Решите следующие задачи и предоставьте решение в виде ссылок на [sqlfiddle.com](http://sqlfiddle.com/) и [codeshare.io](https://codeshare.io/). **Предоставление решения в правильном виде — один из критериев оценки.**

Задание №1

Существует школа, в которой есть учителя, ученики, и каждый учитель может обучать любое количество учеников, причём разные учителя могут обучать одних и тех же учеников (схема — ниже). Напишите следующие запросы:

* вывести список учителей и для каждого учителя в отдельной колонке список его учеников через запятую
* вывести любых двух учителей, у которых максимальное количество общих учеников

Схема:   
  
CREATE TABLE teacher (

id INT(10) PRIMARY KEY NOT NULL AUTO\_INCREMENT,

name VARCHAR(100) NOT NULL

);

CREATE TABLE student (

id INT(10) PRIMARY KEY NOT NULL AUTO\_INCREMENT,

name VARCHAR(100) NOT NULL

);

CREATE TABLE teacher\_student (

teacher\_id INT(10) NOT NULL,

student\_id INT(10) NOT NULL

);

INSERT INTO teacher (id, name) VALUES

(1, 'Маргарита Горшкова'),

(2, 'Иван Воронов'),

(3, 'Эмма Бобылева'),

(4, 'Римма Гурьева'),

(5, 'Дмитрий Смирнов');

INSERT INTO student (id, name) VALUES

(1, 'Федор Шашков'),

(2, 'Виктор Мельников'),

(3, 'Руслан Дементьев'),

(4, 'Роман Кузьмин'),

(5, 'Тимофей Овчинников');

INSERT INTO teacher\_student (teacher\_id, student\_id) VALUES

(1, 1), (1, 2),

(2, 3), (2, 4), (2, 5),

(3, 1), (3, 4), (3, 5),

(4, 1), (4, 2), (4, 4), (4, 5),

(5, 2), (5, 5);

Задание №2

Вас попросили исправить ошибку на сайте в коде счётчика посетителей. Код выглядит так:  
  
<?php file\_put\_contents("./counter.txt", file\_get\_contents("./counter.txt") + 1) ?>   
  
Что именно, как и почему вы будете исправлять?

Задание №3

На странице присутствуют упражнения, которые выполняет ученик. Упражнение это некий интерактивный элемент с правильными/неправильными ответами например:

* <select> в котором ученик должен выбрать правильный ответ.
* <input>, где ученик вписывает правильный ответ

Конкретная реализация не имеет значения (**писать упражнения не нужно**), достаточно просто знать что упражнения могут быть абсолютно разными.

Пример задания упражнений:

$exercises = [

**new** SelectExercise(**"select1", ["Apple", "Banana"]** */\* тут задаются свойства упражнения, этот момент можно опустить\*/*),

**new** InputExercise(**"input1"**)

];

Каждое упражнение должно уметь:

* сохранять ответы ученика в хранилище (например, базу данных)
* начислять или снимать баллы за правильный или неправильный ответ

На сервер приходят ответы от ученика, которые должны быть обработаны и отправлен результат: правильный ли ответ и сколько баллов начислено или снято.

Пример запроса:

[

{

**"exerciseId"**: **"select1"**,

**"answer"**: **0**

},

{

**"exerciseId"**: **"input1"**,

**"answer"**: **"despicable"**

}

]

Пример ответа:

[

{

**"isCorrect"**: **true**,

**"score"**: 1

},

{

**"isCorrect"**: **false**,

**"score"**: -0.5

}

]

**Нужно реализовать архитектуру классов для данной системы. Решение разместить на** [**https://codeshare.io/**](https://codeshare.io/)

**Важно учесть:**

1. Способ сохранения ответа в базу для всех упражнений сейчас один и тот же, но в будущем для **некоторых** упражнений способ сохранения может измениться и нужно, чтобы это можно было сделать легко.
2. Расчёт бонусных и штрафных баллов происходит сейчас для всех упражнений одинаково и зависит только от:
   * общего количества возможных ответов в конкретном упражнении;
   * максимального балла для упражнения.

Но для **некоторых** упражнений в будущем способ расчета может измениться.

Ради экономии времени очень важно писать как можно меньше кода: не делать реализацию сохранения, расчета баллов и так далее. Достаточно показать общую структуру классов, их методы, зависимости, так, чтобы было понятно, что происходит. Примерно так:

**class** Exercise {

**public $answers**;

**public function** saveToDatabase($newAnswer) {

*// сохраняем в базу*

}

}