Федеральное государственное бюджетное образовательное учреждение высшего образования   
«Московский технологический университет» 

**курсовая работа по дисциплине**

**«Управление и обработка данных»**

Выполнил:   
студент 1 курса   
дневного отделения   
группа БСБО-10-17   
шифр: 17Б0161

Вариант 1   
Рябов К.А.   
  
Научный руководитель:

------------------

Москва  
 2018

**Содержание:**

* Задача
* Основы PHP и сохранения данных в файл

1. **Задание**

## Цель задания

1. Ознакомление с возможностями организации файлов на внешних носителях в используемой ЭВМ.

2. Получение практических навыков работы с внешними файлами.

## 

## Постановка задачи

Подготовить данные об абитуриентах, поступающих в техникум. Информацию о каждом абитуриенте оформить в виде записи, содержащей следующие поля:

1. Фамилия, имя, отчество.

2. Год рождения.

3. Год окончания школы.

4. Оценки в аттестате с указанием предметов.

5. Признак - нуждается ли в общежитии.

6. Оценки вступительных экзаменов.

## Указания

1. Для заполнения данных по каждому студенту разработать 2 формы. В первой форме вводить общие данные: фамилия, имя, отчество; год рождения; год окончания школы; признак - нуждается ли в общежитии.
2. Во второй форме вводить оценки в аттестате с указанием предметов и оценки вступительных экзаменов с указанием предметов.

## Задание

Разработать программу записи подготовленных данных во внешний файл и программу обработки созданного внешнего файла.

I. Удалить из внешнего файла все записи абитуриентов, получивших хотя бы одну оценку 2, и распечатать информацию, оставшуюся в файле.

II. Добавить 2 записи в начало (конец) внешнего файла и распечатать записи полученного файла список абитуриентов, имеющих в аттестате оценки только 5.

1. **Основы работы с PHP**
   1. **Что такое PHP?**

**PHP –** это широко используемы сценарный язык с открытым кодом. **PHP** скрипты выполняются на сервере.

Аббревиатура **PHP** расшифровывается как «**Hypertext Preprocessor**» (Препроцессор Гипертекста).

Кратко о возможностях **PHP**:

* Создание динамического контента страницы
* Создание, открытие, чтение, запись, удаление и закрытие файлов на сервере
* Формы сбора данных
* Добавление, удаление и модифицирование информации, хранящейся в базе данных
* Управление пользовательским доступом
* Шифрование данных

…

* 1. **Базовый Синтаксис**

Для того, чтобы писать на PHP, нужно прописывать следующие теги, внутри которых сам код.

<?php

// код PHP

?>

(наиболее рекомендованный)

Или

<script language=”php”>

// код PHP

</script>

Или

<?

// код PHP

?>

**PHP** имеет встроенную функцию **echo**, которая используется для вывода.

* 1. **Переменные**

Переменные используются в качестве именованной области памяти, в которой можно хранить различную информацию (типы данных, поддерживаемые языком **PHP**: **String**, **Integer**, **Float**, **Boolean**, **Array**, **Object**, **NULL**, **Resource**). Для их обозначения в самое начало прописывается символ «$», после которого следует имя переменной.

Правила для **PHP** переменных:

- Имя переменной должно начинаться с буквы или с нижнего подчеркивания

- Имя переменной не может начинаться с числа

- Имя переменной может содержать только буквенно-цифровые символы и нижние подчеркивания

- Имена переменных чувствительны к регистру

**PHP** не имеет особой команды для объявления переменной. Она создается в момент первого присваивания ей значения.

**PHP** переменные могут быть объявлены в любом месте скрипта.

Область видимости переменной является часть скрипта, в которой ссылается на переменную, или ее использовать. Области видимости могут быть локальными или глобальными.

(Например, переменная, объявленная вне функции имеет глобальную область видимости, а переменная, объявленная внутри – локальную область видимости).

* 1. **Операторы**

1. Арифметические операторы работают с числовыми значениями для выполнения стандартных арифметических операций.

|  |  |  |
| --- | --- | --- |
| Оператор | Название | Пример |
| + | Сложение | $x + $y |
| - | Вычитание | $x - $y |
| \* | Умножение | $x \* $y |
| / | Деление | $x / $y |
| % | Деление по модулю | $x % $y |

1. Операторы присваивания могут быть использованы вместе с арифметическими операторами.

$x = $y

|  |  |  |
| --- | --- | --- |
| Присваивание | Аналог | Пример |
| $x+=$y | $x = $x + $y | Сложение |
| $x-=$y | $x = $x + $y | Вычитание |
| $x\*=$y | $x = $x + $y | Умножение |
| $x/=$y | $x = $x + $y | Деление |
| $x%=$y | $x = $x + $y | Деление по модулю |

1. Операторы сравнения сравнивают два значения (числа или строки).

|  |  |  |  |
| --- | --- | --- | --- |
| Оператор | Название | Пример | Результат |
| == | Равно | $x == $y | Возвращает **true**, если $x равно $y |
| === | Идентично | $x === $y | Возвращает **true**, если $x равно $y, и они одного типа |
| != | Не равно | $x != $y | Возвращает **true**, если $x не равно $y |
| <> | Не равно | $x <> $y | Возвращает **true**, если $x не равно $y |
| !== | Не идентично | $x !== $y | Возвращает **true**, если $x не равно $y, или не одного типа |

1. Логические операторы используются для комбинаций условных выражений.

|  |  |  |  |
| --- | --- | --- | --- |
| Оператор | Название | Пример | Результат |
| and  ( && ) | И | $x and $y  ($x && $y) | **true**, если и $x и $y истины |
| or  ( || ) | Или | $x or $y  ($x || $y) | **true**, если либо $x либо $y истины |
| xor | Исключающее или | $x xor $y | **true**, если либо $x либо $y истины, но не оба вместе |
| ! | Не | !$x | **true**, если $x не истинно |

* 1. **Массивы**

Массив – это специальная переменная, которая может хранить в себе больше одного значения.

Различают индексированные и ассоциативные массивы:

1. Индексированные массивы может быть инициализирован следующим способом:

$num = array(“one”, “two”, “three”);

Или

$num[0] = “one”;

$num[1] = “two”;

$num[2] = “three”;

1. Ассоциативные массивы – это массивы, которые используют именованные ключи.

$people = array(“Ilia”=>”34”, “Victor”=>”65”, “Maxim”=>”12”);

Или

$people[‘Ilia’] = “34”;

$people[‘Victor’] = “65”;

$people[‘Maxim’] = “12”;

* 1. **Управляющие структуры**