**Stack технологий**

Spring boot (starter data JPA, starter web), любая резидентная (in-memory) база данных, Maven, Git.

**Задание**

Реализовать REST приложение позволяющие вести учет сотрудников организации. Так же должна быть возможность вести учет «списков ознакомления» сотрудников с новыми правилами организации. После выполнения задания должна быть предоставлены ссылка на Git репозиторий с проектом.

**Обязательные** **требования к структуре таблиц**

*Дизайн структуры БД определяются* ***на усмотрение выполняющего****!*

1. Список обязательных полей, для сущности «сотрудник»:

UUID id (обязательно типа UUID)

String name not null

Integer age not null

Boolean married not null

Employee boss руководитель сотрудника, сотрудник может быть руководителем у нескольких сотрудников

1. Список обязательных полей для сущности «лист ознакомления»:

Long id - индфикатор сущности должен быть типа Long

String name not null - пример: «Приказ по правилам техники безопасности»

String text not null - пример: «Длинный текст приказа…»

Instant createDate not null - дата создания листа

Лист ознакомления должен хранить метку времени, когда сотрудник ознакомился с правилами.

**Требования к REST API**

Должна быть обработка ошибок. Все exceptions должны «прокидываться» на фронт.

Валидация параметров на уровне контролера.

База данных должна генерироваться скриптом (скрипт расположить в resources)

1. Сущности «Сотрудник»

Реализовать методы, такие как:

* удаление нескольких сотрудников
* получить всех подчинённых сотрудника
  + получить всех сотрудников организации. Должна быть пагинация через limit и offset. Должна быть сортировка по полям: name, age, married. Если будет доступна сортировка по полям, которые не вошли в этот список – будет считаться за ошибку. Реализовать поиск по полям: name (like), age (equals), married (equals)
* реализовать возможность исключить не ознакомившихся сотрудников из списка передав «массив» их ID.

2. Сущность «Лист ознакомления»

Реализовать методы, такие как:

* удаление нескольких листов ознакомления
* возможность удалять несколько сотрудников из листа за раз. Нельзя удалить сотрудника из листа ознакомления, если он уже ознакомился (поставил метку) в таком случае выдавать ошибку
* добавление сотрудника в лист ознакомления. В «лист ознакомления» нельзя добавить сотрудника дважды, если отсутствует метка ознакомления. Если метка стоит, то сотрудника можно добавить повторно.
* получить все листы ознакомления, где конкретный сотрудник не проставил отметки об ознакомлении
* получить все листы ознакомления (Должна быть пагинации через limit и offset, должна быть сортировка по полю name, если будет доступна сортировка по полям, которые не вошли в этот список будет считаться за ошибку. Так же добавить поиск по полям name(like).

**Опциональное задание**

Код необходимо покрыть JUnit тестами.