

Loops

UC Berkeley Graduate School of Journalism

An array can be accessed by index

```
someArray[0]
```

```
someArray[1]
```

```
someArray[2]
```

```
someArray[3]
```

```
someArray[4]
```

```
someArray[5]
```

```
someArray[6]
```

An array can be accessed by index

`someArray[0]`

`someArray[1]`

`someArray[2]`

`someArray[3]`

`someArray[4]`

`someArray[5]`

`someArray[6]`

`someArray[i]`

For Loop

```
for(var i=0; i<10; i++){
```

```
}
```

For Loop

```
for(var i=0; i<10; i++){
```

Code here executes over and over

```
}
```

For Loop

```
for(var i=0; i<10; i++){
```

The first time

`i = 0;`

```
}
```

For Loop

```
for(var i=0; i<10; i++){
```

The second time

i = 1;

```
}
```

For Loop

```
for(var i=0; i<10; i++){
```

The third time

`i = 2;`

```
}
```


For Loop

```
for(var i=0; i<10; i++){
```

The fourth time

`i = 3;`

```
}
```

For Loop

```
for(var i=0; i<10; i++){
```

The fifth time

`i = 4;`

```
}
```

Let's say I have an array...

```
var myDogs = ["Fido", "Rex", "T-Bone"];
```

Will add every dog

```
for(var i=0; i<10; i++){  
    $('#results').append(myDogs[i]);  
}
```

How a For Loop Works

for( ;  ; )

Three parts:

How a For Loop Works

for ( ;  ; )

Three parts:

initialization

How a For Loop Works

for ( ;  ; )

Three parts:

initialization

condition

How a For Loop Works

for ( ;  ; )

Three parts:

initialization

condition

afterthought

How a For Loop Works

Initialization Will do this before the loop begins

Condition The loop keeps going as long as this is a TRUE statement. Once it's FALSE, the loop stops.

Afterthought After each iteration, it will execute this.

For Loop

```
for(var i=0; i<10; i++){
```

}

For Loop

```
for(var i=0; i<10; i++){
```

The initialization

We set "i" to zero

```
}
```

For Loop

```
for(var i=0; i<10; i++){
```

The condition
Keep going while "i"
is less than 10.

```
}
```

For Loop

```
for(var i=0; i<10; i++) {
```

The afterthought
Increment "i" by 1
each time

```
}
```

A few more operators

<code>+=</code>	Increment a variable by certain amount
<code>-=</code>	Decrement a variable by a certain amount

A few more operators

<code>+=</code>	Increment a variable by certain amount
<code>-=</code>	Decrement a variable by a certain amount
<code>++</code>	Increment a variable by 1

A few more operators

<code>+=</code>	Increment a variable by certain amount
<code>-=</code>	Decrement a variable by a certain amount
<code>++</code>	Increment a variable by 1
<code>--</code>	Decrement a variable by 1

What does this loop do?

```
for(var i=3; i<50; i++){
```

What does this loop do?

```
for(var i=3; i<50; i++){
```

values for i:

3, 4, 5, 6 ... 47, 48, 49

```
}
```


What does this loop do?

```
for(var i=0; i<=5; i++){
```

values for i:

0, 1, 2, 3, 4, 5

```
}
```

What does this loop do?

```
for(var i=0; i>10; i++){
```

What does this loop do?

```
for(var i=0; i>10; i++){
```



values for i:

```
}
```

What does this loop do?

```
for(var i=0; i<10; i+=2){  
  
  
  
  
  
  
  
  
  
}
```

What does this loop do?

```
for(var i=0; i<10; i+=2){
```

values for i:

0, 2, 4, 6, 8

```
}
```


What does this loop do?

```
for(var i=10; i>5; i--){  
  
  
  
  
  
  
  
  
}
```

What does this loop do?

```
for(var i=10; i>5; i--){
```

values for i:

10, 9, 8, 7, 6

```
}
```

Practical Use

```
for(var i=0; i<100; i++){  
  
    $('#results').append(myDogs[i]);  
  
}
```

Determining how many times to loop

```
var someArray = ["bob", "jones", "sara"];
```

Determining how many times to loop

```
var someArray = ["bob", "jones", "sara"];
```

```
someArray.length
```

Determining how many times to loop

```
var someArray = ["bob", "jones", "sara"];
```

`someArray.length`

Real World Example

```
for(var i=0; i < myDogs.length; i++){
```

```
}
```

Real World Example

```
for(var i=0; i < myDogs.length; i++){
```

```
}
```


Real World Example

```
for(var i=0; i < myDogs.length; i++){  
    $( '#results' ).append(myDogs[i]);  
  
}
```