

Indicar si los siguientes códigos son equivalentes.

a_i.php

```
1 <?php
2 $i = 1;
3 while ($i <= 10) {
4     print $i++;
5 }
6 ?>
```

a_ii.php

```
1 <?php
2 $i = 1;
3 while ($i <= 10):
4     print $i;
5     $i++;
6 endwhile;
7 ?>
```

a_iii.php

```
1 <?php
2 $i = 0;
3 do {
4     print ++$i;
5 } while($i < 10);
6 ?>
```

Estos tres códigos son equivalentes entre sí en cuando a la salida visual.

El primero inicializa la variable \$i en 1 y en cada iteración del bucle while va mostrando su valor y luego incrementándolo, generando la salida “12345678910”.

El segundo cuenta con la misma inicialización que el primero, pero cuenta con una sintaxis alternativa para la estructura de control while que también es válida. La salida generada es “12345678910”.

El tercero inicializa la variable \$i en 0, y dentro del bucle while incrementa la variable antes de realizar el print, por lo que de igual forma genera la salida “12345678910”.

Tienen una diferencia interna en el valor final de la variable \$i. Siendo igual a 11 en los dos primeros ejemplos, e igual a 10 en el tercer código.

b_i.php

```
1 <?php
2 for ($i = 1; $i <= 10; $i++) {
3     print $i;
4 }
5 ?>
```

b_ii.php

```
1 <?php
2 for ($i = 1; $i <= 10; print $i, $i++) ;
3 ?>
```

b_iii.php

```
1 <?php
2 for ($i = 1; ;$i++) {
3     if ($i > 10) {
4         break;
5     }
6     print $i;
7 }
8 ?>
```

b_iv.php

```
1 <?php
2 $i = 1;
3 for (;;) {
4     if ($i > 10) {
5         break;
6     }
7     print $i;
8     $i++;
9 }
10 ?>
```

Estos cuatro códigos también son equivalentes entre sí.

El primero se trata de un bucle for común que inicializa una variable y muestra su valor mientras lo va incrementando, mostrando en la pantalla

El segundo en vez de realizar el print dentro del cuerpo del bucle, lo hace dentro de la expresión que se ejecuta al finalizar cada iteración.

El tercero y el cuarto, al no tener la expresión de condición, actúan como un bucle infinito, y se detendrán cuando se cumpla la condición de que \$i sea mayor a 10 en ambos casos.

c_i.php

```
1 <?php
  ...
  ...
2 if ($i == 0) {
3     print "i equals 0";
4 } elseif ($i == 1) {
5     print "i equals 1";
6 } elseif ($i == 2) {
7     print "i equals 2";
8 }
9 ?>
```

c_ii.php

```
1 <?php
  ...
  ...
2 switch ($i) {
3     case 0:
4         print "i equals 0";
5         break;
6     case 1:
7         print "i equals 1";
8         break;
9     case 2:
10        print "i equals 2";
11        break;
12 }
13 ?>
```

Por último, estos dos códigos también son equivalentes.

Mientras que el primero se trata de múltiples ifs anidados que indican si la variable \$i es 0, 1 o 2, el segundo realiza la misma tarea pero a través de la estructura de control switch.