Flow Control

While Loop

The Syntax of While Loop

```
while (condition) {
  // executes while condition is true
}
```

```java
// print out numbers from 0 to 9
int i = 0;
while (i < 10) {
  System.out.println(i);
  i++;
}

// it's possible that body is never executed
int a = 5, b = 7;
while (a > b) {
  System.out.println("Hello");
    // never executes
}
```

```java
// infinite loop
while (a < b) {
  System.out.println("Hello");
  // prints "Hello" forever
}

// simplest infinite loop
while (true) { }
```

```
// break statement exits the loop


int n = 0;
while (true) {
   System.out.println(n);
   if (n == 9) break;
   n++;
}
```

```
0
1
2
3
4
5
6
7
8
9
```

```
// nested loops
int i = 0, j = 0;
while (i < 3) {
  i++;
  j = 0;
  while (j < 3) {
    j++;
    System.out.print("(" + i + ", " + j + ") ");
  }
}
```

```
(1, 1) (1, 2) (1, 3) (2, 1) (2, 2) (2, 3) (3, 1) (3, 2) (3, 3)
```

```
// using break in nested loops
int i = 0, j = 0;
while (true) {
  i++;
  j = 0;
  while (true) {
    j++;
    System.out.print("(" + i + ", " + j + ") ");
    if (j == 3) break;
  }
}

// break exits only the inner loop, the outer loop is still an infinite loop
```

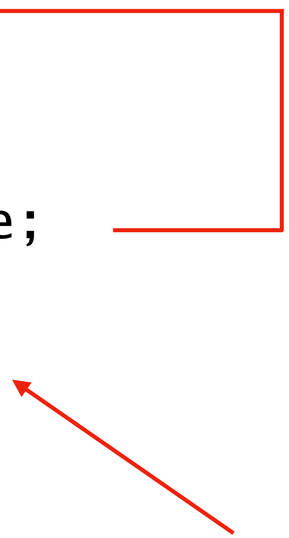
```
// using break in nested loops, with labels
int i = 0, j = 0;
OUTER_LOOP: while (true) {
            i++;
            j = 0;
            INNER_LOOP: while (true) {
              j++;
              System.out.print("(" + i + ", " + j + ") ");
              if (j == 3) break OUTER_LOOP;
             }
}
```

```
(1, 1) (1, 2) (1, 3)
```

```
// be careful about the unreachable code

int i = 0, j = 0;

OUTER_LOOP: while (true) {

            i++;

            j = 0;

            INNER_LOOP: while (true) {

                        j++;

                        System.out.print("(" + i + ", " + j + ") ");

                        if (j == 3) break OUTER_LOOP;

             }

            System.out.println("Hello");

}
```

unreachable code => does not compile!

```
// continue statement skips one iteration of the loop


// task: print all even numbers between 0 and 20
int i = -1;
while (i < 20) {
  i++;
  if (i % 2 == 1) continue;
  System.out.println(i);
}
```



this is not reached if number is odd

```
// return statements breaks the execution of the loop (exits the method)
public void printPairs() {
  int i = 0, j = 0;
  while (true) {
    i++;
    while (true) {
      j++;
      System.out.print("(" + i + ", " + j + ") ");
      if (j == 4)
        return;
      }
  }
}

printPairs();
```

exits the method

```
(1, 1) (1, 2) (1, 3) (1, 4)
```