

## M4.250 - Introducción a la programación en JavaScript I aula 1

### PEC 1

Alumno: Ivan Schartun

2.-Comprobad si estos fragmentos de código JavaScript muestran un mensaje de error cuando se ejecutan. Decid qué mensaje muestran y por qué:

**A.**

```
var x;  
try {  
  x = y - 1;  
}  
catch(err) { console.log(err.name); }
```

// Al ejecutar el fragmento se generaría por consola el siguiente error: **ReferenceError**

Esto se debe a que se está tratando de ejecutar una operación que involucra una variable que no ha sido definida por tanto no existe una referencia a la misma. Una referencia no válida.

**B.**

```
var x=1;  
try {  
  x.toLowerCase();  
}  
catch(err) { console.log(err.name); }
```

// Al ejecutar el fragmento se generaría por consola el siguiente error: **TypeError**

Esto se debe a que la variable x tiene un valor numérico 1. Aun cuando JS es un lenguaje de verificación de tipos leves no se podrá aplicar el método dentro del "try" que pide devolver un valor en minúsculas de la cadena a la cual realiza la llamada. No es de un tipo válido.

**C.**

```
var x=0.9999;  
try {  
  console.log(x.toFixed(1000));  
}  
catch(err) { console.log(err.name); }
```

// Al ejecutar el fragmento se generaría por consola el siguiente error: **RangeError**

Esto se debe a que el método dentro del "try" pide aplicar un valor no contemplado entre el rango que maneja el parámetro (digits) que puede ser entre 0 y 20 (El número de dígitos que debe aparecer después del punto decimal). Está fuera de su rango válido

D.

```
try {  
  varr x=1;  
}  
catch(err) { console.log(err.name); }
```

// Al ejecutar el fragmento se generaría por consola el siguiente error: **SyntaxError**

Esto se debe a que dentro del “try” se pide asignar a un valor x, usando una palabra no válida en JS para la declaración de variables ( palabra no válida “varr”). Para crear una variable se debería usar la palabra “var”.