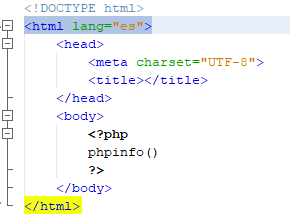
***HOJA DE EJERCICIOS 2***

*Ejer1*

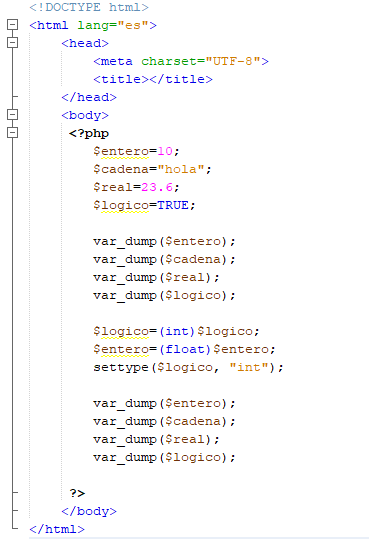
Indicar la salida que se produce con el siguiente código php

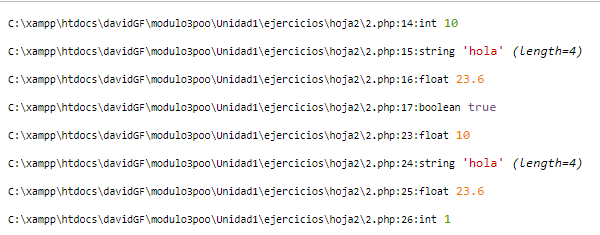


La salida que produce es la información y características del servidor en el cual se encuentra alojada la página php

*Ejer2*

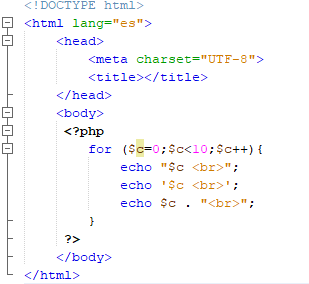
Indicar la salida que se produce con el siguiente código php



La salida que produce es la información de las diferentes variables recibiendo algunos cambios en el segundo var\_dump lógico se cambia de booleano True a un valor numérico integro, Entero se cambia de numérico integro a numérico flotante, la cadena y en real se quedan igual

*Ejer3*

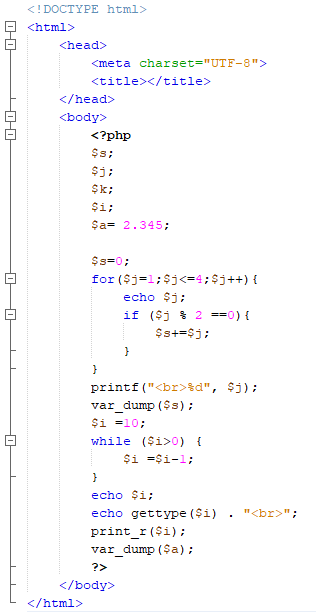
Indicar la salida que se produce con el siguiente código php



La salida que produce es una “lista” progresiva hasta llegar a 10 el cual no realiza en la cual, el primer echo y el tercero muestran el valor de la variable c mientras que el segundo echo muestra $c todo el tiempo.

*Ejer4*

Indicar la salida que se produce con el siguiente código php



Para empezar, muestra una sucesión hecha por el bucle for cuyo resultado es 1234, después de un salto de línea imprime el ultimo valor de $s (4) incrementado en 1 (5) que se almacena en $j luego un var\_dump de $s, luego del While imprime $i el cual vale 0 (que es donde se quería llegar) y el tipo de dato que es, después de un salto de línea vuelve a imprimir $i, finalizando con un var\_dump de $a

*Ejer5*

Indicar la salida que se produce con el siguiente código php



Con los Echo fuera de los <div> me imprime los nombres de los Alumnos, pero Ramón y José están unidos, mientras que los Alumnos Pepe y Ana están con saltos de Línea, en el Echo del <div> el cual se inicia después de la Alumna Ana se vuelven a imprimir cada uno con su salto de línea correspondiente.

*Ejer6*

Modificar el ejercicio anterior y utilizar un Array en vez de utilizar variables de tipo cadena sueltas. Para imprimir utilizar un bucle

Después de la modificación queda de esta manera, el resultado en este caso es el nombre de todos los alumnos metidos en el Array con un salto de línea

*Ejer7*

Indicar la salida que se produce con el siguiente código php



me imprime los nombres de los Alumnos cada uno con su salto de línea correspondiente de manera repetida, el primero usa la variable $c (la cual va aumentándose con $c++ en cada vuelta) para ir “contando” los valores del Array, mientas que en el segundo lo va imprimiendo a medida que se repite el bucle

*Ejer8*

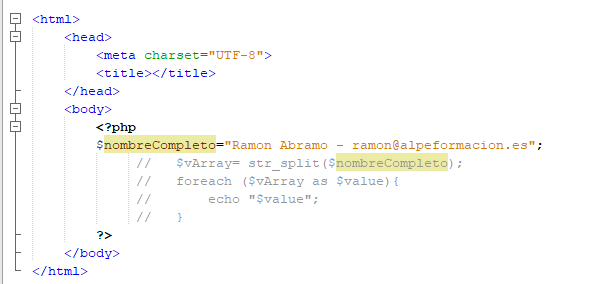
Indicar que salida se produce con el siguiente código php



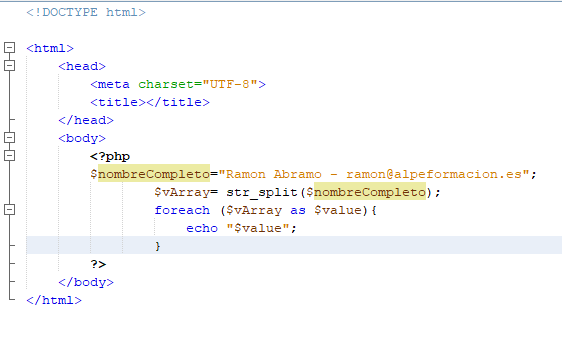
Me muestra $v5 de manera literal y con un escape me muestre la posición que ocupa cada valor y acto seguido me muestra el valor que tiene cada posición

*Ejer9*

Completar el siguiente programa para que imprima cada uno de los caracteres de la variable “nombreCompleto” en una línea para ello utilizar la variable como si fuera un array (aunque sea un string)



Antes de la modificación se ve sin lo que esta comentado, después de la modificación se vería de la siguiente manera, al descomentar esas líneas lo que se logra es que interprete nombre completo como un array el cual con el ForEach se imprimen todos los valores de manera seguida



*Ejer10*

Indicar la salida que se produce con el siguiente código php.



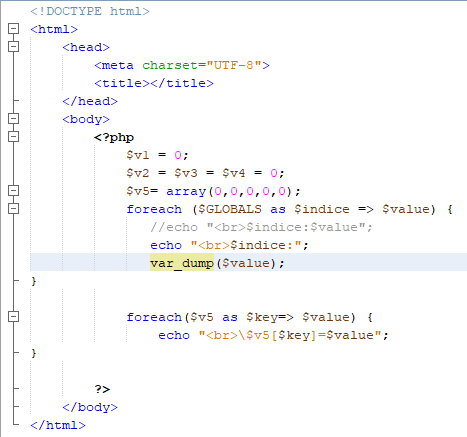
En el primer bucle es usado el for para ver la “longitud” del valor de $cadena y va sumando 1 al contador después de imprimir el echo hasta terminar la “longitud” de $cadena mostrándolos cada pieza de $cadena por separado. Durante el segundo Bloque cuyo resultado es el mismo trata el String de $cadena como si fuera un Array.

*Ejer11*

Indicar la Salida que se produce con el siguiente código php.



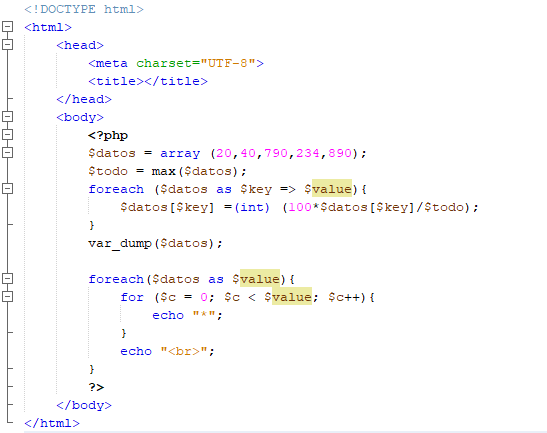
Con el código de esta manera nos casca un “fallo” debido a que estamos intentando imprimir un Array



Pero al cambiarlo al siguiente código ya no nos casca el “fallo” debido a que $GLOBALS es un Array con múltiples Arrays dentro, los cuales se deben recorrer con el var\_dump, el segundo Foreach desde el principio imprime los valores del array $v5

*Ejer12*

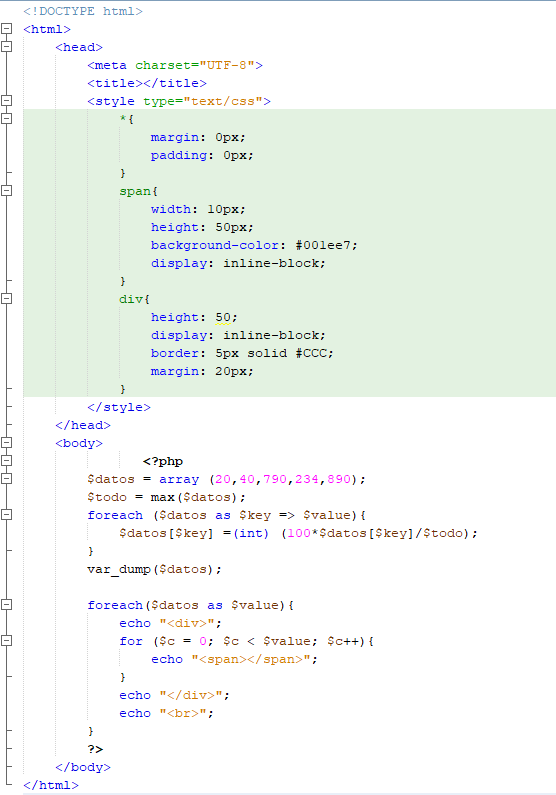
Indicar la salida que se produce con el siguiente código php



Con el primer Foreach se calcula los nuevos valores del Array $datos, luego con un Var\_dump nos muestra esos valores calculados, en el segundo Foreach mediante un bucle for el cual imprime el siguiente signo \* nos muestra el resultado almacenado en $datos

*Ejer13*

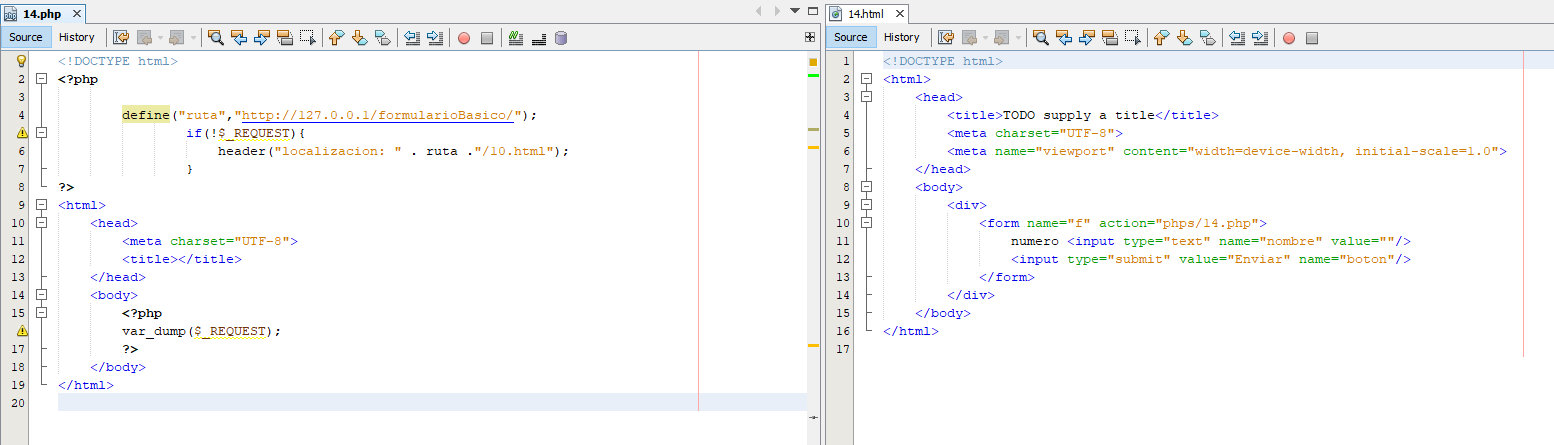
Indicar la salida que se produce con el siguiente código php



El resultado es prácticamente igual al ejercicio anterior con la única diferencia de que en vez del signo \* se llenan unas barras de color azul

*Ejer14*

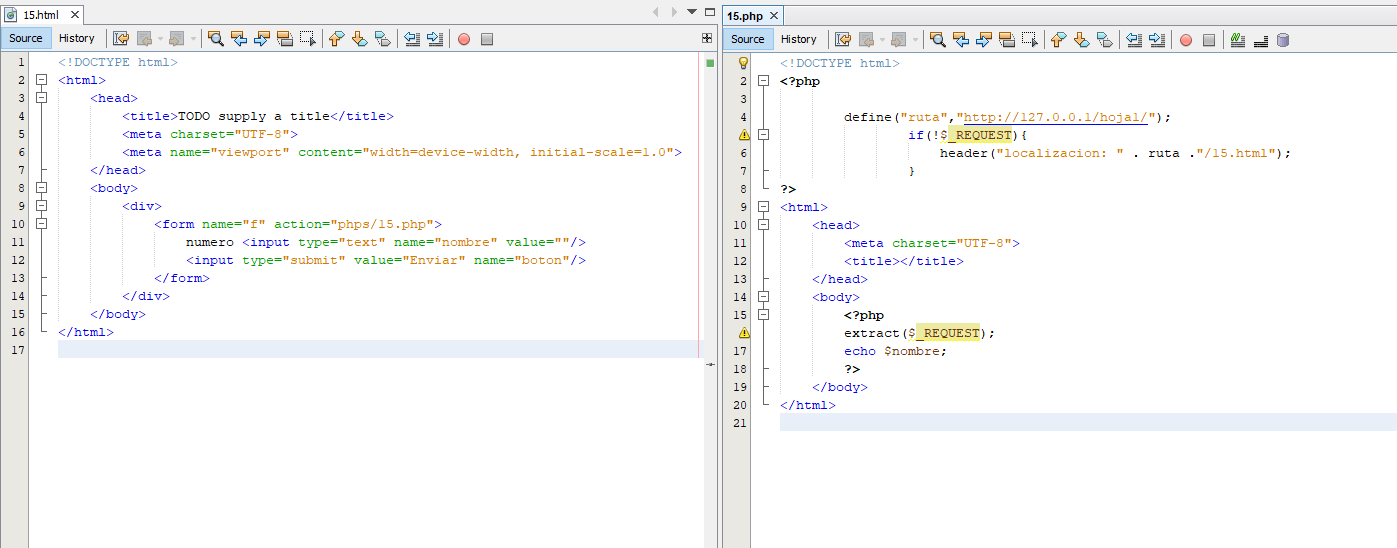
Indicar la salida que se produce con el siguiente código php



Con el HTML que acompaña al php hacemos un “formulario” con el cual al introducirle un dato nos muestra información de ese dato con el Var\_dump, el define hace que se “cree” una ruta de la cual se tira en el Header que hay en el bucle if

*Ejer15*

Indicar la salida que se produce con el siguiente código php



Al igual que en el ejercicio anterior con el HTML que acompaña al php hacemos un “formulario” con el cual al introducirle un dato nos lo muestra en el echo del Php sacada esa información con el Extract, el define hace que se “cree” una ruta de la cual se tira en el Header que hay en el bucle if

*Ejer16*

Indicar la salida que se produce con el siguiente código php



Nos genera un mini “formulario” en el cual al poner un dato y dar a enviar nos devuelve un var\_dump mostrando la información del dato introducido y el botón para enviar el dato, si se diera el caso que fuera una variable vacía repetiría la página del mini “formulario”

*Ejer17*

Indicar la salida que se produce con el siguiente código php



El código nos muestra lo mismo que el ejercicio anterior con la única diferencia de que en este caso única y exclusivamente permite la inserción de números y códigos alfanuméricos que empiecen por números, ya que en la parte donde se comprueba (el php antes de iniciar el HTML) comprueba que sea una variable de tipo numero integro superior a 0, si eso no se cumple nos obliga a meter otro valor a la variable

*Ejer18*

Indicar la salida que se produce con el siguiente código php



El código nos muestra lo mismo que el ejercicio anterior con la única diferencia de que en este caso única y exclusivamente permite la inserción de números y códigos alfanuméricos que empiecen por números, ya que en la parte donde se comprueba (el php antes de iniciar el HTML) comprueba que sea una variable de tipo número superior a 0, si eso no se cumple nos obliga a meter otro valor a la variable

*Ejer19*

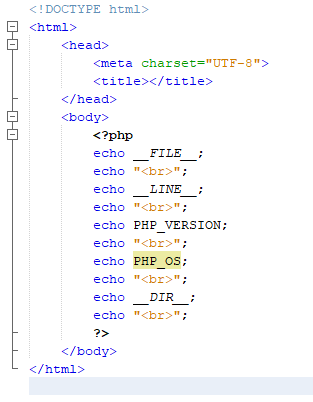
Indicar la salida que se produce con el siguiente código php



De nuevo hace lo mismo que el ejercicio anterior con ciertas diferencias, las cuales están en que esta vez pide dos números y que tiene una hoja de estilo interna la cual hace que el formulario quede con mejor aspecto

*Ejer20*

Indicar la salida que se produce con el siguiente código php



Nos muestra los detalles de:

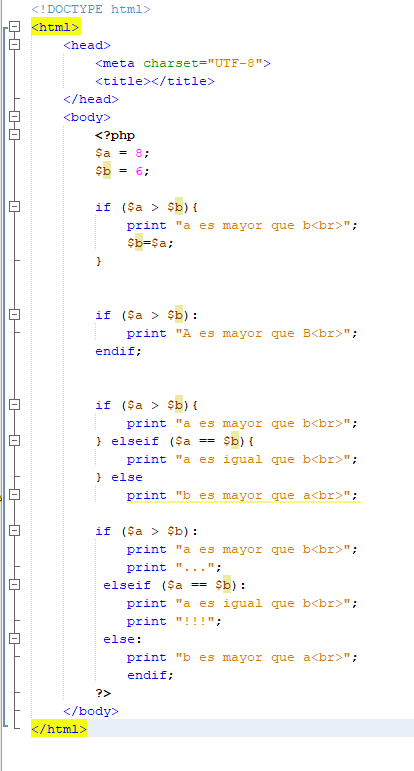
\_\_FILE\_\_: ruta dentro del ordenador donde se encuentra el archivo, incluyendo al archivo

\_\_LINE\_\_: número de línea donde se encuentra esta variable

PHP\_VERSION: la versión de PHP de nuestro Ordenador  
PHP\_OS: nuestro sistema operativo  
\_\_DIR\_\_: ruta dentro del ordenador donde se encuentra el archivo parando en la carpeta contenedora del archivo

*Ejer21*

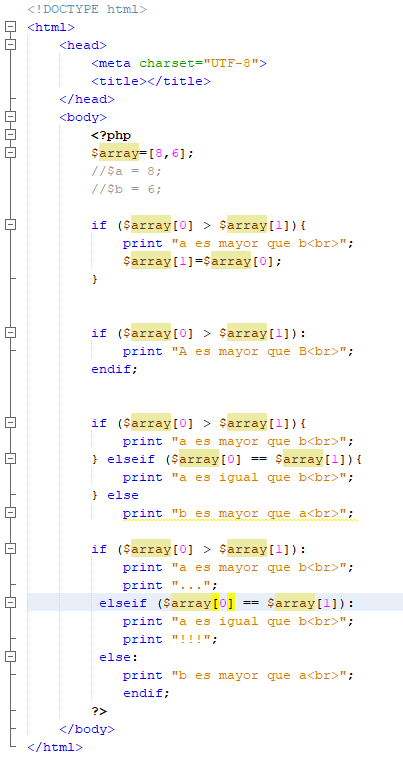
Indicar la salida que se produce con el siguiente código php



Se comprueban las variables que al cumplir la condición inicial muestran el primer If e igualan los valores de las variables, entonces realiza una comprobación que no da resultado y empieza a comprobar de nuevo si se cumplen condiciones de dos bucles if encontrando dos veces un resultado que nos muestra

*Ejer22*

Modificar el ejercicio anterior utilizando un array de dos elementos en vez de dos Variables enteras



Al ser una modificación del ejercicio anterior lo que nos imprime es lo mismo la única gran diferencia es la utilización de un Array en el cual metemos los valores de lo que anteriormente eran nuestras dos variables