**Введение в FastApi**

1. Создать пустую директорию с именем **booking\_app**. Создать и активировать виртуальное окружение. Добавить файл **.gitignore** в пустую директорию. Установить библиотеки **fastapi, uvicorn, pydantic**. Создать файл **requirements.txt** добавить все установленные библиотеки.
2. Создать директорию **src** . Внутри директории создать файл c именем **main.py** . Создать объект класса **FastAPI() .** Создать один эндпоинт с **get** запросом, который будет возвращать список из 5 случайных чисел. Числа генерируются с помощью модуля **random**.
3. Создать 3 эндпоинта для **get** запросов, которые возвращают словари. 1 возвращает данные которые прокидываются в качестве **path params** название **{item\_id}** тип **int**. 2 возвращает данные переданные в качестве **query params.** Имена произвольные двух типов **int** и **str.** 3 возвращает 1 **path params** тип str и два **query params** тип **int, int**
4. Используя **pydantic** создайте класс **Actor** c полями **(actor\_id int, name str, surname str, age int, sex str)** Создайте эндпоинт c **post** запросом который принимает в **body** (**actor: Actors**). и возвращает актёра с переданными полями в теле запроса.
5. Используйте возможности pydantic и провалидируйте поля класса **Actor** следующим образом: **actor\_id** сделайте положительным числом, **age** определите от 0 до 100, **name** и **surname** сделайте не более 20 символов и не менее 2, **sex** добавьте возможность выбора только **male** и **female** .

**Дополнительные задания**

1. Создайте движок для подключения к Sqlite. И подключитесь к базе **cinema.** Измените эндпоинт созданный в 4 пункте, добавив в него сохранение в таблицу **actors**, данных об актёре полученных в body. Для сохранения в базу используйте на выбор или raw Sql или sqlalchemy orm.
2. Cоздайте эндпоинт c **get** запросом, который будет принимать в **path params** параметр **{actor\_id}.** И с помощью raw Sql или sqlalchemy orm получите данные об актёре с полученным actor\_id или выведите, что такого актёра нет в БД.