

1. Импортировать maven зависимости
2. Установить соответствующие настройки 'spring.datasource' в файле 'application.properties'
Путь к файлу: FinalProject\src\main\resources\application.properties
3. Maven Projects > Lifecycle > package > run maven build
4. Запуск приложения с файла
Путь к файлу: FinalProject\FinalProject\src\main\java\findev\Application.java
5. Инициализировать файлы для создания и заполнения базы данных
Путь к файлу: FinalProject\src\main\resources\initDB.sql
Путь к файлу: FinalProject\src\main\resources\populateDB.sql

Сотрудники

- **done** CRUD методы

- a) чтение по id [GET] <http://localhost:8080/employees/1>
- b) чтение всех [GET] <http://localhost:8080/employees>
- c) добавление [POST] <http://localhost:8080/employees>

```
request body {  
  "firstName": " Raj",  
  "lastName": " Koothrappali",  
  "email": " koothrappali@mail",  
  "hourRate": 15,  
  "positionId": 2,  
  "departmentId": 2,  
  "statusId": 3  
}
```
- d) редактирование [POST] <http://localhost:8080/employees> + `RequestBody`

```
request body {  
  "id": 4,  
  "firstName": " Raj",  
  "lastName": " Koothrappali",  
  "email": " koothrappali@mail",  
  "hourRate": 15,  
  "positionId": 2,  
  "departmentId": 2,  
  "statusId": 3  
}
```
- e) удаление по id [POST] <http://localhost:8080/employees/3>

Событие

- **done** событие может быть создано администратором либо модератором
- **done** формат создания события:
[POST] <http://localhost:8080/events>

```
request body {  
  "eventTypeId": 1,  
  "date": "2018-01-12",  
  "hours": 8,  
  "employeesIds": [1,2,3]  
}
```
- **done** количество часов применимо ко всем сотрудникам события

- **done** событие будет добавлено в базу данных только в том случае если все из перечисленных сотрудников имеют статус "Free" на момент создания события
- **not done** после создания события статусы всех перечисленных в событие сотрудников изменяются на "Busy" и так же автоматически будут изменены на "Free" по истечению времени указанного в событие

Сотрудник

Тестовый gmail аккаунт для всех сотрудников:

gmail: **findevemployee@gmail.com**
password: **q7k4b8j1w5**

- **done** получает на электронную почту оповещение о добавлении в базу данных содержащие username и сгенерированный временный пароль
- **not done** может изменять свой пароль
- получает первого числа каждого месяца электронное письмо содержащее информацию о зарплате сотрудника за предыдущий месяц
- имеет доступ к:
 - a) **done** чтению данных о себе – возвращает все данные о сотруднике:

```
[GET] http://localhost:8080/employees/currentuser
response body {
  "firstName": "Sheldon",
  "lastName": "Cooper",
  "email": "cooper@mail",
  "hourRate": 15,
  "positionName": "Physicist",
  "departmentName": "Physics",
  "statusName": "Free"
  "suerUsername": "sheldoncooper"
}
```
 - b) **done** получению данных о своей зарплате за любой период

```
[GET] http://localhost:8080/income/currentuser/{dateFrom}/{dateTo}
response body: bigdecimal
```
 - c) **not done** чтению данных о своем рабочем времени – возвращает список событий попадающих в указанный промежуток времени в которых принимал участие сотрудник:

```
[GET] http://localhost:8080/events/currentuser
response body [
  {
    "eventTypeName": "Working day",
    "date": "2018-01-02",
    "hours": "8"
  }
  {
    "eventTypeName": "Presentation",
    "date": "2018-01-03",
    "hours": "7"
  }
  {
    "eventTypeName": "Technical study",
    "date": "2018-01-06",
    "hours": "3"
  }
]
```

Доход сотрудника

- **done** доход сотрудника от одного события рассчитывается как часы события умноженные на почасовую ставку сотрудника умноженную на коэффициент соответствующего события ('Working day': 1.0, 'Technical study', 0.8, 'Presentation', 0.6 и тп)
- **done** зарплата сотрудника равна сумме доходов от всех событий за месяц в которых принимал участие сотрудник
- **done** отдельной таблицы для сохранения зарплаты не предусмотрено

Модератор

- **done** может добавлять сотрудников в базу данных:
[POST] <http://localhost:8080/employees>
request body {
 "firstName": " Raj",
 "lastName": " Koothrappali",
 "email": " koothrappali@mail",
 "hourRate": 15,
 "positionId": 2,
 "departmentId": 2,
 "statusId": 3
}
}
- **done** после добавления модератором нового сотрудника в базу данных автоматически создается аккаунт сотрудника:
 - a) **done** добавляется в таблицу employees
 - b) **done** добавляется в таблицу users
 - c) **done** отправляется электронное письмо на указанный электронную почту сотрудника
- **done** получать данные о зарплате сотрудников за любой период
[GET] <http://localhost:8080/income/{firstName}/{lastName}/{dateFrom}/{dateTo}>
response body: bigdecimal