- Reminder:
  Assignment 1 is due Monday 11:59 pm
- Reading for next class is LDC Chapter 5
- **SI sessions** and drop-in hours starting!
  - Check out the course online calendar.

- **Java has a boolean type that can take the value true or false**
- Booleans arise naturally when using **relational operators** to compare two values

<table>
<thead>
<tr>
<th></th>
<th>3 &lt; 5</th>
<th>3 &lt; 2</th>
<th>3 &gt; 2</th>
<th>5 &lt;= 1</th>
<th>5 &gt;= 1</th>
<th>5 == 5</th>
<th>5 == 6</th>
<th>5 != 6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Boolean values can be manipulated with the logical operators `!` (not), `&` (and), and `||` (or)

|      | ! (3 < 5) | ! (3 == 5) | (3 > 5) && (7 < 8) | (3 < 5) && (7 < 8) | (3 > 5) || (7 < 8) | (3 > 5) || (7 > 8) |
|------|-----------|-------------|---------------------|---------------------|---------------------|---------------------|
|      |           |             |                     |                     |                     |                     |
A **predicate** is any method that returns a **boolean** value.

```java
public static boolean isEven(int n){
    return (n % 2) == 0;
}

public static boolean isDivisibleBy(int num, int factor){
    return (num % factor) == 0;
}

public static boolean isBetween(double n, double lo, double hi){
    return (lo <= n) && (n <= hi);
}
```

Write your own predicate to determine if \( n \) is odd. Then, can you write it another way?

```java
public static boolean isOdd(int n){
    return (n%2) == 1;
}

public static boolean isOdd(int n){
    return !isEven(n);
}
```

To choose between two courses of action, to control the program flow, we use **conditional statements** such as if, else if, and else.

```java
public static double abs(double n){
    if (n < 0) {
        return -n;
    } else {
        return n;
    }
}

public static void main(String[] args){
    int x = 28; String s = "meow";
    if(x < 30 && s.length() < 10){
        x = x + 5;
        int y = s.length();
        if(x+y > 36){
            System.out.println("hello " + x);
        }else if(x+y < 33){
            System.out.println("howdy " + y);
        }else{
            System.out.println("hi!");
        }
    } else {
        x = x - 10;
        int y = s.length() + 5;
        if (x == 15) System.out.println("Salut " + x);
        else System.out.println("Ciao " + y);
    }
}
```
**while**

- **Iteration** refers to a sequence of steps that is repeated until some stopping condition is reached.

```java
while(boolean_expression){
    statement 1;
    statement 2;
    ...
}
```

(1) evaluate boolean expression
(2) if true, execute body of loop and go back to step (1)
(3) if false, go to statement after while loop

---

**for**

```java
for (initialization of index variable; boolean expn; update index variable) {
    statements
}
```

- Execute this statement once before entering loop
- If true, execute body of loop
- If the boolean expression was true, the body of the loop will be executed
- If the boolean expression evaluates to false, drop down to here

Write a for loop that prints the numbers from 1 to 10.

```java
for (int i = 1; i < 10; i++) {
    System.out.println("CS230");
}
```

---

**Math Class**

```java
System.out.println(Math.max(100, 50));
System.out.println(Math.sqrt(25));
System.out.println(Math.log(10));
```

// Given area of circle, returns the circle's radius.
// Since area=pi*r*r, we have r = squareRoot(area/pi).
**public static double getCircleRadius(double area) {**
    return Math.sqrt(area/Math.PI);
**}**

```java
System.out.println(getCircleRadius(100));
```

---

**Random Class**

```java
import java.util.Random;

public class RandomExample {
    public static void main(String[] args) {
        Random rand = new Random();
        for (int i = 0; i < 15; i++) {
            System.out.println(rand.nextInt(10));
        }
    }
}
```
// Returns true if character is lower-case
// vowel (a, e, i, o, u), false otherwise.
public static boolean isVowel(char ch)

// Returns the number of occurrences of
// vowels in the String s
public static int countVowels(String s)