## Лабораторная работа 3 - PHP

[**PHP Code Style**](https://docs.google.com/document/d/1Rfb0ol6OQZygOqNmT2Zc3oaDHCA04iVLs-fkX9Qy7TE/edit)

**[PHP Code Style](https://docs.google.com/document/d/1Rfb0ol6OQZygOqNmT2Zc3oaDHCA04iVLs-fkX9Qy7TE/edit)**

При выполнении этой лабораторной во всех файлах перед выводом стоит указывать

header("Content-Type: text/plain");

0) Создание git репозитория:

* Зарегистрироваться на сайте github.com
* Создать репозиторий web-development
* Закоммитить исходный код первой и второй лабораторной в папке lw1 и lw2 соответственно

В результате структура файлов в репозитории должна быть следующая:

web-development/

web-development/lw1/index.php

web-development/lw2/task1.pas

web-development/lw2/task2.pas

web-development/lw2/task3.pas

web-development/lw2/task4.pas

Все последующие задания нужно выкладывать в git репозиторий, каждая лабораторная в отдельной папке и каждое задание в отдельной папке.

1) Разработайте приложение Remove Extra Blanks на языке PHP. В запросе GET передается параметр text. Скрипт выводит в стандартный поток вывода этот же текст без пробелов в начале и в конце, между словами один пробел.

2) Разработайте PHP приложение Check Identifier, которое проверяет является ли переданный в GET параметр identifier идентификатором по правилу SR3. Программа должна выводить yes или no, а также поясняющую информацию в том случае, если переданная строка не является идентификатором.

<идентификатор> ::= <буква>

| <идентификатор><буква>

| <идентификатор><цифра>

3) Разработайте программу для проверки надежности пароля Password Strength. В GET параметре password передается пароль для анализа. Пароль может состоять только из английских символов в верхнем и нижнем регистрах, а также из цифр.

Надежность пароля вычисляется по следующему принципу, (len это длинна пароля)

* Изначально считаем надежность равной 0.
* К надежности прибавляется (4\*n), где n - количество всех символов пароля
* К надежности прибавляется +(n\*4), где n - количество цифр в пароле
* К надежности прибавляется +((len-n)\*2) в случае, если пароль содержит n символов в верхнем регистре
* К надежности прибавляется +((len-n)\*2) в случае, если пароль содержит n символов в нижнем регистре
* Если пароль состоит только из букв вычитаем число равное количеству символов.
* Если пароль состоит только из цифр вычитаем число равное количеству символов.
* За каждый повторяющийся символ в пароле вычитается количество повторяющихся символов

Например: abcd1a, вычитаем -2 поскольку символ a встречается дважды. Программа должна выводить на экран надежность пароля в виде числа.

4) Разработайте PHP приложение Survey Saver, которое сохраняет анкеты пользователей в файловой системе. Данные передаются в строке запроса. Возможные параметры запроса: first\_name, last\_name, email, age. Все файлы необходимо сохранять в директорию data. Название файла: <email>.txt Некоторые параметры могут отсутствовать, параметр email обязательный. В случае если такой файл <email>.txt уже существует, данные в этом файле обновляются.

5) Разработайте PHP приложение Survey Info, которое выводит данные анкеты пользователя, сохраненные в задании 4. Если значение отсутствует, выводить вместо него пробел.

Выводить данные необходимо в виде:

First Name: <name>

Last Name: ...

Email: <email>

Age: 30

В GET параметрах запроса передается email в виде ?email=ivan@mail.com.