживает **Spring Security** для защиты API.
- Все зависимости описаны в `pom.xml`, включая Lombok для упрощения кода (геттеры/сеттеры).
- Локальная база данных очищается при перезапуске приложения.