

Week 2: Different types of Java loops.

What are the different loops Java provides us with?

<https://www.developer.com/design/using-different-types-of-java-loops-looping-in-java/>

There are three types of loops Java provides us with:
For, while and do while.

For the while loop; this code will continue to execute the code until the statements condition is met. In other words the code will be

```
While (x <= 10) {  
    System.out.println(x);  
    x++;  
}
```

There will be a count up until 10, to which point 11 would be next. Would be no longer less than 10. So literally "while" x is less than or equal to 10, the count up will be executed. Although the "For" loop is similar to the "while" loop it is different in the fact that the control variable, loop condition and the increment statements can all be stated in a single line of the code.

```
    cntrl Var   condition increment  
For( int x = 1;      x <= 10;      x++) {  
    System.out.println(x);  
}
```

For loops will start with initialized x = 1 then continue to x <= 10. Then will skip initialized and continue on with the condition until the statement is met. The onto the increment. If satisfied the code will be complete.

Unlike the other two loops discussed, the "do/while" loop is different in the fact that the increment (variable) is located and described at the top of the code. also is given all directions in the middle, and at the bottom is the while statement. This is why the "do/while" loop is considered an exit controlled loop

```
Int x = 1; —> Cntrl variable  
do {  
    System.out.println(x);  
    counter--; —>Increment  
}while( x <= 10); —Statement
```

The importance of using these loops is to continuously work on problems without rewriting the same code over and over again. These loops will execute code in a structured and more organized manner. They will reduce the equations down to the last possible outcome. An example that comes to mind is a magician whom has

paper bags with a sharp object in one. He will smash the first one to see nothing there. Then the next and will know which one has the object in it. Just less risky I'd say; and not necessarily the process of elimination.