

آشنایی با جاوا اسکریپت

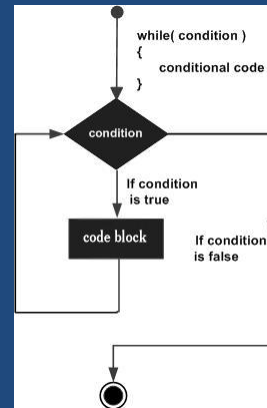
ساختارهای تکرار در جاوا اسکریپت

- ساختار تکرار while
- ساختار تکرار do while
- ساختار تکرار for

ساختار تکرار while

```
while (condition) {
    statements;
}
```

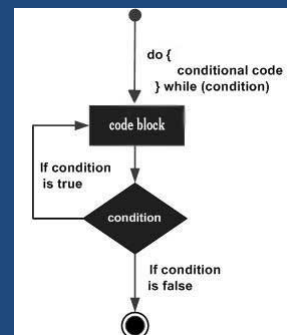
JS



ساختار تکرار do while

```
do {
    statements;
} while (condition);
```

JS



ساختار تکرار for

```
for (begin; condition; step) {  
    statements;  
}
```

توضیحات در جاوا اسکریپت

```
// comment
```

```
/* comment (avoid using this form...) */
```

تعریف آرایه ها

- آرایه یک متغیر مخصوص بوده که می تواند بیش از یک مقدار را همزمان در خود نگه دارد.

```
var array_name = [item1, item2, ...];
```

```
var aValues = new Array();
```

مقداردهی آرایه ها

```
var aColors = ["red", "green", "blue"];
```

```
var aColors = new Array("red", "green", "blue");
```

مقداردهی آرایه ها

```
var stringArray = ["one", "two", "three"];

var numericArray = [1, 2, 3, 4];

var decimalArray = [1.1, 1.2, 1.3];

var booleanArray = [true, false, false, true];

var mixedArray = [1, "two", "three", 4];
```

```
var stringArray = new Array();
stringArray[0] = "one";
stringArray[1] = "two";
stringArray[2] = "three";
stringArray[3] = "four";

var numericArray = new Array(3);
numericArray[0] = 1;
numericArray[1] = 2;
numericArray[2] = 3;

var mixedArray = new Array(1, "two", 3, "four");
```

آرایه

- یک آرایه نیز می تواند **Object** باشند. به همین دلیل شما می توانید در یک آرایه مشخص، متغیرهایی با نوع های مختلف داشته باشید. حتی می توان در خانه های یک آرایه از یک شی استفاده نمود.

```
var student1 = {Name : "ali", FName : "panahi", age:22, BDate:"1994/10/10"}
var student2 = {fName : "sara", FName : "ahmadi", age:20, BDate:"1996/04/10"}
var student3 = {Name : "reza", FName : "shojaee", age:21, BDate:"1995/06/10"}
var students = [student1, student2, student3]
```

```
alert(student1["Name"])
alert(students[1].firstName)
```

آرایه

بدست آوردن تعداد مقادیر درون یک آرایه در جاوا اسکریپت

```
var x = cars.length;
```

دسترسی به آخرین عنصر آرایه

```
var last = fruits[fruits.length-1];
```

مرتب سازی آرایه ها

- از دو تابع برای مرتب سازی عناصر آرایه استفاده میشود.
- تابعی به نام `reverse()` برای مرتب سازی عکس آرایه استفاده می شود.
- تابعی به نام `sort()` عناصر آرایه را به صورت صعودی بر حسب مقادیرشان مرتب می کند.

توابع (Function)

- تعریف تابع

```
function functionName()  
{  
  code to be executed  
}
```

متغیری که در داخل یک function تعریف می‌شود، تنها داخل همان تابع قابل استفاده است

مثال:

- ```
function welcome()
{
 document.write ("Just For Fun")
}
```

## توابع

- یک تابع می تواند، چندین متغیر را به عنوان پارامتر ورودی دریافت کند. پارامترهای یک تابع را باید در هنگام تعریف تابع، در پرانتز مقابل نام آن تعیین کرد، که پارامترها را با کاما از هم جدا می کنیم .
- در هنگام فراخوانی یک تابع که دارای پارامتر است، باید در پرانتز مقابل نام آن، مقادیر متناظر با پارامترهایش را اعلام کرد. که به این مقادیر آرگومان گفته می شود. این آرگومان باید از لحاظ تعداد و نوع کاملاً یکسان با پارامترهای تعریف شده در تابع باشند .
- ساختار تعریف تابع به صورت زیر میباشد:

```
function Name (Parametr 1 , Parametr 2 , ...)
{
Code
}
```

## توابع

- یک تابع می تواند پس از انجام دستورات در نظر گرفته شده برای آن، مقداری را به عنوان خروجی به نقطه ای که از آن فراخوانی شده است، باز گرداند.
- برای تعیین مقدار بازگشتی یک تابع از دستور `return` استفاده کرده، که مقدار خروجی را در پرانتز مقابل آن دستور به شکل زیر تعریف می کنیم.

**return (مقدار بازگشتی) ;**

- نکته : مقدار بازگشتی ، می تواند یک رشته یا عدد ، یک متغیر و یا یک عبارت محاسباتی باشد.