

Introduction to Java Programming

Lecture 2 of 8
March 26, 2019
Emerson Family School

Maxwell Young

if/else Statements

Allows you to specify actions of your code by testing if a condition or property is satisfied

```
if(condition A holds){ ..... note open bracket
    // insert your code here
} ..... note closed bracket
else{
    // insert some different code here
}
```

You don't need to include an else statement, can have

```
if(condition A holds){
    // insert your code here
}
```

if/else Statements

Create a new project, name it whatever you wish.

In main(), declare an double **x** and initialize it to some value in between -100 and 100.

In the next few lines, write code to determine if **x** is less than 0, or equal to or greater than 0.

For each of these two cases, print the appropriate message to the screen:

The number [your value] is positive

The number [your value] is negative

The actual value should be printed to the screen

if/else Statements

Repeat the same logic, but declare a integer **y** and test whether it is odd or even.

if/else Statements

You can generalize this:

```
if(condition A holds)
    // insert your code here
else if(condition B)
    // insert some different code here
else if(condition C)
    // insert some different code here
else
    // insert some different code here
```

while Loops

Remember I asked you to output values from an array?

It was tedious because you had to rewrite println statements

Loops allow you execute such operations more efficiently

```
while( Condition ){  
    // some code  
}
```

Loop will run until condition becomes false

while Loops

Here's an example:

```
int i=0;
while( i<5 ){
    System.out.println("Hello, World!");
    i = i+1;
}
```

What is this code doing?

while Loops

I could have used the ++ operator to make code a little shorter:

```
int i=0;
while( i<5 ){
    System.out.println("Hello, World!");
    i++;
}
```

I could have used the ++ operator to make code a little shorter:

while Loops

What happens if I remove the i++ line?

```
int i=0;
while( i<5 ){
    System.out.println("Hello, World!");
}
```

while Loops

What happens if I remove the `i++` line?

```
int i=0;
while( i<5 ){
    System.out.println("Hello, World!");
}
```

Condition `i<5` always remains true.

Loop will execute forever.

while Loops

1D arrays have a length

```
int[] myArray = new int[20];  
System.out.println("Length of array is: " + myArray.length);
```

Using these new skills, write a while loop that prints out all values in myArray

while Loops

Now, write a loop that sets the value of `myArray[i]` to `i`

for Loops

Similar to while loops, but sometimes more appropriate

```
for( int i=0; i<5; i++ ){  
    System.out.println("Hello, World!");  
}
```

Can I have an infinite loop?

Lab 2

You will build a tic-tac-toe game. Gain practice with:

if/else statements

while loops

arrays

input from user