香港科技大學
THE HONG KONG
UNIVERSITY OF SCIENCE
AND TECHNOLOGY

INTRODUCTION TO EVENTS AND FUNCTIONS

PROF. DAVID ROSSITER

THE DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系

AFTER THIS PRESENTATION

- You'll appreciate the concept of events
- You'll understand how to use functions

WE WILL LOOK AT

Events	`onload`
Functions	`function`
	`return`

EVENTS

- An event is when something happens
- For example:
  - Click on something
  - Move the mouse
  - Press a key on the keyboard
- You can arrange for some code that you write to be executed when the event occurs

ONLOAD EVENT

- *onload* is triggered when the object has loaded

```
<body onload="alert('Hello!')">
```

. . . the main web page content goes here . . .

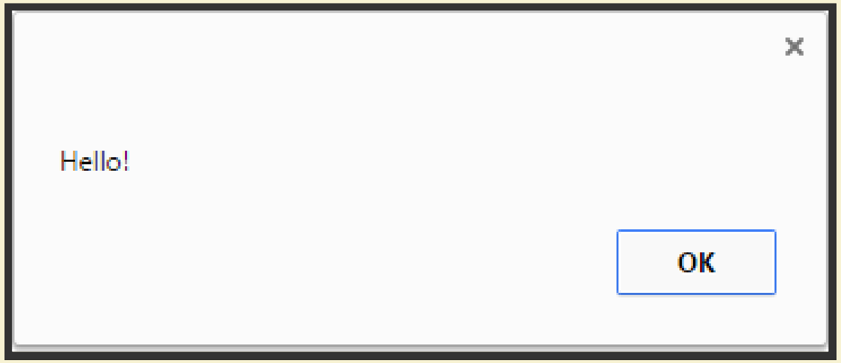
```
</body>
```

EXAMPLE

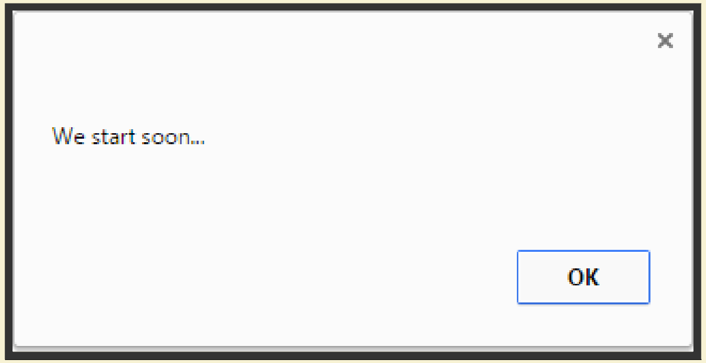
```html
<!doctype html>
<html>
    <body onload="alert('Hello!')">
        <p>
            A message is shown as soon
            as the page is loaded.
        </p>
    </body>
</html>
```

You can execute as much code as you like

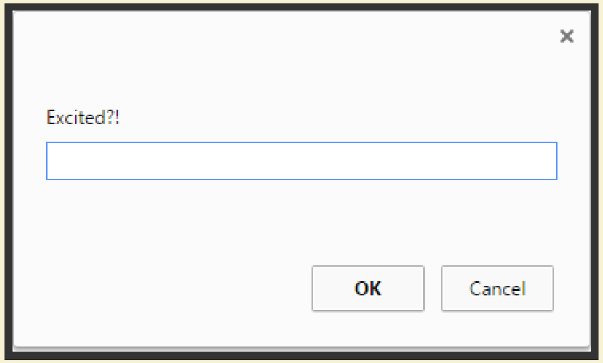
```
<!doctype html>
<html>
    <body onload="alert('Hello!');
            alert('We start soon...');
            prompt('Excited?!') ">
        <p>
            3 popup windows are shown as
            soon as the page is loaded.
        </p>
    </body>
</html>
```



Hello!



We start soon...



Excited?!

FUNCTIONS

- A function is a group of code:

```
function do_something() {
```

. . . code goes here . . .

```
}
```

- Run the function like this:

```
do_something();
```

```html
<!doctype html>
<html>
    <head>
        <title>Example of a function</title>
        <script>
            function greet_the_user(){
                alert('Hello!');
                alert('We start soon...');
                prompt('Excited?!')
            }
        </script>
    </head>
    <body onload="greet_the_user()">
    </body>
</html>
```

FUNCTION PARAMETERS

You can pass something to a function

```
function purchase( cats ) {
```

. . . *code here uses* cats . . .

```
}
```

- Run the function like this:

```
purchase( 10 );
```

FUNCTION RESPONSE

You can get a response from a function

```
function do_something() {
```

. . . *code here stores something in* answer . . .

```
    return answer; }
```

- Use the function like this:

```
result = do_something();
```

```
<!doctype html>
<html><body onload="check_user_age()" style="position:absolute">
    <h1>This is my naughty home page.</h1>
    <script>
        function check_user_age(){
            if (age_of_user() < 18)
                alert("Please go to another page.");
        }
        function age_of_user(){
            var age_text, age;
            age_text=prompt("What is your age?");
            age=parseInt(age_text);
            return age;
        }
</script></body></html>
```

A RECURSIVE FUNCTION

A function can call itself

```
function do_something( control_value ) {
```

. . . *code here calls* do_something(. . .)

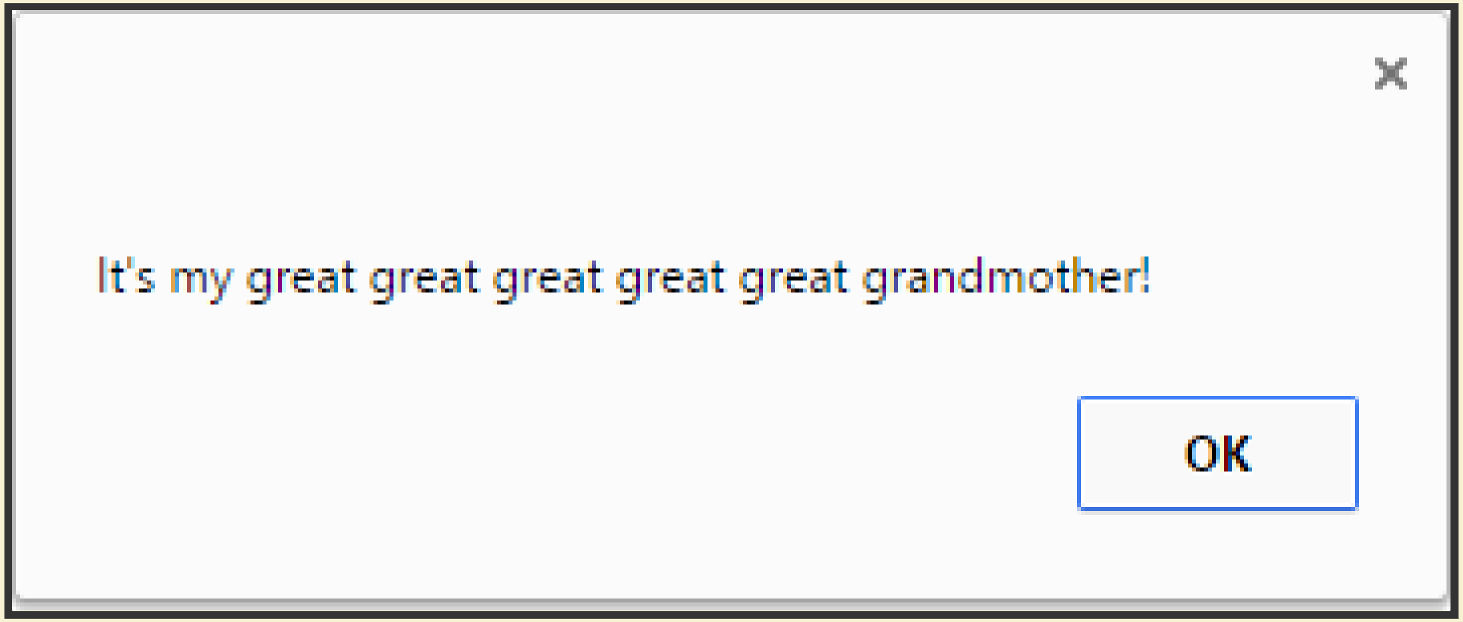
```
}
```

- Start the function like this:

```
result = do_something( 10 );
```

```
<!doctype html>
<html><body>
  <script>
    alert("It's my " + build_great(5) +
          "grandmother!");

    function build_great( depth ) {
      if (depth > 0)
        return "great " + build_great( depth - 1 );
      else
        return "";
    }
</script>
</body></html>
```



It's my great great great great great grandmother!