

(JavaScript)

تعريف ال JavaScript :

- هي لغة برمجة عالية المستوى، تستخدم أساساً في متصفحات الويب لإنشاء صفحات أكثر تفاعلية. تطورها حالياً لجنة TC39 التي تديرها منظمة EACMA للمعايير.
- جافا سكريبت هي لغة برمجة صممت في الأصل لإضافة مؤثرات وتفاعلات لصفحات الويب، ويعنى بالتفاعلات كل ما يتم عمله أو إجراؤه على مستوى المتصفح من دون الحاجة لإعادة تحميل الصفحة.

اولا:المتغيرات Variables:

هو عنوان للتخزين في الذاكرة مرتبط باسم رمزي و قد يحتوي على عدد معلوم أو غير معلوم من المعلومات، أو ما يطلق عليه اسم «قيمة». اسم المتغير هو الطريقة الاعتيادية للرجوع إلى القيمة المخزنة، هذا الفصل ما بين اسم المتغير ومحتواه يتيح استخدام الاسم بمعزل عن القيمة التي يمثلها. يمكن للاسم الرمزي في شيفرة المصدر أن يُربط بقيمة في وقت التنفيذ، وبالتالي يمكن أن تتغير هذه القيمة خلال تنفيذ البرنامج .

امثله علي المتغيرات في ال (JS):

```
let name = "أحمد";  
let message = "!مرحبًا بك";
```

```
let age = 25;  
let price = 9.99;
```

ثانياً: المصفوفات Arrays:

هي هيكل بيانات يستخدم لتخزين وتنظيم مجموعة من القيم المرتبطة. تعريف المصفوفة يتضمن إنشاء متغير يحمل اسم المصفوفة وتعيينه بقيمة مجموعة من العناصر المرتبة في ترتيب معين.

ملحوظة:

يمكن إنشاء مصفوفة في JavaScript باستخدام الأقواس المربعة [] وتحتوي العناصر داخل المصفوفة على فواصل.

س: كيف يمكن الوصول إلى العناصر في المصفوفة؟

يمكن الوصول إلى عناصر المصفوفة باستخدام فهرسها. تبدأ الفهارس من الصفر، ويمكن استخدام الفهرس للوصول إلى قيمة محددة في المصفوفة.

أمثله علي المصفوفات في ال(JS):

١. مصفوفة بعناصر محددة:

```
let numbers = [1, 2, 3, 4, 5];  
let names = ["أحمد", "محمد", "سارة", ""];  
let empty = [];
```

٢. مصفوفة فارغة:

```
let emptyArray = [];
```

٣. المصفوفة يمكن أن تحتوي على أنواع بيانات مختلفة:

```
let mixedArray = [1, "أحمد", true, null];
```

٤. يمكن الوصول إلى عنصر في المصفوفة باستخدام فهرسه:

```
let numbers = [10, 20, 30, 40, 50];  
let firstNumber = numbers[0]; //  
10العنصر الأول:  
let thirdNumber = numbers[2]; //  
30العنصر الثالث:
```

٥. يمكن تعديل قيمة عنصر في المصفوفة عن طريق تحديد فهرسه وتعين القيمة الجديدة:

```
let numbers = [10, 20, 30, 40, 50];  
numbers[1] = 25; // تغيير القيمة الثانية إلى  
25
```

٦. يمكن استخدام خاصية length لمعرفة عدد العناصر في المصفوفة:

```
let numbers = [10, 20, 30, 40, 50];  
let arrayLength = numbers.length; //  
عدد العناصر: 5
```

ملحوظة:

هناك ما يعرف بالـ map لتظهر المصفوفه بشكل افضل مثال:

```
1 var trainees=["assem","mohamed","ahmed","omar"]  
2 trainees.map((trainee,index)=> console.log(index+"_"+trainee))
```

ثالثا: الكائنات Objects:

- هي هيكل بيانات مرن وقوي يستخدم لتخزين وتنظيم البيانات. يتم تمثيل الكائنات في JavaScript عن طريق مجموعة من الخصائص (Properties) والوظائف (Methods) المرتبطة بها.

- تعريف الكائن في JavaScript ينطوي على إنشاء متغير وتعيينه بقيمة كائن فارغ أو ببعض الخصائص والوظائف. يمكن إضافة وحذف وتعديل الخصائص والوظائف في الكائن بشكل ديناميكي.

امثله علي المصفوفات في ال(JS):

١. تعريف كائن فارغ:

```
let person = {};
```

٢. تعريف كائن مع خصائص محددة:

```
let person = {  
  name: "أحمد",  
  age: 30,  
  gender: "ذكر"  
};
```

٣. الوصول إلى خصائص الكائن:

يمكن الوصول إلى الخصائص في الكائن باستخدام العمليات المنفذة (.) أو عن طريق الفهرس. على سبيل المثال:

```
let person = {  
  name: "أحمد",  
  age: 30,  
  gender: "ذكر"  
};  
  
console.log(person.name); // أحمد  
console.log(person["age"]); // 30
```

٤. الوظائف في الكائن:

يمكن إضافة الوظائف كخصائص في الكائن. تسمى هذه الوظائف بوظائف الكائن (Object Methods). على سبيل المثال:

```
let person = {  
  name: "أحمد",  
  age: 30,  
  sayHello: function() {  
    console.log("مرحبًا");  
  }  
};  
  
person.sayHello(); // مرحبًا
```

رابعاً: الدوال Functions:

- هي قطعة من الكود تقوم بتنفيذ مجموعة من العمليات المحددة عندما تُستدعى. الدالة تسمح بتجميع الكود وإعادة استخدامه بشكل فعال، وتسهل البرمجة وتحسن قابلية الصيانة للبرامج.

- تعريف الدالة في JavaScript يتضمن إنشاء كتلة من الكود بواسطة استخدام الكلمة المحجوزة "function"، مع تحديد اسم للدالة وقائمة بالمعاملات (إن وجدت) داخل الأقواس المستديرة، وكتلة الكود المراد تنفيذها داخل الأقواس السحابية. يمكن استدعاء الدالة في أي مكان في البرنامج باستخدام اسم الدالة متبوعاً بأقواس مستديرة.

امثله علي الدوال في ال(JS):

مثال ١:

```
function name_function(){  
  console.log("hello worled")  
}  
  
name_function()
```

مثال ٢:

```
function sub(x,y){  
  console.log(x-y)  
}  
  
sub(8,4)
```

مثال ٣:

```
<button onclick="greet()">اضغط هنا</button>

<script>
function greet() {
  console.log("!مرحبًا بك");
}
</script>
```

مثال ٤:

```
<input type="text" id="nameInput">
<button onclick="greet()">مرحبًا</button>

<script>
function greet() {
  var name = document.getElementById("nameInput").value;
  console.log("مرحبًا " + name + "!");
}
</script>
```
