

## For

Los ciclos **for** son los más complejos en PHP. Se comportan como sus homólogos en el lenguaje de programación C. La sintaxis general de un ciclo **for** es:

```
for (expr1; expr2; expr3)  
    sentencia
```

La primera expresión (*expr1*) es evaluada (ejecutada) una vez incondicionalmente al comienzo del bucle. *Invariablemente es la sentencia donde se inicializa la variable de control del ciclo.*

En el comienzo de cada iteración, se evalúa *expr2*. Si se evalúa como **TRUE**, el bucle continúa y se ejecutan la/sy sentencia/s anidada/s. Si se evalúa como **FALSE**, finaliza la ejecución del bucle. *Las sentencias contenidas en el cuerpo del ciclo se ejecutan si se cumple la condición de expr2.*

Al final de cada iteración, se evalúa (ejecuta) *expr3*. *Regularmente es la instrucción de incremento de la variable de control.*

Ejercicio:

Asignar a una variable un valor numérico entre el 1 y el 9. Mediante un ciclo realizar el código para imprimir un triángulo como el siguiente:

```
1  
2 2  
3 3 3  
4 4 4 4  
....
```

## While

Los ciclos *while* son el tipo más sencillo de ciclos en PHP. La forma básica de una sentencia *while* es:

```
while (expr)  
    sentencia
```

El significado de una sentencia *while* es simple. Le dice a PHP que ejecute las sentencias anidadas, tanto como la expresión *while* se evalúe como **TRUE**. El valor de la expresión es verificado cada vez al inicio del bucle, por lo que incluso si este valor cambia durante la ejecución de las sentencias anidadas, la ejecución no se detendrá hasta el final de la iteración (cada vez que PHP ejecuta las sentencias contenidas en el bucle es una iteración). A veces, si la expresión *while* se evalúa como **FALSE** desde el principio, las sentencias anidadas no se ejecutarán ni siquiera una vez.

Ejemplo:

Imprimir los números pares entre el número 100 y el número 200.