## Java Script (Methods).

## JavaScript popup box are extremely useful when you start coding in JavaScript. They enable you to write basic programs based on the Input/Process/Output very easily.

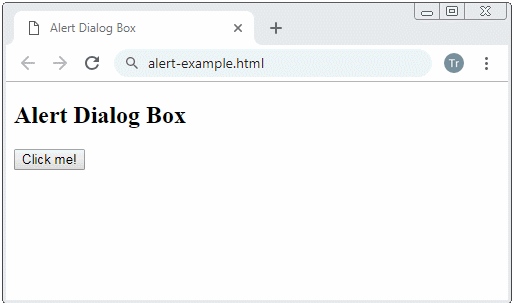
## JavaScript Dialog Box

JavaScript provides three important Dialog Boxes, which include Alert Dialog Box for users, Confirmation DialogBox, and Prompt Dialog Box.

## Java Script (Functions).

## 1- Alert Dialog Box

**Alert Dialog Box** is mainly used to display a notice, warning, or error to users. Basically, you cannot customize dialog box icon or title, ... you can only provide the message that the dialog box will display. In addition, **Alert Dialog Box** has only one **OK**button to close a dialog box.



To display a **Alert Dialog Box,** you call the **alert(message)** function, in which the **message** is the content that the dialog box will display.

alert-example.js

<!DOCTYPE **html**>

<**html**>

<**head**>

<**title**>Alert Dialog Box</**title**>

<**script** type="text/javascript">

**function** **testAlertDialog**() {

**alert**("Something Error!");

}

</**script**>

</**head**>

<**body**>

<**h2**>Alert Dialog Box</**h2**>

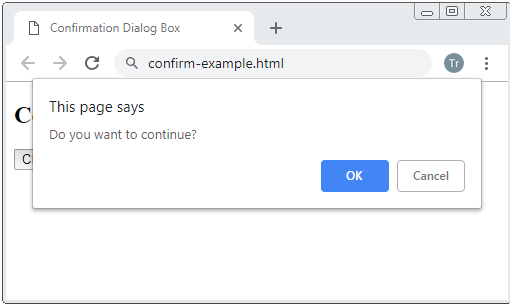
<**button** onclick="testAlertDialog()">Click me!</**button**>

</**body**>

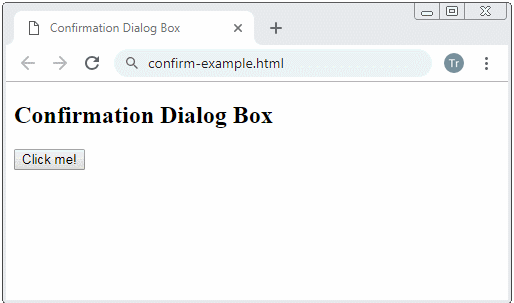
</**html**>

## 2- Confirmation Dialog Box

**Confirmation Dialog Box**  is used to ask the user to confirm something. This dialog is very simple, you cannot customize the icon or the title of the dialog box, you can only provide a message asking the user to confirm. This dialog box has 2 **OK** and **Cancel** buttons.



To display a **Confirmation Dialog Box** you call the **confirm(message)**function, in which the **message** is one requesting an user to confirm. If the user clicks the **OK**button, this function returns **true**, otherwise if the user clicks the **No** button, this function returns**false**.



confirm-example.js

<!DOCTYPE **html**>

<**html**>

<**head**>

<**title**>Confirmation Dialog Box</**title**>

<**script** type="text/javascript">

**function** **testConfirmDialog**() {

**var** result = **confirm**("Do you want to continue?");

**if**(result) {

**alert**("OK Next lesson!");

} **else** {

**alert**("Bye!");

}

}

</**script**>

</**head**>

<**body**>

<**h2**>Confirmation Dialog Box</**h2**>

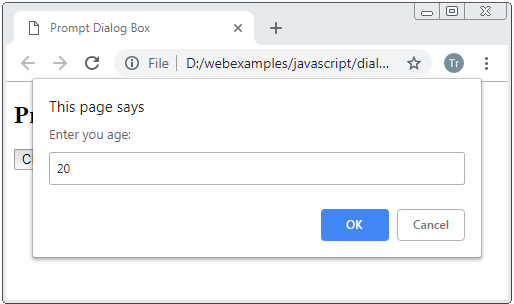
<**button** onclick="testConfirmDialog()">Click me!</**button**>

</**body**>

</**html**>

## 3- Prompt Dialog Box

**Prompt Dialog Box**  is used for users to enter an information. This dialog box is very simple. It includes a **Text Field** for users to enter information. The dialog box has 2 **OK** and **Cancel**buttons.



To display a **Prompt Dialog Box** you call the **prompt(message, defaultValue)** function in which the **message** is one for user. **defaultValue**is default value prefilled in the **Text Field**.

If an user clicks **OK**, the function returns contents on **Text Field**, otherwise, if the user clicks **Cancel**, the function returns **null**.

prompt-example.js

<!DOCTYPE **html**>

<**html**>

<**head**>

<**title**>Prompt Dialog Box</**title**>

<**script** type="text/javascript">

**function** **testPromptDialog**() {

**var** result = **prompt**("Enter you age:", "20");

**if**(result != null) {

**alert**("Your age is " + result);

}

}

</**script**>

</**head**>

<**body**>

<**h2**>Prompt Dialog Box</**h2**>

<**button** onclick="testPromptDialog()">Click me!</**button**>

</**body**>

</**html**>

# Comparing Alert, Confirm and Prompt popup boxes

Go through these details to understand when to use Alert, Confirm and Prompt boxes and the differences between them.

#### Alert popup Box

* **Description**: Used to show a message or warning .
* **Input Field**: None
* **Buttons**: Only a single “OK” button is provided.
* **When to use?**: When we need to print a warning/message for the user to alert him about his/her action.
* **Example**: An alert box displayed when you are being redirected to another page

#### Confirm popup Box

* **Description**: Used to take confirmation to proceed with an event.
* **Input Field**: None
* **Buttons**: Two botton are provided “OK” and “Cancel”.
* **When to use?**: When we need to take a confirmation from the user to proceed with an event like redirecting to another page etc.
* **Example**: A confirm box displayed by a website to show you notifications with an “OK” and “Cancel” button.

#### Prompt popup Box

* **Description** :Used to take a single input from the user and proceed with event depending upon the input.
* **Input Field** :Single input field(the input in treated as string or null if user does not enter anything).
* **Buttons** :Two buttons are provided “OK” and “Cancel”.
* **When to use?** :When we need to take a confirmation from the user to proceed with an event like redirecting to another page etc.
* **Example** :A prompt box displayed by a website to get your name or age.

# JavaScript Functions

**JavaScript functions** are used to perform operations. We can call JavaScript function many times to reuse the code.

#### **Advantage of JavaScript function**

There are mainly two advantages of JavaScript functions.

1. **Code reusability**: We can call a function several times so it save coding.
2. **Less coding**: It makes our program compact. We don’t need to write many lines of code each time to perform a common task.

## JavaScript Function Syntax

The syntax of declaring function is given below.

function functionName([arg1, arg2, ...argN]){

 //code to be executed

}

JavaScript Functions can have 0 or more arguments.

## JavaScript Function Example

Example of function in JavaScript that does not has arguments.

<!DOCTYPE html>

<html>

<head>

<title> ONE </title>

<script>

function msg()

{

alert("hello! this is message");

}

</script>

</head>

<body>

<input type="button" onclick="msg()" value="call function"/>

</body>

</html>

## JavaScript Function Arguments

We can call function by passing arguments. Let’s see the example of function that has one argument.

<!DOCTYPE html>

<html>

<head>

<title> ONE </title>

<script>

function getcube(number)

{

alert(number\*number\*number);

}

</script>

</head>

<body>

<form>

<input type="button" value="click" onclick="getcube(4)"/>

</form>

</body>

</html>

## Function with Return Value

We can call function that returns a value and use it in our program. Let’s see the example of function that returns value.

<!DOCTYPE html>

<html>

<head>

<title> ONE </title>

<script>

function getInfo(){

return "Are you a student at Zetech University?";

}

document.write(getInfo());

</script>

</head>

<body>

</body>

</html>

# JavaScript Math

The **JavaScript math** object provides several constants and methods to perform mathematical operation. Unlike date object, it doesn't have constructors.

## JavaScript Math Methods

Let's see the list of JavaScript Math methods with description.

|  |  |
| --- | --- |
| **Methods** | **Description** |
| [abs()](https://www.javatpoint.com/javascript-math-abs-method) | It returns the absolute value of the given number. |
| [acos()](https://www.javatpoint.com/javascript-math-acos-method) | It returns the arccosine of the given number in radians. |
| [asin()](https://www.javatpoint.com/javascript-math-asin-method) | It returns the arcsine of the given number in radians. |
| [atan()](https://www.javatpoint.com/javascript-math-atan-method) | It returns the arc-tangent of the given number in radians. |
| [cbrt()](https://www.javatpoint.com/javascript-math-cbrt-method) | It returns the cube root of the given number. |
| [ceil()](https://www.javatpoint.com/javascript-math-ceil-method) | It returns a smallest integer value, greater than or equal to the given number. |
| [cos()](https://www.javatpoint.com/javascript-math-cos-method) | It returns the cosine of the given number. |
| [cosh()](https://www.javatpoint.com/javascript-math-cosh-method) | It returns the hyperbolic cosine of the given number. |
| [exp()](https://www.javatpoint.com/javascript-math-exp-method) | It returns the exponential form of the given number. |
| [floor()](https://www.javatpoint.com/javascript-math-floor-method) | It returns largest integer value, lower than or equal to the given number. |
| [hypot()](https://www.javatpoint.com/javascript-math-hypot-method) | It returns square root of sum of the squares of given numbers. |
| [log()](https://www.javatpoint.com/javascript-math-log-method) | It returns natural logarithm of a number. |
| [max()](https://www.javatpoint.com/javascript-math-max-method) | It returns maximum value of the given numbers. |
| [min()](https://www.javatpoint.com/javascript-math-min-method) | It returns minimum value of the given numbers. |
| [pow()](https://www.javatpoint.com/javascript-math-pow-method) | It returns value of base to the power of exponent. |
| [random()](https://www.javatpoint.com/javascript-math-random-method) | It returns random number between 0 (inclusive) and 1 (exclusive). |
| [round()](https://www.javatpoint.com/javascript-math-round-method) | It returns closest integer value of the given number. |
| [sign()](https://www.javatpoint.com/javascript-math-sign-method) | It returns the sign of the given number |
| [sin()](https://www.javatpoint.com/javascript-math-sin-method) | It returns the sine of the given number. |
| [sinh()](https://www.javatpoint.com/javascript-math-sinh-method) | It returns the hyperbolic sine of the given number. |
| [sqrt()](https://www.javatpoint.com/javascript-math-sqrt-method) | It returns the square root of the given number |
| [tan()](https://www.javatpoint.com/javascript-math-tan-method) | It returns the tangent of the given number. |
| [tanh()](https://www.javatpoint.com/javascript-math-tanh-method) | It returns the hyperbolic tangent of the given number. |
| [trunc()](https://www.javatpoint.com/javascript-math-trunc-method) | It returns an integer part of the given number. |

**Example: A program to Calculate 3 to power 4:**

<!DOCTYPE html>

<html>

<head>

<title> ONE </title>

<script>

document.write("Power=",Math.pow(3,4))

</script>

</head>

<body>

</body>

</html>

# JavaScript Events

The change in the state of an object is known as an **Event**. In html, there are various events which represents that some activity is performed by the user or by the browser. When [javascript](https://www.javatpoint.com/javascript-tutorial) code is included in [HTML](https://www.javatpoint.com/html-tutorial), js react over these events and allow the execution. This process of reacting over the events is called **Event Handling**. Thus, js handles the HTML events via **Event Handlers**.

**For example**, when a user clicks over the browser, add js code, which will execute the task to be performed on the event.

Some of the HTML events and their event handlers are:

## Mouse events:

|  |  |  |
| --- | --- | --- |
| **Event Performed** | **Event Handler** | **Description** |
| click | onclick | When mouse click on an element |
| mouseover | onmouseover | When the cursor of the mouse comes over the element |
| mouseout | onmouseout | When the cursor of the mouse leaves an element |
| mousedown | onmousedown | When the mouse button is pressed over the element |
| mouseup | onmouseup | When the mouse button is released over the element |
| mousemove | onmousemove | When the mouse movement takes place. |

## Keyboard events:

|  |  |  |
| --- | --- | --- |
| **Event Performed** | **Event Handler** | **Description** |
| Keydown & Keyup | onkeydown & onkeyup | When the user press and then release the key |

## Form events:

|  |  |  |
| --- | --- | --- |
| **Event Performed** | **Event Handler** | **Description** |
| focus | onfocus | When the user focuses on an element |
| submit | onsubmit | When the user submits the form |
| blur | onblur | When the focus is away from a form element |
| change | onchange | When the user modifies or changes the value of a form element |

## Window/Document events

|  |  |  |
| --- | --- | --- |
| **Event Performed** | **Event Handler** | **Description** |
| load | onload | When the browser finishes the loading of the page |
| unload | onunload | When the visitor leaves the current webpage, the browser unloads it |
| resize | onresize | When the visitor resizes the window of the browser |

## Click Event Example:

<html>

<head>

<script language="Javascript" type="text/Javascript">

function clickevent()

{

document.write("ZETECH UNIVERSITY");

}

</script>

</head>

<body>

<form>

<input type="button" onclick="clickevent()" value="Where do yu study?"/>

</form>

</body>

</html>

## Mouse Over Event Example:

<html>

<head>

</head>

<script language="Javascript" type="text/Javascript">

function mouseoverevent()

{

alert("I am a student at Zetech University");

}

</script>

<body>

<h1> Javascript Events </h1>

<p onmouseover="mouseoverevent()"> Keep cursor over me</p>

</body>

</html>

## Focus Event Example

<html>

<head></head>

<script>

function focusevent()

{

document.getElementById("input1").style.background=" aqua";

}

</script>

<body>

<h2> Enter something here</h2>

<input type="text" id="input1" onfocus="focusevent()"/>

</body>

</html>

## Key Down Event Example

<html>

<head></head>

<script>

function keydownevent()

{

document.getElementById("input1");

alert("Pressed a key");

}

</script>

<body>

<h2> Enter something here</h2>

<input type="text" id="input1" onkeydown="keydownevent()"/>

</body>

</html>

## Load Event Example

<html>

<head></head>

<script>

<!--

document.write("The page is loaded successfully");

//-->

</script>

</br>

<body onload="window.alert('Page successfully loaded');">

</body>

</html>