2. (100 points) The following statements are equivalent:

```
if(booleanVariable == true)
```
is the same as
```
if(booleanVariable)
```

Explain in your own words why this is. Why can we simply use the boolean variable versus comparing it to the desired value?

Then, write two equivalent statements for checking if `booleanVariable` is `false`.

The natural state of boolean is true so putting == true is redundant. We can leave out the == true because the program is already checking.

if (booleanvariable == false
is same as
if (!(booleanVariable))

3. (100 points) Why should we never compare floating point values using exact equality comparison? What is a better way to compare floating point values? Show a code snippet of an if statement using this method to compare if a floating point variable is equal to 5.5 to at least the 6th decimal place of precision.

The way floating point values round is different than normal rounding causing ridiculous round off errors. Checking a close range of values allows the program to be more accurate

```
if ( floatingPoint >= 5.499 999 || floatingPoint <= 5.500001 )
```