BOOLEAN, IF-ELSE & SWITCH

BY ANNEMARIE CABALLERO
I.T. GIRLS AUGUST 13-17

BOOLEAN

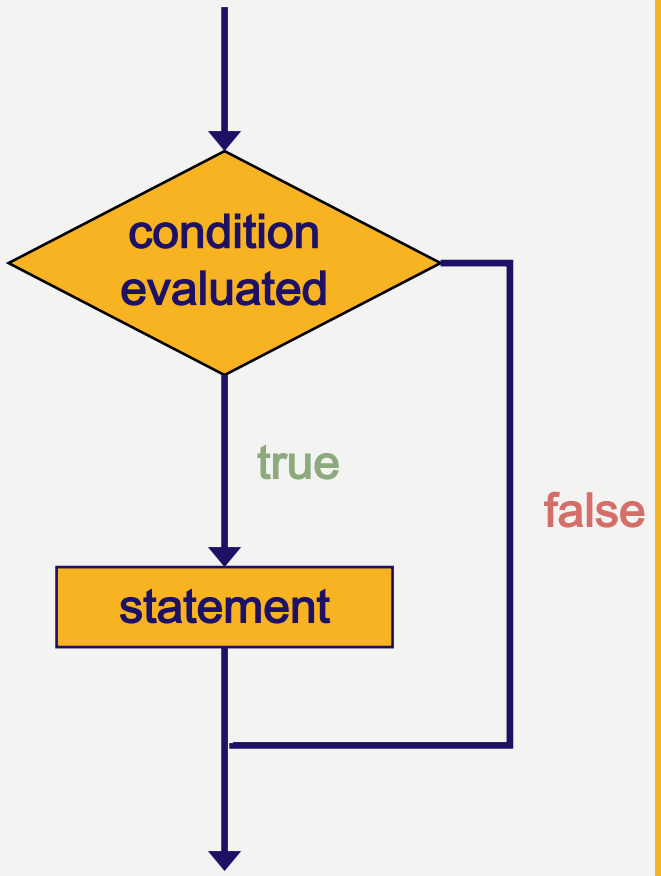
One of the primitive data types

Two values: true and false

Can be used in conditionals

CONDITIONALS

- Have a test statement (which must evaluate to true or false)
- Often used to allows programs to respond to the user
- There are two types
  - If-Else statements
  - Switch statements
    - (mostly used when there are multiple possibilities)
- Example:
  - Executing code in a guessing game, when the user enters the right guess



```
1 public class IfStatements {
2     public static void main(String[] args) {
3         boolean helloWorld = false;
4 
5         if(helloWorld)
6             System.out.println("boolean helloWorld is true");
7     }
8 }
```

IF SYNTAX

```java
import java.util.Random;
public class ElseStatements {
    public static void main(String[] args) {
        // just sets helloWorld to true or false randomly
        Random rand = new Random();
        boolean helloWorld = rand.nextBoolean();

        if(helloWorld)
            System.out.println("Hello World");
        else {
            System.out.println("Something other than HelloWorld");
            System.out.println("You need {} if there's more than one statement");

        }
    }
}
```

IF-ELSE SYNTAX

OPERATORS

Syntax:	Means:
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to
==	**Equals (diff for String)**
!=	Not equals

= means assignment
== means comparison

Syntax:	Means:
&&	AND
||	OR
!	NOT

BUT WHAT ABOUT MULTIPLE POSSIBLE VALUES FOR THE TEST CONDITION?

```
1  public class IfElseStatements {
2      public static void main(String[] args) {
3          //the statement below just sets random
4          //to a number between 1 & 9
5          int rand = (int)(Math.random() * 9 + 1);
6
7
8          if(rand < 4)
9              System.out.println("123, eyes on me");
10         else if(rand >= 4 && rand <= 6)
11             System.out.println("456, nvm not good at rhymes");
12         else
13             System.out.println("789, refer to above");
14     }
15 }
```

NESTED IFS – putting ifs in other ifs

```
1  public class ChristmasTest {
2     public static void main(String[] args) {
3        int month = 12;
4        int day = (int) (Math.random() * 31 + 1);
5
6        //just testing for xmas
7        if(month == 12 && day == 25)
8           System.out.println("Merry Christmas");
9        else
10          System.out.println("Still not Christmas");
11
12       //testing for december holidays
13       if(month == 12) {
14          if(day == 25)
15             System.out.println("Merry Christmas!");
16          else if(day >= 2 && day <= 10)
17             System.out.println("Happy Hanukkah!");
18          else
19             System.out.println("Not Christmas or Hanukkah");
20       }
21    }
22 }
```

SWITCH STATEMENTS

- Easier (cleaner) option for testing multiple values
- Checks a variable for multiple possible options (does not use a single boolean)
- Case statements will continue until they hit break or return a value

```
1  public class SwitchStatement {
2     public static void main(String[] args) {
3        int month = (int) (Math.random() * 12 + 1);
4
5        switch(month) {
6           case 12: case 1: case 2:
7              System.out.println("Winter is cold");
8              break;
9           case 3: case 4: case 5:
10             System.out.println("Spring is sprung");
11             break;
12          case 6: case 7: case 8:
13             System.out.println("What time is it? SUMMERTIME");
14             break;
15          case 9: case 10: case 11:
16             System.out.println("Pro: Apple Cider, Con: School");
17             break;
18          default:
19             System.out.println("I screwed something up");
20
21       }
22    }
23 }
```