

**while loop:**

A while loop is a control flow statement that allows code to be executed repeatedly based on a given Boolean condition. The while loop can be thought of as a repeating if statement. (while 迴圈是一種控制流程語句，它允許程式碼根據給定的布林條件反覆執行。while 迴圈可以被視為重複執行的 if 語句。)

**Syntax :**

```
while (boolean condition)
{
    loop statements...
}
```

**Example:**

```
class WhileLoop
{
    public static void main(String[] args)
    {
        int i = 1 ;

        while (i <= 10 )
        {
            System.out.println( "value of i :" + i);
            i++;
        }
    }
}
```

---

**for loop:**

for loop provides a concise way of writing the loop structure. Unlike a while loop, a for statement consumes the initialization, condition and increment/decrement in one line thereby providing a shorter, easy to debug structure of looping. (for 迴圈提供了一種簡潔的方式來編寫迴圈結構。與 while 迴圈不同，for 語句將初始化、條件判斷和遞增/遞減操作整合在同一行，從而提供了一種更簡短且易於除錯的迴圈結構。)

**Syntax:**

```
for (initialization condition; testing condition; increment/decrement)
{
    statement(s)
}
```

**Example:**

```
class ForLoop
{
    public static void main(String[] args)
    {
        for ( int i = 1; i <= 10; i++)
            System.out.println( "value of i :" + i);
    }
}
```

---

**do while:**

do while loop is similar to while loop with only difference that it checks for condition after executing the statements, and therefore is an example of Exit Control Loop.(do while 迴圈與 while 迴圈類似，唯一的不同點是它在執行語句後才進行條件判斷，因此是一種出口控制迴圈的範例。

**Syntax:**

```
do
{
    statements..
}
while (condition);
```

**Example:**

```
class DoWhileLoop
{
    public static void main(String[] args)
    {
        int i = 1 ;
        do
        {
            System.out.println( "value of i :" + i);
            i++;
        }
        while (i <= 10 );
    }
}
```

### Enhanced for loop:

Enhanced for loop provides a simpler way to iterate through the elements of a collection or array. It is inflexible and should be used only when there is a need to iterate through the elements in sequential manner without knowing the index of currently processed element. (增強型 for 迴圈提供了一種更簡單的方法來遍歷集合或陣列的元素。它的靈活性較低，應僅在需要按順序遍歷元素且不需要知道當前處理元素的索引時使用。)

Also note that the object/variable is immutable when enhanced for loop is used i.e it ensures that the values in the array can not be modified, so it can be said as read only loop where you can NOT update the values as opposite to other loops where values can be modified. (此外，請注意，當使用增強型 for 迴圈時，物件/變數是不可變的，也就是說，它確保陣列中的值無法被修改。因此，可以將其視為唯讀迴圈，在這種迴圈中，您無法更新數值，而與其他迴圈不同，其他迴圈允許修改值。)

### Syntax:

```
for (T element: Collection obj/array)
{
    statement(s)
}
```

### Example:

```
public class EnhancedForLoop
{
    public static void main(String[] args)
    {
        String array[] = { "John" , "Anderson" , "Stephen" };

        for (String s: array)
        {
            System.out.println(s);
        }
    }
}
```