

Creación de funciones

Una de las tareas más importantes del desarrollo con JavaScript es crear código que otro código pueda reutilizar. Para hacer esto, se debe organizar el código en funciones que realicen tareas específicas. Una función es una serie de instrucciones combinadas en un solo bloque al que se le asigna un nombre. El código en el bloque se puede ejecutar haciendo referencia a ese nombre.

Definición de funciones

Las funciones se definen mediante la palabra clave `function` seguida de un nombre que describe el uso de la función, una lista de cero o más argumentos entre paréntesis y un bloque de una o más declaraciones de código entre `{}` corchetes.

```
function myFunction(){  
  print("Hola Mundo");  
}
```

Para ejecutar el código incluido en `myFunction()`, todo lo que se necesita hacer es llamar o invocar a la función.

```
myFunction();
```

Parámetros de las funciones

Con frecuencia, se deben pasar valores específicos a las funciones que se utilizarán. Los valores se pasan delimitados por comas. La definición de la función necesita una lista de nombres de variables entre paréntesis () que coincidan con el número que se pasa.

```
function saludo(nombre, ciudad){  
  print("Hola " + nombre);  
  print("Cómo está el clima en " + ciudad);  
}
```

Para llamar/invocar a la función saludo(), se debe pasar el valor **nombre** y un valor para **ciudad**. El valor puede ser directo o una variable previamente definida.

```
var nombre = "Juan Pablo";  
saludo(nombre, "Venecia");
```

Retorno de las funciones

Con frecuencia, las funciones necesitan devolver un valor al código de llamada. Agregar la palabra clave `return` seguida de una variable o valor devuelve ese valor de la función.

```
function formatoSaludo(nombre, ciudad)
{
    var retStr = "";
    retStr += "Hola " + nombre + "\n";
    retStr += "Bienvenido a " + ciudad + "!";
    return retStr;
}
var saludo = formatoSaludo("Juan Pablo", "Venecia");
print(saludo);
```

Salida ->

```
Hola Juan Pablo
Bienvenido a Venecia!
```

Combinación de cadenas

Se pueden combinar varias cadenas usando el signo + o la función concat() en la primera cadena.

```
var palabra1 = "El ";  
var palabra2 = "es ";  
var palabra3 = "\"El ";  
var palabra4 = "mejor \"";  
var sentencia1 = palabra1 + palabra2 + palabra3 + palabra4;  
  
print(sentencia1)
```

Salida -> El es "El mejor "

Trabajar con matrices

El objeto array proporciona un medio para almacenar y manejar un conjunto de otros objetos. Las matrices pueden almacenar números, cadenas u otros objetos de JavaScript. Existen varios métodos para crear matrices de JavaScript.

El primer método define arr y establece el contenido en una sola declaración usando [].

El segundo método crea el objeto arr2 y luego le agrega elementos mediante la asignación de índice directo.

El tercer método crea el objeto arr3 y luego usa la mejor opción para extender matrices: usa el método push() para insertar elementos en la matriz.

```
var arr = ["uno", "dos", "tres"];
```

```
var arr2 = new Array();  
arr2[0] = "uno";  
arr2[1] = "dos";  
arr3[2] = "tres";
```

```
var arr3 = new Array();  
arr3.push("uno");  
arr3.push("dos");  
arr3.push("tres");
```

Longitud de matrices

Para obtener la cantidad de elementos en la matriz, usa la propiedad `length` del objeto `Array`.

```
var semana = ["lunes", "martes", "miércoles", "jueves", "viernes"];  
var numDeItems = semana.length;  
  
print("Cantidad de elementos del array: ", numDeItems)
```

Salida ->

```
Cantidad de elementos del array: 5
```

Índices de matrices

Las matrices son un índice basado en cero, lo que significa que el primer elemento está en el índice 0, y así sucesivamente.

```
var semana = ["lunes", "martes", "miércoles", "jueves", "viernes"];  
var primero = semana[0];  
var ultimo = semana[semana.length-1];  
  
print("Primero: ", primero);  
print("Último: ", ultimo);
```

Salida ->

```
Primero:  lunes  
Último:  viernes
```


Iterar a través de matrices

Es posible iterar a través de una matriz usando for o for/ inloop.

```
var semana = ["lunes", "martes", "miércoles", "jueves", "viernes"];

for (var i=0; i<semana.length; i++)
{
    print(semana[i]);
}
```

Salida ->

```
lunes
martes
miércoles
jueves
viernes
```

```
var semana = ["lunes", "martes", "miércoles", "jueves", "viernes"];

for (dayIndex in semana)
{
    print(semana[dayIndex]);
}
```

Salida ->

```
lunes
martes
miércoles
jueves
viernes
```