CO452
Programming Concepts

Lecture 4

Iteration

Iteration

Iteration allows us to repeat statements within a block providing the condition is **true**

while

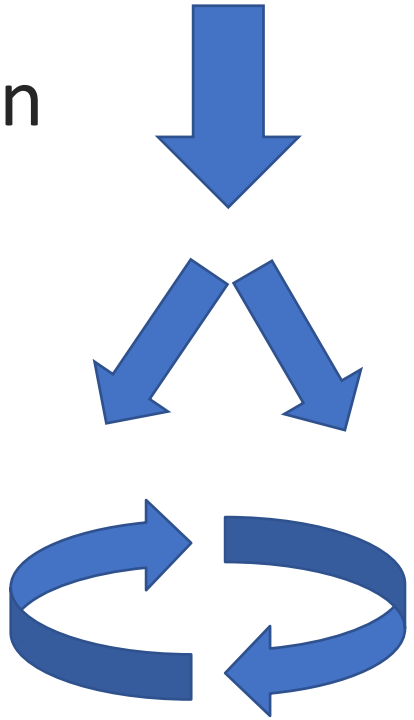
do while

for

for each

Sequence, Selection and Iteration

- **Sequence** mandates that statements be executed in order (line by line)
- **Selection** (conditional) statements will execute a **block of code once** when the condition is true
- **Iteration** allows us to **repeat** statements within a block **whilst** the condition is **true**



while

while loop

The code within the braces of an **while** statement will execute **whilst** the comparison evaluates to **true**

```
int count = 1;
while(count <= 3)
{
   System.out.println("This loop has executed " + count + " times");
   count = count + 1;
}
```

while loop

The code within the braces of an **while** statement will execute **whilst** the comparison evaluates to **true**

```
int count = 1;

while(count <= 3)
{
   System.out.println("This loop has executed " + count + " times");
   count = count + 1;
}
```

Variable initialisation

Condition: is the value of count < 3 ?

If yes, execute the contents of the loop

Increase the value of count by 1

do while

do while loop

Due to the order of statements, the body of the **do while** loop **will execute at least once** (even if comparison is false)

```
int count = 0;
do
{
   System.out.println("This loop has executed " + (count+1) + " times");
   count = count + 1;
}
while(count < 3);
```

do while loop

Due to the order of statements, the body of the **do while** loop **will execute at least once** (even if comparison is false)

```
int count = 0;
do
{
   System.out.println("This loop has executed " + (count+1) + " times");
   count = count + 1;
}
while(count < 3);
```

No condition to check! So enter the braces

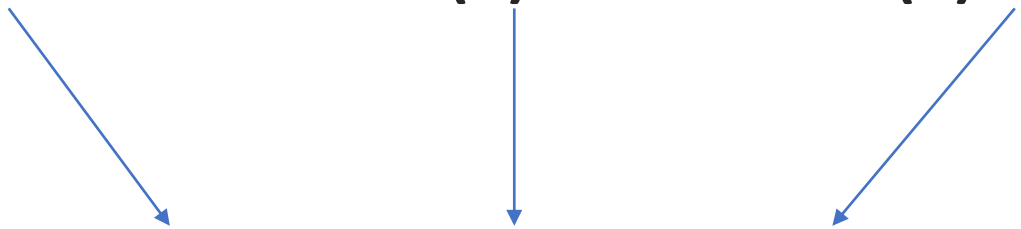
Check at the end of braces – remember ;

for

for loop

The for loop has three parts:

(1) variable initialisation (2) condition (3) increment



```
for(int count = 0; count < 3; count++)
{
   System.out.println("This loop has executed " + (count+1) + " times");
}
```

for loop

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

Initialise count to 0

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed... ");
}
```

for loop

Is 0 < 3 ?

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

Yes! (true) Therefore execute code in braces

for loop

After executing code in braces, increment count

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

count now has value of 1

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

We don’t re-initialise count back to 0!

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

Is 1 < 3 ?

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

Yes! (true) Therefore execute code in braces

for loop

After executing code in braces, increment count

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

Count now has the value of 2

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

Is 2 < 3 ?

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

Yes! (true) Therefore execute code in braces

for loop

After executing code in braces, increment count

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

Count now has the value of 3

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

Is 3 < 3 ?

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed… ");
}
```

for loop

```
for(int count = 0; count < 3; count++)
{
  System.out.println("This loop has executed... ");
}
```

No! (false) Therefore end loop and continue program

```
//continue with program...
```

for each

for each loop with collection

The for each loop can be used to iterate through collections of objects.

Requires an object to be declared of the type of item that is in the collection:

```
for(Student student : students)
{
    student.print();
}
```

for each loop with collection

The for each loop can be used to iterate through collections of objects.

Requires an object to be declared of the type of item that is in the collection:

```
for(Student student : students)
{
    student.print();
}
```

Type **variable** **ArrayList**

call print on each item in the ArrayList