**Apuntes entorno servidor**

# **Objetos predefinidos de javascript**

## **String:**

| charAt() | Devuelve el carácter especificado por la posición que se indica entre paréntesis. |
| --- | --- |
| charCodeAt() | Devuelve el Unicode del carácter especificado por la posición que se indica entre paréntesis. |
| concat() | Une una o más cadenas y devuelve el resultado de esa unión. |
| indexOf() | Devuelve la posición de la primera ocurrencia del carácter buscado en la cadena. |

| lastIndexOf() | Devuelve la posición de la última ocurrencia del carácter buscado en la cadena. |
| --- | --- |
| match() | Busca una coincidencia entre una expresión regular y una cadena y devuelve las  coincidencias o null si no ha encontrado nada. |
| replace() | Busca una subcadena en la cadena y la reemplaza por la nueva cadena  especificada. |

| search() | Busca una subcadena en la cadena y devuelve la posición dónde se encontró. |
| --- | --- |
| slice() | Extrae una parte de la cadena y devuelve una nueva cadena. |
| split() | Divide una cadena en un array de subcadenas. |
| substr() | Extrae los caracteres de una cadena, comenzando en una determinada posición y con el número de caracteres indicado. |
| substring() | Extrae los caracteres de una cadena entre dos índices especificados. |
| toLowerCase() | Convierte una cadena en minúsculas. |
| toUpperCase() | Convierte una cadena en mayúsculas.. |

## **Math:**

| E | Devuelve el número Euler (aproximadamente 2.718). |
| --- | --- |
| LN2 | Devuelve el logaritmo neperiano de 2 (aproximadamente 0.693). |
| LN10 | Devuelve el logaritmo neperiano de 10 (aproximadamente 2.302). |
| PI | Devuelve el número PI (aproximadamente 3.14159) |
| SQRT2 | Devuelve la raíz cuadrada de 2 (aproximadamente 1.414). |

| abs(x) | Devuelve el valor absoluto de x. |
| --- | --- |
| acos(x) | Devuelve el arcocoseno de x, en radianes. |
| cos(x) | Devuelve el coseno de x (x está en radianes). |
| ceil(x) | Devuelve el número x redondeado al alta hacia el siguiente entero. |
| floor(x) | Devuelve el número x redondeado a la baja hacia el anterior entero. |

| max(x1,x2,...,x3) | Devuelve el número más alto de los que se pasan como parámetros. |
| --- | --- |
| min(x1,x2,...,x3) | Devuelve el número más bajo de los que se pasan como parámetros. |
| pow(x) | Devuelve el resultado de x elevado a y. |
| random() | Devuelve un número al azar entre 0 y 1. |
| round(x) | Redondea x al entero más próximo. |

## **Number:**

| MAX\_VALUE | Devuelve el número más alto disponible en JavaScript. |
| --- | --- |
| MIN\_VALUE | Devuelve el número más pequeño disponible en JavaScript. |
| NEGATIVE\_INFINITY | Representa a infinito negativo (se devuelve en caso de overflow). |
| POSITIVE\_INFINITY | Representa a infinito positivo (se devuelve en caso de overflow). |

| toFixed(x) | Formatea un número con x digitos decimales después del punto decimal. |
| --- | --- |
| toPrecision(x) | Formatea un número a la longitud x. |
| toString() | Convierte un objeto Number en una cadena.  Si se pone 2 como parámetro se mostrará el número en binario.  Si se pone 8 como parámetro se mostrará el número en octal.  Si se pone 16 como parámetro se mostrará el número en hexadecimal |

# **Expresiones Regulares**

# **Parametro rest**

A partir de ECMA6 en una función se permite que el último argumento que se le pasa, sea especial y se denota con tres puntos suspensivos. Eso nos indica que este último argumento, recibirá todos los parámetros de más de la función en forma de array.

function ejemplo (param1, param2, ...restParams) {

console.log(restParams);

}

ejemplo('a','b','c','d','e','f');

//el resultado es el array ["c", "d", "e", "f"]

Por tanto permite tratar todos los argumentos como si fueran un arrray y aplicar métodos de dicha estructura.

# **Metodos array:**

