

WHILE LOOPS

PROF. DAVID ROSSITER

AFTER THIS PRESENTATION

- You'll be able to create `while` loops
- You'll be able to create `do ... while` loops

WE WILL LOOK AT

`while`

`do ... while`

LOOPS

- A loop repeats some code again and again
- Here we will look at:
 - `while`
 - `do ... while`

WHILE LOOPS

- A while loop is the simplest loop

```
while (condition) {
```

... code goes here ...

```
};
```

- Each time the loop content is executed we call it an *iteration*

INDEXOF()

- `string.indexOf("text")`
gives you the location of the first "text" in the string
- For example:

```
var text = "The cat's hat was wet";  
result = text.indexOf("at");
```

- *result* will be 5

```
<!doctype html>
<html><head>
  <title>Example of while()</title>
  <script>
    var response, finished;
    finished=false;
    alert("Rossiter is a great name.");
    while (!finished){
      response=prompt("Do you agree?");
      if (response.indexOf("y")==0)
        finished=true;
    }
  </script>
</head></html>
```

×

Rossiter is a great name.

OK

×

Do you agree?

I do not

OKCancel

×

Do you agree?

No

OKCancel

×

Do you agree?

yea

OKCancel

DO ... WHILE LOOPS

- do ... while is an 'upside-down' version of while

```
do {
```

... code goes here ...

```
} while (condition);
```

- A do ... while loop will be executed at least once

```
<!doctype html>
<html><head>
  <title>Example of do .. while()</title>
  <script>
    var response, finished;
    finished=false;
    alert("Rossiter is a great name.");
    do {
      response=prompt("Do you agree?");
      if (response.indexOf("y")==0)
        finished=true;
    } while (!finished);

  </script>
</head></html>
```