

Warning

Introduction
to
Programming

Laura Hobbes
LeGault

Review

Using Loops

We gonna write a *lot* of code today.

Three types of loops

Introduction
to
Programming

Laura Hobbes
LeGault

Review

Using Loops

```
while(<cond>) { ... }
```

```
for (<init>; <cond>; <update>) { ... }
```

```
do { ... } while (<cond>);
```

Note: the conditions we're testing are not “are we there yet?”, but rather “should we keep going?”

→ Make sure the body of the loop will eventually make the condition false

Constructing a good loop

Introduction
to
Programming

Laura Hobbes
LeGault

Review

Using Loops

From last time: Write a loop to add all odd numbers between (already declared in prior code) `int x` and `int y`, inclusive.

If I make my condition (`i < y`), under which conditions will this be correct? Under which conditions will this be incorrect?
→ This is called an *off-by-one error* (even though here we'd be off by two)

Take home message: Be very careful of your starting conditions, as well as (`<` vs. `<=`) and (`>` vs. `>=`).

User Input Validation

Introduction
to
Programming

Laura Hobbes
LeGault

Review

Using Loops

Because this is *really important*:

Write some code to get an integer between 1 and 10 from the user. Declare any variables you need (but you don't need to import anything).

Using loops to do useful operations!

Introduction
to
Programming

Laura Hobbes
LeGault

Review

Using Loops

We began today writing some code to sum all odd numbers in a range. We can do other things with loops, too:

- 1 Averaging numbers
- 2 Counting occurrences (e.g. number of 'a's in a String)
- 3 First/last match (`str.indexOf()` or `str.lastIndexOf()`)
- 4 Max/min value entered

Nesting loops

Introduction
to
Programming

Laura Hobbes
LeGault

Review

Using Loops

Just as we could nest `if` statements, we can put loops inside other loops.

Example: explaining square roots to a four-year-old (extra challenge: write a program which determines the root using `Random`, outputs the square, and asks the user for the root)