

Паттерны программирования
Лабораторная работа № 6. Фильтрация и сортировка

Каждое задание должно быть загружено на личный git-репозиторий отдельным коммитом. Лабораторная работа выполняется в одной папке. Защита работы возможна на любой лабораторной работе от 1 до 16. Каждое из 4 заданий проверяется отдельно с учетом вопросов преподавателя. Задание засчитывается отдельно, лабораторная работа зачтена в случае выполнения всех 4 заданий.

Если часть задач выполнена в один коммит, работа не проверяется. Если все коммиты сделаны в один час, работа не проверяется.

Задание 1. Фильтрация по одному полю.

Задачи

1. Создать возможность реализовывать метод `get_k_n_student_short_list` так, чтобы данный метод или вновь написанные возвращали список студентов, у которых есть гит, и список студентов, гит которых содержит введенную подстроку для источника данных БД.
2. Построить в контроллере обработчик кнопки ОБНОВИТЬ. При нажатии на эту кнопку должна происходить выгрузка данных, отфильтрованных по пункту 1, после чего вызов уже готового метода `Refresh_data`. Обработать исключительные ситуации в контроллере.
3. Построить UNIT тесты для измененных или новых методов.
4. Построить диаграмму последовательности для прецедента Отфильтровать записи о студентах по гиту.
5. Внести изменения в диаграмму классов.

Задание 2. Фильтрация по всем полям в БД.

Задачи

1. Разобрать паттерн обертка(wrapper) и привести тривиальный пример, можно из учебника.
2. Организовать паттерн обертка для реализации возможности поиска по полям ФамилияИнициалы, гит, и наличию гита или его отсутствию для источника данных БД.

3. Обновить в контроллере обработчик кнопки ОБНОВИТЬ. При нажатии на эту кнопку должна происходить запись нового состояния в Student_list, после чего вызов уже готового метода Refresh_data. Обработать исключительные ситуации в контроллере.
4. Для корректной реализации предыдущего пункта придется внести изменения в класс Student_short, чтобы обеспечить выдачу требуемого поля.
5. Изменить диаграмму последовательности для прецедента Отфильтровать записи о студентах.
6. Построить UNIT тесты для измененных или новых методов.
7. Внести изменения в диаграмму классов.
8. Протестируйте работу программы на БД.
9. * Организовать паттерн обертка для реализации возможности поиска по полям ФамилияИнициалы, гит, телефон, и наличию телефона, гита, или их отсутствию для источника данных БД.
10. Для корректной реализации предыдущего пункта придется внести изменения в класс Student_short, чтобы обеспечить выдачу требуемого поля.
11. Изменить диаграмму последовательности для прецедента Отфильтровать записи о студентах.
12. Построить UNIT тесты для измененных или новых методов.
13. Внести изменения в диаграмму классов.
14. Протестируйте работу программы на БД.
15. * Организовать паттерн обертка для реализации возможности поиска по полям ФамилияИнициалы, гит, телефон, телеграмм, и наличию телефона, телеграмма, гита, или их отсутствию для источника данных БД.
16. Для корректной реализации предыдущего пункта придется внести изменения в класс Student_short, чтобы обеспечить выдачу требуемого поля.
17. Изменить диаграмму последовательности для прецедента Отфильтровать записи о студентах.
18. Построить UNIT тесты для измененных или новых методов.
19. Внести изменения в диаграмму классов.
20. Протестируйте работу программы на БД.
21. * Организовать паттерн обертка для реализации возможности поиска по полям ФамилияИнициалы, почта, телефон, телеграмм, гит, и наличию почты, телефона, телеграмма, гита, или их отсутствию для источника данных БД.
22. Для корректной реализации предыдущего пункта придется внести изменения в класс Student_short, чтобы обеспечить выдачу требуемого поля.
23. Изменить диаграмму последовательности для прецедента Отфильтровать записи о студентах.
24. Построить UNIT тесты для измененных или новых методов.

25. Внести изменения в диаграмму классов.
26. Протестируйте работу программы на БД.

Вопросы.

- a. Опишите, в каком случае рекомендуется применять паттерн Состояние.
- b. Напишите на бумаге пример такого паттерна.
- c. Опишите, в каком случае рекомендуется применять паттерн Обертка.
- d. Напишите на бумаге пример такого паттерна.
- e. Опишите Вашу реализацию паттерна обертка.

Задание 3. Фильтрация в файлах.

Задачи

1. Внесите коррективы так, чтобы на БД и файле JSON/YAML/TXT функционал работал.
2. Отметьте коррективы на диаграмме классов.

Задание 4. Сортировка.

Задачи

1. Внести модификации в класс Student_list и метод get_k_n_student_short так, чтобы возвращались первые k объектов, отсортированные по необходимому полю.
2. Внести модификации в классы так, чтобы при нажатии на столбец Фамилия Инициалы происходила сортировка записей студентов по этому полю.
3. Изменить диаграмму последовательности для прецедента Отфильтровать записи о студентах и Сортировать записи о студентах.
4. Построить UNIT тесты для измененных или новых методов.
5. Внести изменения в диаграмму классов.
6. Протестируйте работу программы на БД.
7. Внесите коррективы так, чтобы на БД и файле JSON/YAML/TXT функционал работал.
8. Изменить диаграмму последовательности для прецедента Отфильтровать записи о студентах и Сортировать записи о студентах.
9. Построить UNIT тесты для измененных или новых методов.
10. Внести изменения в диаграмму классов.

11. Внести модификации в классы так, чтобы при нажатии на столбец происходила сортировка записей студентов по этому полю.
12. Протестируйте работу программы на БД.
13. Изменить диаграмму последовательности для прецедента
Отфильтровать записи о студентах и Сортировать записи о студентах.
14. Построить UNIT тесты для измененных или новых методов.
15. Внести изменения в диаграмму классов.
16. Внесите коррективы так, чтобы на БД и файле JSON/YAML/TXT функционал работал.
17. Изменить диаграмму последовательности для прецедента
Отфильтровать записи о студентах и Сортировать записи о студентах.
18. Построить UNIT тесты для измененных или новых методов.
19. Внести изменения в диаграмму классов.