Práctica N.º 4

PHP: VARIABLES, TIPOS, OPERADORES, EXPRESIONES, ESTRUCTURAS DE CONTROL

Ejercicio 1

- Variables:
  - **a:** boolean.
  - **b:** string.
  - **c:** string.
  - **d:** integer.
  - **f:** integer.
  - **g:** integer.
- Operadores:
  - Unarios:
    - ++.
  - Binarios:
    - *.
    - +=.
    - =.
  - Ternarios:
    - ?:.
- Funciones:
  - doble($i).
  - gettype($var).
  - is_int($var).
  - is_string($var).
- Estructuras de Control:
  - if(){}.
- Salida por pantalla:
  - Booleanstringstringinteger184444

Ejercicio 2

a) Son equivalentes.
b) Son equivalentes.
c) Son equivalentes

Ejercicio 3

a) Lo que hace el código es crear automáticamente una tabla de 5 filas y 2 columnas, dentro de las celdas solo inserta un espacio en blanco. El ancho de la tabla ocupa un 90% de la pantalla y tiene un borde de 1px.
b) El código verifica si el input *submit* tiene un valor definido, en caso de que no lo tenga muestra un formulario para ingresarlo, si esta ingresado controla si la edad corresponde a un mayor de 21 y muestra el mensaje *Mayor de edad*, de lo contrario muestra *Menor de edad.*

Ejercicio 4

La primer salida del código es un Warning de que ni la variable *flor* ni *color* se encuentran y muestra el texto sin las mismas. Luego del include el archivo ya se

encuentra por lo tanto las variables si esta disponibles y muestra el mensaje *El clavel blanco /n*. El */n* se muestra ya que la forma correcta de insertar un salto de línea sería

Ejercicio 5

En el caso del ejercicio la ejecución no funcionaba hasta añadir *php* luego de *<?*, una vez añadido y ejecutando, la pagina visitas invoca a la pagina contador, la cual abre el archivo contador.dat y lee el valor de la cantidad de visitas que se encuentra en el mismo.

Luego de almacenarlo, lo cierra y vuelve a abrirlo nuevamente pero con permisos de escritura, donde aumenta en uno el valor y escribe el archivo y muestra la cantidad de visitas aumentada.

PHP: ARRAYS, FUNCIONES

Ejercicio 1

Ambos códigos son equivalentes, los dos crean un array asociativos con las mismas claves y valores.

Ejercicio 2

a) Devuelve el valor *bar1*, ya que el valor *true* se muestra como 1.
b) Muestra *5942*.
c) No tiene salidas ya que no hay ninguna sentencia *echo* o *print*.

Ejercicio 3

a) Devuelve el texto escrito con la hora, minuto, segundo, día, mes y año de cuando se ingresó a la página.
b) Devuelve *5+6=11*.

Ejercicio 4

Controla y notifica que el usuario enviado por parámetro posea solo caracteres alfanuméricos, - o _ y también que su longitud sea de entre 3 y 20 caracteres. De ser válido muestra el usuario seguido de la cadena *es válido*. De lo contrario, muestra el usuario y la cadena *no es válido*.

Un script para probarlo sería el siguiente, dentro del archivo *"control.php"* se encuentra la función presentada en el enunciado.

```
<!—Página que verifica el usuario -->
<html>
<head></head>
<body>
<?php
    include './control.php';
    if (!isset($_POST['submit'])) {
    ?>
<form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
    Usuario: <input type="text" name="user">
    <input type="submit" name="submit" value="Registrar">
</form>
<?php
    }
    else {
```

```
        $usuario = $_POST['user'];
        comprobar_nombre_usuario($usuario);
    }
?>
</body>
</html>
```