# Домашнее задание #5

1. Спросите **имя** пользователя с помощью формы. Результат запишите в переменную **$name**. Выведите на экран фразу **'Привет, %Имя%'**
2. Спросите у пользователя **имя**, **возраст**, а также попросите его ввести **сообщение** (его сделайте в **textarea**). Выведите эти данные на экран в **формате**, приведенном под данной задачей. Позаботьтесь о том, чтобы пользователь **не мог вводить теги** (просто удаляйте их) и таким образом сломать сайт.

Формат:

Привет, Дмитрий, 25 лет.  
Твое сообщение: ...

1. Спросите **возраст** пользователя. Если форма была **отправлена** и **введен** возраст, то выведите его на экран, а форму уберите. Если же форма не была отправлена (это будет при **первом** заходе на страницу) - просто покажите ее.
2. Спросите у пользователя **имя**, а также попросите его ввести сообщение (**textarea**). Сделайте так, чтобы после отправки формы значения его полей не пропадали.
3. Пользователь приходит на сайт. Используя cookie сделать так, что б впервые пришедший пользователь видел фразу: "Добро пожаловать, новый пользователь!" Если пользователь уже посещал сайт в течении последних 2-х часов, выводить фразу: "С возвращением, друг!"
4. Используя cookie реализовать вывод на страницу сообщения с датой и временем последнего визита.
5. Используя cookie реализовать вывод на страницу сообщения с количеством посещений страницы.
6. Представим себе сайт с прохождением теста: на каждой странице находится вопрос, варианты ответа и кнопка “Далее”. Тест содержит некоторое количество страниц (допустим 3). На последней, 4-й странице пользователь должен получить результат теста. Задача – создать подобный тест.

Подсказки:

• Для создания страниц вопросов используйте формы (с элементами type=”radio”), которые будут отправлять результаты на следующую страницу с вопросом или результатами.

• Ответы пользователя на вопросы на предыдущих страницах можно (и нужно) сохранять в сессии

• Для расчета результатов теста сравните ответы пользователя (хранятся в сессии) с заранее определенными правильными ответами (могут храниться, например, в массиве или переменных).

• После окончания текста сессию следует очистить

1. Для класса User cделайте метод **compare**, который параметром будет принимать два объекта и возвращать **true**, если они имеют одинаковые свойства и значения являются экземплярами одного и того же класса, и **false**, если это не так.

И сделайте метод **compare2**, который параметром будет принимать два объекта и возвращать **true**, если переданные переменные ссылаются на один и тот же объект, и **false**, если на разные.

class User{

private $name;

private $age;

public function \_\_construct($name, $age){

$this->name = $name;

$this->age = $age;

}

public function getName(){

return $this->name;

}

public function getAge(){

return $this->age;

}

}

**Помощь**:

Для очистки текста от тегов, можно воспользоваться функцией strip\_tags