

## 2- Переменные и типы данных в PHP

Переменные невероятно важны, так как позволяют хранить информацию и использовать её в дальнейшем. Вначале может быть не совсем понятно зачем вообще что-то записывать в переменную, если можно просто оперировать значениями без них. Понимание переменных придет немного позже, когда мы начнем создавать более сложные программы и нам потребуется хранить информацию в каком-либо месте.

Типы переменных в языке **PHP** не объявляются очевидно, тем не менее они присутствуют. Интерпретатор понимает что записывается в переменную и на основании этого добавляет тип к этой переменной.

В ходе программы мы можем перезаписывать значение переменной, при этом мы можем указывать новый тип переменной.

К примеру, изначально был записан тип **float** (число с точкой), но потом можно записать другой тип данных:

```
$first_num = 23.2; // Тип данных float
```

```
$first_num = "1"; // Тип данных string
```

Ниже приводим несколько существующих типов данных в языке PHP:

- `$some = 1` Integer - целые числа;
- `$some = 1.12` Float - числа с плавающей точкой;
- `$some = "Привет"` String - строки;
- `$some = true` Boolean - принимают только true или же false.

### Создание константы

В **PHP** помимо создания простых переменных можно создавать константы.

Константы – это значения, что нельзя изменить входе выполнения программы. Создав один раз константу и присвоив ей значение, мы не сможем больше изменить это значение, так как будет выдаваться ошибка.

Чтобы создать константу необходимо использовать метод `define()`:

```
define("SOME", "Значение");
```

```
echo SOME; // Вывод константы
```

Первым параметром указывается название константы, а вторым параметром - значение. Принято записывать названия констант в верхнем регистре, хотя при написании в нижнем регистре ошибок выдано не будет.

Вы можете проверить существует ли константа используя метод `defined()`.

### Исходной код: Выполнить пример

```
<!doctype html>

<html lang="ru">

<head>

    <meta charset="UTF-8">

    <meta name="viewport"

        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0,
minimum-scale=1.0">

    <meta http-equiv="X-UA-Compatible" content="ie=edge">

    <title>PHP</title>

</head>

<body>

<?php

    define(MY_AGE, 24);

    echo MY_AGE . '<br>';

    $number = 5; // int

    // $number = 45;

    $num = -0.55; // float

    $str = "Переменная"; // string

    $bool = false; // boolean

    $a = 0.5;

    $b = "0.5";

    echo $a + floatval($b);

    echo $str . ': ' . $number . '. Var 2: ' . $num;

?>

</body>

</html>
```

## Самостоятельная работа

### Задание 1. Работа с переменными

Создайте переменные для хранения:

- имени и фамилии;
- даты рождения;
- возраста;
- уровня радости от 0 до 1.

После установки значений выведите все переменные на экран.

Для имени и фамилии необходимо использовать строковый тип данных:

```
$name = "Павел";
```

```
$surname = "Иванов";
```

Чтобы вывести дату рождения вы можете использовать строковую переменную или три числовых переменных:

```
$date = 1;
```

```
$month = 11;
```

```
$year = 1999;
```

Количество лет и уровень радости можно записать следующим образом:

```
$age = 20;
```

```
$happy = 0.76;
```

Теперь осталось лишь вывести все данные:

```
echo "Имя: $name, фамилия: $surname";
```

```
echo "Дата рождения: $date $month $year";
```

```
echo "Возраст: $age";
```

```
echo "Уровень счастья: $happy»;
```

### Задание 2. Необычные переменные

Создайте переменные со значениями: false, -0.78902.

Выведите все значения в консоль.

### **Задание 3. Простая константа**

Создайте константу с название NUMBER\_E и со значением 2,71828. Выведите константу на экран.

### **Задание 4. Проверка данных**

Проверьте существует ли константа NUMBER\_E. Результат проверки выведите на экран.