Описание структуры проекта (логика работы программы подробно описана в комменатриях):

**Класс** DBStudent – создан для работы с таблицой students

**Класс** DBDiscipline – создан для работы с таблицой students

**Класс** Config - Конфигурационный файл для хранения констант (для подключения к БД)

DBStudent - имплементирует два интерфейса:

1. ActionStudent - описано поведение класса DBStudent
2. Constansts – класс для хранения констант.

DBDiscipline- имплементирует два интерфейса:

1. ActionDiscipline - где описано поведение класса DBDiscipline
2. Constansts – класс для хранения констант.

Все методы (DELETE, PUT, GET) классов DBStudent, DBDiscipline - возвращают оболочку Optional, в котором хранится информация: запрос был успешный или не успешный. То есть, если вызван метод **GET** (optional хранит информацию: результат запроса, либо то, что запрос был неудачный и такого элемента в БД нет), **PUT** (optional хранит информацию: сколько строк изменилось: 0 – не получилось внести информацию в БД, >0 – строка успешно занесена), **DELTE** (optional хранит информацию: сколько строк изменилось: 0 – не получилось удалить строку в БД, >0 – строка успешно удалена).

**Утилитный** **класс** SrvDB – для работы с двумя таблицами одновременно. Он позволяет идентифицировать и правильно вызывать метод в нужную таблицу и обработать полученное значение.

**Утилитный Класс** SrvInput – класс для работы с вводом и определением какой из методов нужно вызвать (GET, PUT, DELETE)

**Класс** Main c него стартует программа. Там есть функция, для запуска тестов и предоставлена возможность писать запросы самостоятельно через консоль.