

Microsoft

Technology Associate



Worldwide
Technological
Certification.

JavaScript Fundamentals

Examen de Certificación Microsoft MTA 98-382
Modulo 1: Sesion 2 : Variables, DataTypes,
condicionales



Worldwide
Technological
Certification.

Objetivos de la Sesión



- Comprender la propuesta de JavaScript frente a la gestión de variables, tipos de datos y funciones

1. HTTP/2

- Va a permitir descargar simultáneamente distintas definiciones (assets) para un mismo sitio
- El objetivo es demorar menos en cargar un sitio
- Se sugiere entonces que las referencias externas a JS estén en la parte de abajo aún y cuando el código fuente no requiera que el ViewPort haya sido desplegado
- El código JS puede cargarse inmediatamente, de manera asíncrona o diferida

2. Marco de trabajo en JS

- Case Sensitive:
 - `var nombresApellidos`
 - `var fecha = Date()`
- Se anima a utilizar la norma CamelCase:
 - `var nombresClientes`
 - `var fechaContratacion`
 - `var nombresApellidosClientes`
- Objetos y clases deben iniciar con mayúsculas:
 - `var fechaContratacion = Date();`
- Constantes deben escribirse en mayúsculas:
 - `const = VALORDELIVA`

Marco de trabajo en JS

- Se anima a utilizar espacios en blancos para que el código sea legible:
 - `document.body.innerHTML = "<h1>" + fechaContratacion + "</h1>";`
- Agregar punto y coma al final de cada linea (aún y cuando no es una obligación para ejecutar el código)
- Es aconsejable usar comentarios sin abusar de ellos (estándares de codificación)

3. Variables

- `var a = 5;`
- `var b = 4;`
- `var suma = a+b;`
- `alert("la suma de las variables es: "+ suma);`

4. Data Types

- Data Types primitivos:
- Numeric = cualquier representacion numerica
- String = cadenas alfanumericas (puede usarse comillas simples o dobles)
- Boolean = true / false (sin comillas)
- Null = ausencia de valor
- Undefined = creacion de una variable sin asignar ningun valor
- Symbol *
- Demo: Ejercicio 3

5. Asignacion

- Diseñe un formulario que permita ejecutar las siguientes operaciones
- Capture los nombres, apellidos y edad de un vendedor
- Calcule el año de nacimiento a partir de la edad
- Si vendio 10 articulos de tipo azul calcule un bono del 25% sobre la venta total
- La salida de datos tiene que devolver: nombre completo y entre comillas, edad digitada, año de nacimiento, venta total, bono a pagar

6. Operadores

- Asignacion: =
- Aritmeticos: +, -, *, /
- Reglas algebraicas: $a + (b * c)$
- Operadores para abreviación (shorthand) aritmeticas:
 - $a = a + 4$
 - $A += 4$
 - $-=$, $*=$, $/=$
- Operador unario:
 - `var a = 5;`
 - `a++;`
 - `a--;`

6. Operadores

- `var a = 5;`
- `A++;` // en base al valor original se aumenta el valor en 1
- `++a;` //agrega 1 al valor actual

7. Asignacion

- Diseñe una pagina web que utilizando js calcule las siguientes operaciones:
 - Suma
 - Resta
 - Multiplicacion
 - Division
 - Modulo
 - Operadores unarios
 - Concatenacion y operaciones aritmeticas con cadenas

8. Statement Condicional

- `if(condicion_evaluada){`
 - `//codigo a ejecutar en caso que la evaluacion sea verdadera`
- `}`
- `else{`
 - `//codigo a ejecutar en caso que la evaluacion sea falsa`
- `}`
- Operadores de comparación: `>`, `>=`, `<`, `<=`, `!=`, `!==(`, `true`, `!true`

8. Statement Condicional

- Ejemplo: evaluar si un numero es par:
- `if(numero / 2 == 1){`
 - `alert("el numero es par");`
- `}`
- `else{`
 - `alert("el numero es impar");`
- `}`

8. Statement Condicional

- Asignaciones:
- 1. Mejore el código anterior (enfoque de lógica) de manera que pueda servir para evaluar cualquier número
- 2. Codifique una interfaz en HTML para que el usuario pueda escribir el número que desea evaluar
- Diseñe una interfaz que permita evaluar si un número es primo
- Diseñe una interfaz que permita evaluar el mayor de 3 números
- Diseñe una interfaz que permita evaluar el mayor de 4 números