

Задание

В рамках выполнения практической работы требуется выполнить следующие задачи и продемонстрировать полученные навыки:

- Создать новое node.js приложение;
- Подключить к приложению библиотеки express.js и liquid;
- Разработать HTML-шаблоны для страниц приложения;
- Реализовать логику обработки на сервере согласно варианту задания;
- **Опционально! +3 бонусных балла.** Использовать авторизацию, обеспечив работу нескольких разных пользователей (сохранять информацию о пользователях можно куда угодно, хоть в глобальные переменные);
- **Оценивается отдельно до 5 баллов!** Реализовать функциональность доступа к БД и хранение данных в БД, необходимые для выполнения задания;
- **Оценивается отдельно до 5 баллов!** Применить React.js в приложении.

Для всех заданий в простейшем случае предполагаем, что объекты, о которых идет речь в вариантах задания описываются одной строкой. А вот в большинстве бонусных заданий объекты придется расширить.

Если вы не используете базу данных для хранения, то вполне нормально будет использовать глобальные переменные или сессии.

Пароль для PostgreSQL на машинах института - 12345 или postgres123

Распределение вариантов заданий

Варианты заданий распределяются по первой букве фамилии студента.

Первая буква фамилии	Вариант
А,Б,В,Г,Д,Е,	1
Ж,З,И,К,Л,М,	2
Н,О,П,Р,С,Т,	3
У,Ф,Х,Ц,Ч,Ш,Щ,Ю,Я	4

Варианты

Вариант 1

Необходимо реализовать простое приложение со списком покупок. Не с бумажкой ведь ходить по магазину?

Приложение должно иметь следующую функциональность:

- отображение товаров в виде трех списков - запланированные, купленные, отсутствующие;
- добавление товара в список запланированных при помощи соответствующей формы;
- перенос товара из списка запланированных в список купленных нажатием кнопки;
- перенос товара из списка запланированных в список отсутствующих нажатием кнопки;

- очистка списка кнопкой (удаляются все товары из всех списков).

При добавлении авторизации можно считать, что у каждого пользователя может быть только один экземпляр списков запланированные, купленные, отсутствующие.

Также можно добавить функцию передачи списка для покупок другому пользователю (т.е. пользователь выбирает другого пользователя и этому пользователю в запланированные передаются товары из списка текущего пользователя (+3 бонусных балла)).

Вариант 2

Необходимо реализовать простое приложение для подсчета стоимости работы копирайтера. Представьте себе, что копирайтеру платят за каждую букву в тексте. Не руками же считать это всё?

Приложение должно иметь следующую функциональность:

- возможность добавить в список несколько текстов;
- для каждого текста выводить количество букв;
- для каждого текста должна быть возможность установить стоимость каждой буквы;
- считать и выводить итоговую сумму по всем текстам.

При добавлении авторизации у каждого пользователя должен быть свой набор текстов.

Также можно добавить функцию, которая бы позволяла вычислять общую сумму только для выбранных текстов (т.е. у каждого текста отображается чекбокс для его выделения). Можно сделать подсчет как на клиенте, так и на сервере (по нажатию кнопки) (+3 бонусных балла)

Вариант 3

Необходимо реализовать простое приложение для организации домашней библиотеки. Ведь здорово когда на полке все книги стоят по алфавиту. Не в уме же их сортировать?

Приложение должно иметь следующую функциональность:

- возможность добавить в список список книги (книгу можно описать одной строкой текста);
- возможность указать количество и вместимость полок (2 числа, например 5 полок и 10 книг);
- выводить список добавленных книг в алфавитном порядке по полкам;
- возможность удалить книгу из списка.

Очевидно, что при добавлении авторизации у каждого пользователя будет свой набор книг и количество полок.

Также можно немного поменять логику расчета количества книг на каждой полке. Это можно сделать если помимо названия книги, для каждой книги вводить еще и количество страниц. Далее вместимость полки указывать в страницах. Это позволит учесть разные размеры книг на полке и более точно разместить книги в библиотеке. Ну и еще это позволит получить +3 бонусных балла.

Вариант 4

Необходимо реализовать простое приложение для организации работы отеля. Надо ведь как-то отслеживать что происходит с номерами в вашем отеле. Не на бумаге же это всё записывать?

Приложение должно иметь следующую функциональность:

- отображение номеров в виде трех списков - свободные, забронированные, занятые;
- добавление новых номеров (сначала в список свободные);
- возможность перевода отдельных номеров в список забронированных или занятых (ну и из забронированных в занятые);
- возможность перевода номеров из забронированных или занятых в свободные;

Очевидно, что при добавлении авторизации у каждого пользователя будут свои списки номеров.

Также можно реализовать еще один список – уборка, который будет работать так же, как и все остальные. Но это даст вам *+3 бонусных балла*.