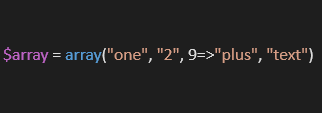
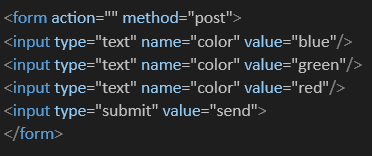
**Вопросы**

1. **Какой индекс будет у элемента "text" в следующем массиве?**



*Индексом элемента 'text' будет номер 10 ... потому как перед ним идёт элемент 9 ... который определяет счётчик ...*

1. **Какое значение будет в переменной $\_POST['color'] после отправки следующей формы?**



*в $\_POST['color'] будет передано последнее значение последнего элемента color = «red» .. ибо оно перебивает все предыдущие*

**3)Как правильно осуществить несколько запросов базе данных?**

mysqli\_query($q1); mysqli\_query($q2);

mysqli\_multi\_query($q1, $q2);

mysqli\_query($q1, $q2);

*Правильным вызовом будет вызов в первой строке (подряд два вызова mysqli\_query(); Выполнение остальных вариантов приведёт к ошибке..*

**Задача 1**

Создайте форму, состав и внешний вид полей которой показан на скриншоте. Скрипт, который будет обрабатывать данные формы на сервере, должен проверять заполнение полей имени и фамилии и в случае ошибки выводить предупреждение и прерывать скрипт. Что касается аватарки, то скрипт должен либо вывести аватарку после имени и фамилии, либо сообщить о незагруженной аватарке (см. скриншот задачи). Для получения имени, размера и временной папки для хранения загруженных файлов используйте суперглобальный массив $\_FILES. Для перемещения загруженной аватарки в целевую папку используйте функцию move\_uploaded\_file() (в качестве целевой создайте папку avatars). Чтобы правильно указать путь к вставляемой на страницу аватарке, не забудьте начать его с http://localhost/test/. При желании можете добавить проверку загружаемой аватарки на соответствие типу изображения и т. д.



**GIT**

1. **Как в Git установить значение «Имя пользователя»?**

git config --global user.name

git config user.name

git user.name

### **Возможен ли git pull, если в файле в рабочем каталоге есть незафиксированные изменения? Ответ нужно обосновать.**

### **Что делают команды «git reset –mixed» и «git merge –abort»?**

**Yii2**

**Задача 1**

Используя фреймворк Yii2 (basic):

1. Реализовать сущности авторы и книги

2. Реализовать административную часть

a. CRUD операции для авторов и книг

b. вывести список книг с обязательным указанием имени автора в списке

c. вывести список авторов с указанием кол-ва книг

3. Реализовать публичную часть сайта с отображение авторов и их книг (простой вывод списка на странице)

4. Реализовать выдачу данных в формате json по RESTful протоколу отдельным контроллером

a. GET /api/v1/books/list получение списка книг с именем автора

b. GET /api/v1/books/by-id получение данных книги по id

c. POST /api/v1/books/update обновление данных книги

d. DELETE /api/v1/books/id удаление записи книги из бд

Результат представить ссылкой на репозиторий.

Желательно, в репозиторий залить пустой каркас приложения, а затем с внесёнными изменениями, чтобы можно было проследить diff.

**Задача 2**

Методом *login* использует фреймворк и базу данных осуществите валидацию и вход на сайт.

**Задача 2.1**

Проведите юнит тестирование написанного года.