WELCOME

Group Members

- M.Tayyab
- Hassan Dar
- Saddam Butt
- Bilal Khan

TOPICS

- Arrays
- Declaring and Allocating Array
- Types of Array
- Array Methods
- Array Attributes

Arrays

- `Array` **inherits from** `Object`**.**
- **Indexes are converted to strings and used as names for retrieving values.**
- **Very efficient for sparse arrays.**
- **Not very efficient in most other cases.**
- **One advantage: No need to provide a length or type when creating an array.**

Declaring and Allocating Arrays

- **JavaScript arrays are `Array` objects.**
- **Creating new objects using the `new` operator is known as creating an instance or instantiating an object**
- **Operator `new` is known as the dynamic memory allocation operator**

Declare + Initialize Arrays

var array_name = [“a” , ”b” , ”c”];

Using the JavaScript Keyword new

```
//Declaration
var sadi = new Array();
//initialization
var sadi = new Array("Butt","Dar","Loan");
var chiler = new Array("Umair","Ateeb","Bilal Khan");
```

Types of Array

- **Associative Array**
- **Index Array**

Associative Array

(keys and values)

```
var  person = [];
person [“ firstName "]  =  “Hassan";
person [“ lastName "]    =  "Dar";
```

OR

```
<script type="text/javascript">
    var ary = {First_name:"Hassan",Last_Name:"Dar"};

    document.write(ary["First_name"]);
</script>
```

Hassan

Indexing Array

Array elements are accessed using their index number:

```
var ary = [“A” , “B” , “C” , “D” , “E” , “F”];
document.write(ary[4]);          //output : E
```

```
<script type="text/javascript">
    var ary = ["A","B","C","D","E"];
    document.write(ary[2]);
</script>
```

C

USING LOOP

```
<script type="text/javascript">
    var ary = ["A","B","C","D","E"];
    for(var i=0; i<ary.length;i++)
    document.write("<br />"+ary[i]);
</script>
```

```
<script type="text/javascript">
    var ary = ["A","B","C","D","E"];
    for(i in ary)
    document.write("<br />"+ary[i]);
</script>
```

A
B
C
D
E

Array Methods

- Concat
- Join
- Push
- Pop
- UnShift
- Shift
- Sort
- Slice
- Splice

CONCAT

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

    var ary1 = new Array("Sadi");
    var ary2 = new Array("Butt");
    var ary3 = ary1.concat(ary2);
    document.write(ary3);
</script>
```

Sadi,Butt

Join

The **join()** method also joins all array elements into a string.
It behaves just like toString(), but you can specify the separator:

```
<script type="text/javascript">
    var ary = ["A", "B", "C", "D"];
    document.write(ary.join("*"));
</script>
```

A*B*C*D

PUSH

The push() method adds a new element to an array (at the end):

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

    var ary = new Array("Sadi Butt","Bilal Khan",
                        "Hassan Dar","M.tayyab");
    ary.push("Juni");
    document.write(ary);
</script>
```

Sadi Butt,Bilal Khan,Hassan Dar,M.tayyab,Juni

POP

The pop() method removes the last element from an array:

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

    var ary = new Array("Sadi Butt","Bilal Khan",
                        "Hassan Dar","M.tayyab");
    ary.pop();
    document.write(ary);
</script>
```

Sadi Butt,Bilal Khan,Hassan Dar

UNSHIFT

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

    var ary = new Array("Sadi Butt","Bilal Khan",
                        "Hassan Dar","M.tayyab");
    ary.unshift("Juni");
    document.write(ary);
</script>
```

Juni,Sadi Butt,Bilal Khan,Hassan Dar,M.tayyab

SHIFT

```
<script type="text/javascript">
    var ary = new Array("Sadi Butt","Bilal Khan",
                        "Hassan Dar","M.tayyab");
    ary.shift();
    document.write(ary);
</script>
```

Bilal Khan,Hassan Dar,M.tayyab

SORT

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

    var ary = new Array("D","B","A","C","E");
    ary.sort();
    document.write(ary);
</script>
```

A,B,C,D,E

REVERSE

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

    var ary = new Array("D","B","A","C","E");
    ary.sort();
    document.write(ary);
    ary.reverse();
    document.write(ary);
</script>
```

A,B,C,D,E

E,C,A,B,D

SLICE...

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

    var ary = new Array("A","B","C","D","E");
    document.write(ary.slice(2));
</script>
```

C,D,E

SLICE

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

    var ary = new Array("A","B","C","D","E");
    document.write(ary.slice(2,4));
</script>
```

C,D

SPLICE

The **splice()** method can be used to add new items to an array and also Remove items from Array:

Add Values

```
<script type="text/javascript">
    var ary = ["A","B","C","D","E"];
    ary.splice(2,0,"S","H");
    document.write(ary);
</script>
```

A,B,S,H,C,D,E

Remove Values

```
<script type="text/javascript">
    var ary = ["A","B","C","D","E"];
    ary.splice(2,2);
    document.write(ary);
</script>
```

A,B,E

ADD & REMOVE

```
<script type="text/javascript">
    var ary = ["A","B","C","D","E"];
    ary.splice(2,2,"S","H");
    document.write(ary);
</script>
```

A,B,S,H,E

Array Attributes

- length
- indexOf
- typeOf

M

- Changing Elements
- Deleting Elements

LENGTH.....

- **The length property provides an easy way to append new elements to an array without using the push() method.**

```
<script type="text/javascript">
    var ary = ["Hassan","Dar","Saddam"];
    ary[ary.length]="Butt";
    document.write(ary);
</script>
```

Hassan,Dar,Saddam,Butt

LENGTH

```
<script type="text/javascript">
    var ary = ["A","B","C","D","E","F","G","H"];
    document.write(ary.length);
</script>
```

8

INDEXOF

```
<script type="text/javascript">
    var ary = ["A","B","C","D","E"];
    document.write(ary.indexOf("D"));
</script>
```

3

Typeof

A common question is: How do I know if a variable is an array? The problem is that the JavaScript operator **typeof** returns "object":

```
<script type="text/javascript">
    var ary = ["A", "B", "C", "D"];
    document.write(typeof ary);
</script>
```

object

Changing Elements

```
<script type="text/javascript">
    var ary = ["A", "B", "C", "D"];
    ary[0] = "H";
    document.write(ary);
</script>
```

H,B,C,D

Deleting Elements

```
delete array[number]
```

- **Removes the element, but leaves a hole in the numbering.**

```
array.splice(number, 1)
```

- **Removes the element and renumbers all the following elements.**

Deleting Elements

```
Ary = ['a', 'b', 'c', 'd` ,`e’];
delete Ary[1];
['a',undefined, 'c', 'd` ,’e’]
```

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

    var ary = new Array("A","B","C","D","E");
    delete ary[1];
    document.write(ary);
</script>
```

A,,C,D,E

THANKS FOR WATCHING