

For Loops }

Client Side Web

Loops

- { What and why?
- { Loop types
- { Creation
- { Loops and Arrays

Loops

- { What and why?
- { Loop types
- { Creation
- { Loops and Arrays



What and why?



When coding gets
too repetitive

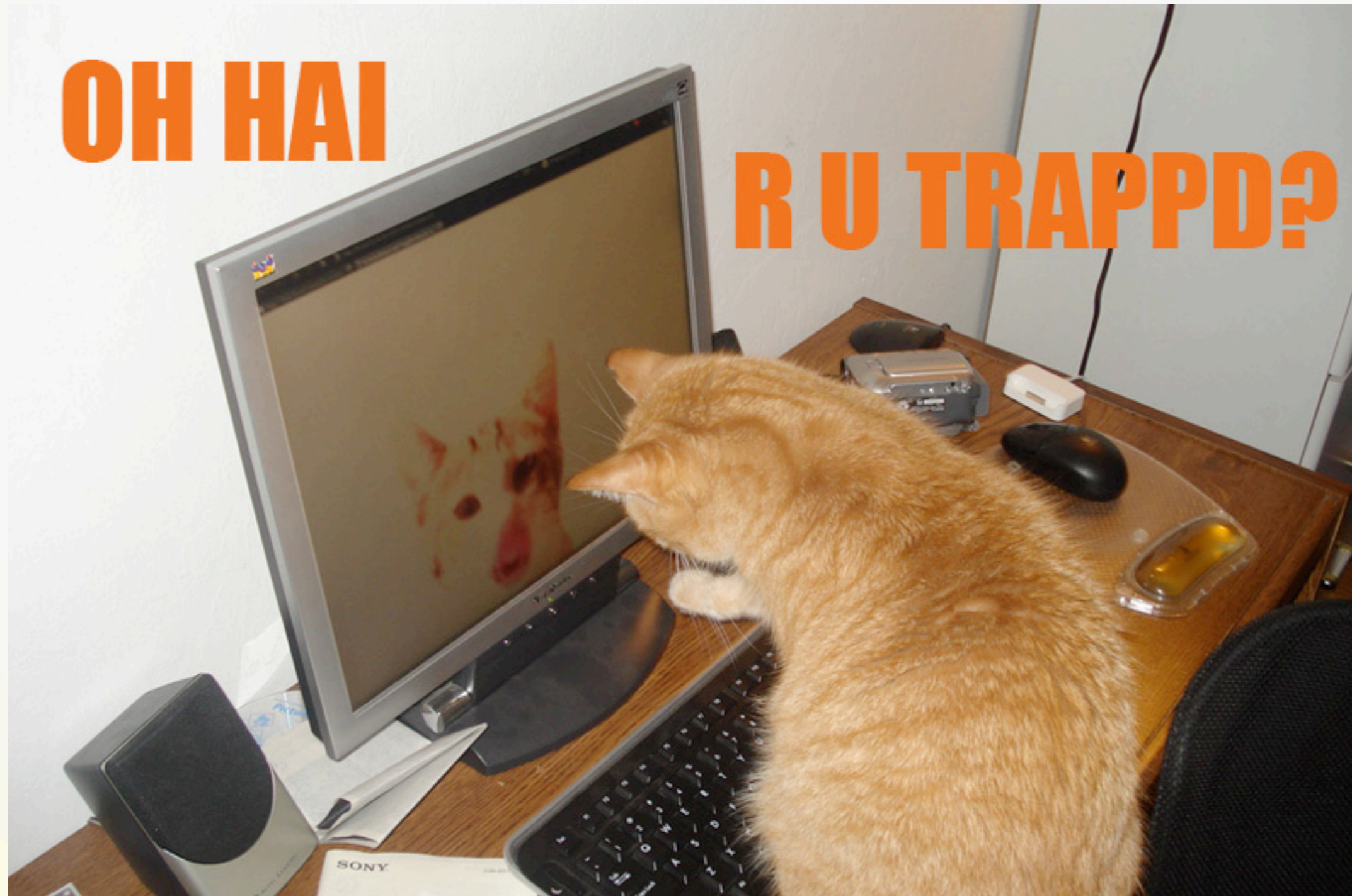




When copy and
paste takes too
way long

OH HAI

R U TRAPPD?





When you're
adjusting more
than you're solving

Going to Stop Lex Luthor



Loop types




Loops

— { for

— { foreach

— { while

— { do while



We will
concentrate on
for loops



Creation

Anatomy of a For Loop

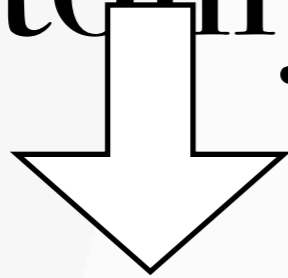
```
for ( ) {  
}  
}
```

Anatomy of a For Loop

```
for ( ) {  
}  
}
```

The for loop starts off just like an if statement

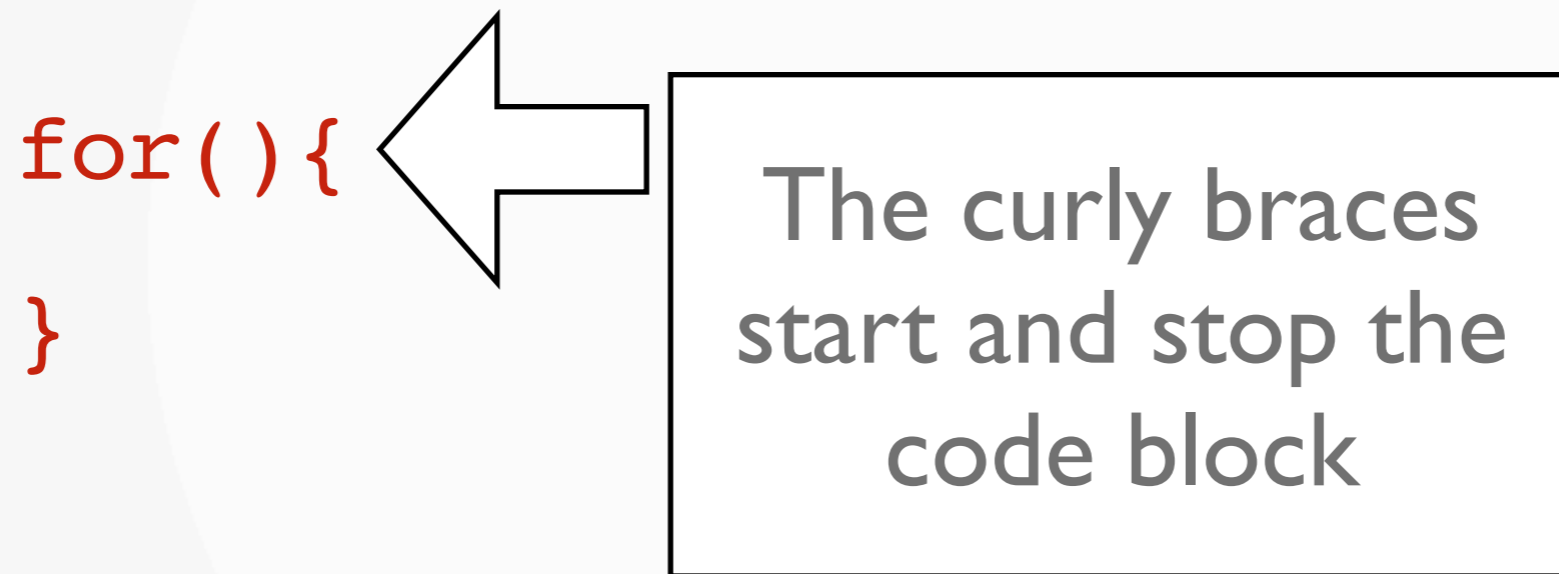
Anatomy of a For Loop



```
for ( ) {  
}  
}
```

The argument is a
condition or sorts

Anatomy of a For Loop



Anatomy of a For Loop

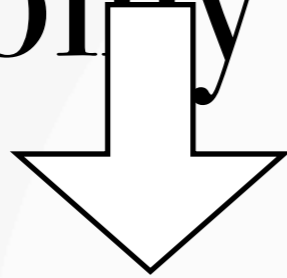
```
for(i=0; i<5; i++){  
    // run task  
}
```

Anatomy of a For Loop

```
for(i=0; i<5; i++){  
    // run task  
}
```

It's not as scary
as it looks

Anatomy of a For Loop



```
for(i=0; i<5; i++){  
    // run task  
}
```

“**i**” is just a variable
that equals 0
(*this is the counter*)

Anatomy of a For Loop

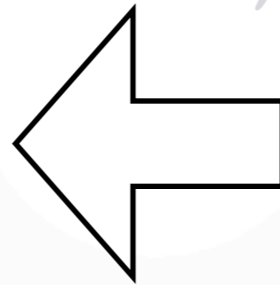


```
for(i=0; i<5; i++){  
    // run task  
}
```

is **i** less than 5? If yes then proceed
(*this is know as the condition*)

Anatomy of a For Loop

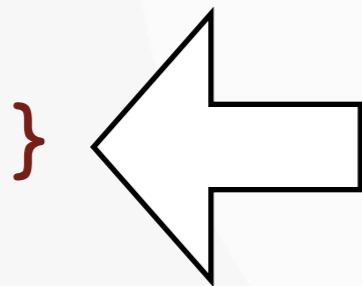
```
for(i=0; i<5; i++){  
    // run task  
}
```



execute code

Anatomy of a For Loop

```
for(i=0; i<5; i++){  
    // run task
```



end of code block

Anatomy of a For Loop



```
for(i=0; i<5; i++){  
    // run task  
}
```

add 1 to i
(*this is the
incrementer*)

Anatomy of a For Loop

```
for(i=0; i<5; i++){  
    // run task  
}
```

```
i = 0  
0 + 1 = 1  
i = 1
```

Anatomy of a For Loop

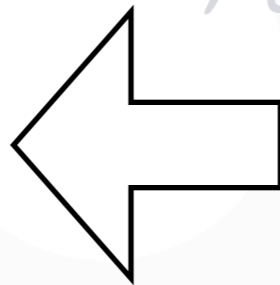


```
for(i=0; i<5; i++){  
    // run task  
}
```

is i less than 5?

Anatomy of a For Loop

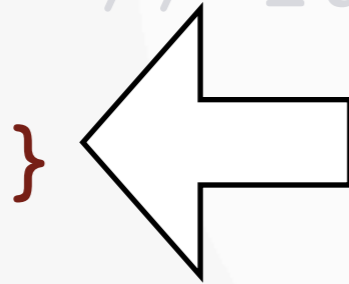
```
for(i=0; i<5; i++){  
    // run task  
}
```



Execute code

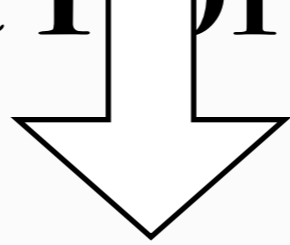
Anatomy of a For Loop

```
for(i=0; i<5; i++){  
    // run task
```



End of code block

Anatomy of a For Loop



```
for(i=0; i<5; i++) {  
    // run task  
}
```

add 1 to i

Anatomy of a For Loop

```
for(i=0; i<5; i++){  
    // run task  
}
```

```
i = 1  
1 + 1 = 2  
i = 2
```

Is 2 less than 5?





Yes, the loop
continues...

Anatomy of a For Loop

```
for(i=0; i<5; i++){  
    // run task  
}
```

Loops and Arrays

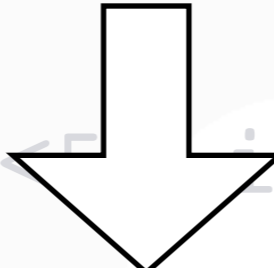


Anatomy of a For Loop

```
for(i=0; i<5; i++){  
    alert(myArray[i])  
}
```

Anatomy of a For Loop


```
for(i=0; i<5; i++) {  
    alert(myArray[i])  
}
```



Take note of *i* as
the index for the
array

Anatomy of a For Loop

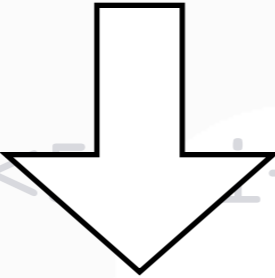
```
for(i=0; i<=i++) {  
    alert(myArray[i])  
}
```



What does i equal?

Anatomy of a For Loop

```
for(i=0; i<=i++){  
    alert(myArray[i])  
}
```



What does i equal?

i = 0...

then 1...

then 2...

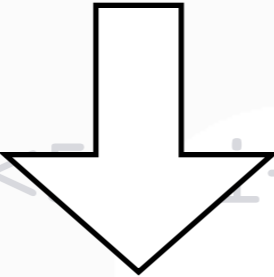
then 3...

then 4...

then... end loop

Anatomy of a For Loop

```
for(i=0; i<=i++) {  
    alert(myArray[i])  
}
```



This statement will output the contents of the array based on the dynamic index value

Anatomy of a For Loop

```
for(i=0; i<5; i++){  
    alert(myArray[i])  
}
```