Общие замечания

расположение тестового сервера:

http://epam-js-sommer2017.eu-4.evennode.com

Репозиторий с исходниками

https://github.com/ewalker1974/epam-js-server

установка npm update

запуск npm start

в режиме отладки

Linux

DEBUG=myapp:\* npm start

Windows

set DEBUG=myapp:\* & npm start

По умолчанию используется порт 3000

для того, чтобы поменять порт

Linux

DEBUG=myapp:\*;PORT=8080 npm start

Windows

set DEBUG=myapp:\*;PORT=8080 & npm start

Все сервисы доступны по адресу api

\*

1. Для приложения, для рисования графика произвольной функции сделать следующие модификации:  
   Перед отрисовкой графика отобразить список сохраненных графиков : GET /api/functions.  
   Отобразить кнопки загрузить и удалить.   
   При нажатии кнопки загрузить необходимо отобразить параметры графика в элементах редактирования и отобразить график  
   Для загрузки графика можно использовать GET /api/functions/function/:graphId, где  
   graphId - внутренний идентификатор записи  
   При нажатии кнопки удалить необходимо удалить выбранный в списке график   
   для удаления использовать DELETE /api/functions/function , параметр function\_id - внутренний идентификатор записи.  
   Отобразить около области редактирования кнопку сохранить.  
   При сохранении нового графика использовать   
   POST /api/functions/ Параметры:  
   name - имя графика;

graph\_function - выражение для функции;

min\_x - минимальное значение аргумента;

max\_x - макимальное значение аргумента;

color - цвет графика.

Для обновления данных использовать

PUT /api/functions/function/:graphId,  
 где graphId - внутренний идентификатор записи.  
 Остальные параметры как и в POST

1. Для графического редактора добавить следующий функционал  
   Перед отрисовкой графика отобразить список сохраненных графиков : GET /api/sketches.  
   Отобразить кнопки загрузить и удалить.  
   При нажатии кнопки загрузить необходимо отобразить имя графика в заголовке редактора и содержимое графика на “холсте”.  
   Параметры графика можно загрузить GET /api/sketches/graph/:graphId, где  
   graphId - внутренний идентификатор записи.  
   При нажатии кнопки удалить необходимо удалить выбранное в списке изображение.   
   Для удаления использовать DELETE /api/sketches/graph, параметр graph\_id - внутренний идентификатор записи.  
   Отобразить около области редактирования кнопку сохранить.  
   При сохранении нового изображения использовать  
   POST /api/sketches/, параметр  
   name - имя изображения,   
   objects - описание изображения в виде массива JSON объектов (преобразованных в строку, например [{“type”:“point”,”x”:1,”y”:2,”color”:”red”},{“type”:“circle”,”x”:100,”y”:20,’’radius”:5,”color”:”red”}] )  
   При обновлении изображения использовать /api/sketches/graph/:graphId, где  
   graphId - внутренний идентификатор записи.  
   Остальные параметры как в POST

1. Расширить приложение проверки тестов следующим образом:  
   Элемент <div> для идентификации теста необходимо добавить следующие параметры  
   data-user - имя студента, который будет проходить тест  
   data-test-id - наименование теста  
     
   Перед прохождением теста необходимо загрузить журнал предыдущего   
     
    Для заданной таблицы атрибутом data-interact реализовать следующие дополнения  
   если задан параметр data-rows=‘число‘ отобразить первое число строк и пагинатор следующего вида 1 2 … n. При клике на цифру отобразить соответствующую порцию данных и обновить пагинатор  
   если задан параметр data-sorting произвести сортировку данных по соответствующему полю при клике на его заголовок (ячейка в первой строке)

если задан параметр data-filter отобразить в первой строке text input который отображает строки таблицы совпадающие по первым символам со значением фильтра

\*\*

Реализовать приложение для отображения таблицы в стиле excel. Таблица содержит заданное число строк и столбцов они пронумерованы строки числами, столбцы буквами A-Z,AA,AB-ZZ,(для задания отобразить подходящие элементы формы, например 2 текстовых поля). Каждая ячейка таблицы отображает введенную в нее информацию. Редактирование информации осуществляется по клику на нее мыши. При клике мыши по ячейке под таблицей отображается строка ввода в которую можно ввести или какое либо значение или формулу. По нажатию клавиши OK значение вычисляется, если это формула и все зависимые ячейки пересчитываются