Es momento de hablar de las funciones de flecha que es algo nuevo que se incorporó en el ECMAScript 6.

Tiene bastantes bondades de las funciones de flecha y deberíamos de utilizarlas hasta donde sean posibles, pero ya vamos a ver cómo funciona.

Voy a crearme varias funciones normales y mostrarle al mostrarles a ustedes como contrapartida una función de flecha para que tengamos una idea voy a empezar con la palabra reservada LET.

Mi función va a ser igual a una función normal ahora cero llaves y voy a Somorrostro de mandemos un

argumento que ponerle a tipos string y voy a retornar a puntico.

Esta es una función tradicional tal vez ustedes van a decir bueno pero esto no de la manera tradicional

como yo las conozco.

Puede ser que sea más o menos así desde aquí.

Función 2 Función función función 2.

Tal vez están acostumbrados a verlo así.

Y ambas son las mismas sólo que esta es una función asignada a una variable y esto es una función tradicional.

Es exactamente lo mismo pero no estoy hablando de esto.

El beneficio de hacer las funciones de esta manera es que yo puedo definir la función como una constante

y de esta manera no puedo sobreescribir la definición de mi función.

Eso tiene ciertos beneficios pero esto es una función tradicional.

Cómo sería esto en una función de flecha.

Vamos a ver voy a crear una constante para hacerlo exactamente igual aunque también podrían hacerlo

con Left Taiwán o bar en el peor de los casos.

Pero voy a definirlo como una constante para no cambiarla se va a llevar mi función f de flecha esto

va a ser igual y quiero ser exactamente esto mismo.

Entonces voy a poner los paréntesis indican el argumento de tipo string y viene la función de flecha

se dice función de flecha porque esto parece una flecha una flecha gorda o un igual sino de mayor o

menor y abrimos y cerramos llaves.

Luego vendría el router del Ah yo sé que aquí más de uno va a decir bueno no es gran beneficio no es

gran ventaja porque simplemente nos ahorramos escribir fungieron y pusimos la función de flecha.

Eso es básicamente todo si pero las funciones de flecha tienen una característica interesante que es

que si ustedes sólo tienen una línea de código y esa línea de código es lo que ustedes quieren retornar

pueden ahorrarse estas llaves Stubbs pueden ahorrarse estas llaves entonces simplemente pondríamos a

OK.

Esto ya se ve más claro.

Básicamente dice Ok digo Perdón esto es una función de flecha que recibe un argumento y lo que hace

es retornar a tal vez no tiene mucho sentido pero digamos que vas a modo a mayúscula al punto tu o por

lo menos hacer algo igual voy a hacer esto acá punto tú o Perquis ya vemos esas funciones puede ser

un consuelo de mi función mi función parentesis y mi función recibe un argumento voy a poner normal

y la otra va a ser la función de flecha flecha como el procedimiento va a retornar lo que sea que yo

le mande pero percas entonces vamos a ver el resultado.

Grabemos los cambios.

Asegúrese de tener corriendo el TSS menos doble o regresemos al navegador web recargó.

Y aquí tengo normal y Flecha ok exactamente el resultado que yo esperaba pero quiero que noten que la

función de flecha es considerablemente más bueno más sencilla y más compacta aunque también yo podría

deshacerme de esto quitar esos paréntesis paréntesis perdón y debería de funcionar claro aquí me está

marcando un error porque yo había definido en mi tierra como de que no implemente nada que tenga de

tipo n.

De esta manera yo no sabría si esto es un número si esto es un extraño un objeto entonces es bueno acostumbrarse

siempre a poner el tipado de datos.

Ok entonces voy a borrar eso.

Sigamos haciendo más ejercicios para que esto quede un poco más claro.

Voy a crearme ahora otra función que me sirva para sumar dos números.

Vamos a ver constante poner sumar sumar n sumar normal.

Esto va a ser a una función que recibe a tipo ver Koma recibe de tipo number ok abrieron llaves y voy

a ser retrón de a más voy a más.

Perfecto ahora de tarea ustedes van a hacer exactamente esto mismo.

O sea van a crearse su función constante sumar F Flecha igual y hagan ustedes la implementación de esta

función de fecha en este momento del intento y al final ustedes hagan el consuelo y manden los números

para ver si los suma correctamente.

Exactamente como hicimos la prueba aquí arriba.

Pero intenten hacer ustedes la implementación de la función de flecha en estos momentos.

No se preocupen si no lo pueden hacer en instantes voy a hacerlo yo.

Pero intenten hacerlo.

Eso sería un buen reforzamiento.

Entonces pasa el video en este instante

ok lo lograron hacer.

Pudieron convertir esto en su función de flecha.

Vamos a ver como se me ocurre a mí que lo podríamos resolver.

Voy a poner paréntesis ahora y sólo si hago la función de flecha aquí.

Estos son los argumentos que yo tengo que enviar.

Estoy recibiendo a de tipo Number Number y estoy recibiendo Boyé de tipo number.

Adicionalmente yo podría hacer abrir las llaves y poner Berroterán pero como lo único que quiero hacer

es el mismo Richard.

Entonces puedo obviar esas llaves y sólo poner a más.

Vamos a probar si esto funciona vamos a colocar en consolas de sumar de sumar n mandamos los argumentos

de 5,5 debería dar 10 y aquí 10 más 10 debería dar 20.

Voy a grabar los cambios regresar a navegador web recargó y aquí tenemos el 10 y el 20.

Genial.

Voy a hacer un pequeño paréntesis para explicarles algo.

Si nosotros miramos el PP apuntó J.V déjeme colocarlo aquí en un lado para que lo podamos ver de forma

comparativa mi función es una función tradicional y aquí tengo ratón tuvo pequeñas.

Y si miramos la segunda como la transformamos el estándar del ECMAScript 5 poco más Panchito van a notar

que es exactamente la misma función sólo arreglarla tabulación un poco son idénticas son exactamente

iguales pero si nosotros cambiáramos el estándar por ejemplo pusiéramos aquí en el 10 config puntos

Jayson y aquí vamos a apuntar a la versión 6 grados los cambios acá déjenme cerrar este archivo voy

a hacer cualquier modificación grabar los cambios vamos a ver que ahora si lo dejo literalmente igual

porque realmente no estamos haciendo nada propio de Skype.

Esto es puramente de JavaScript y por eso si yo lo compiló al ECMAScript 6 lo deja exactamente igual

como yo lo dejé pero obviamente nosotros queremos apunta a el estándar del ECMAscript 5 porque es más

compatible con todos los navegadores web y cerramos esto bien.

Entonces ahora hagamos un ejercicio donde una función tradicional me dé problemas pero lo resolvamos

o lo vamos a resolver con una función de flecha.

Podemos borrar estos casos Locks ya no lo necesito.

Voy a crearme un nuevo objeto llamado Jock que un objeto que tiene su nombre el nombre va a ser Jock

Koma y creemos dentro un método.

Ese método se va a llamar es más que eso.

Es un método común y corriente aquí adentro lo único que quiero hacer es poner un consultor que diga

utilizando Vartox el nombre de este caso el nombre de este objeto disputo nombre seguido de más que

con tres signos de admiración grabo los cambios vamos a utilizar este objeto aquí adentro de la función

después de haberlo declarado llamemos Jock.

Punto es más grave los cambios.

Regreso al navegador web recargó y aquí tengo es más nada del otro mundo es exactamente lo que yo esperaba.

Veremos qué pasaría si ahora cambiamos esto un poquito esta línea yo no la voy a modificar la voy a

dar igual aquí adentro voy a quedarme un set out set time out definirlo aquí nada más para que lo escriba

punto y coma y el septima es una función que me permite a mí ejecutar un Kovac o una función en determinado

tiempo.

En este caso el handle sería una función normal.

Voy a poner Quick function paréntesis hicieron llaves.

Después de estas llaves voy a poner una coma y quiero que se ejecute en un segundo después.

Así funciona la función y cuantos milésimas de segundo quieren ustedes ejecutar dicha función.

Eso es todo lo que hice.

Ahora tomemos nuestro consumo y coloquemos dentro del set time out.

Aquí ustedes ya van a ver algo interesante ya talves que me está diciendo aquí vas a tener un problema

y me dice quedéis está marcado como de tipo veni y podríamos tener ciertos problemas con el CDIS pero

a mí no me importa voy a grabar los cambios recuerden que no importa si tienen errores de script siempre

va a generar su versión de JavaScript.

Aquí pueden observar ustedes que ya lo hizo entonces si yo regreso al navegador web recargó al segundo

tengo on defined Smash y eso fue una ventaja de que esta ya se estaba quejando.

Y efectivamente vemos el resultado que no es el que esperábamos.

Por qué.

Porque a pesar que dice debería de apuntar al objeto cuando colocamos una función normal estamos cambiando

a lo que apunta el objeto Biz.

En este caso estamos apuntando a la función anonimato invocada si no estoy mal pero obviamente esto

no es el resultado que nosotros queremos.

Veremos que sucedería si cambio esto.

Este Folsom borró la palabra Fonso y pongo la función de flecha.

Así nada más.

El error desaparece.

Eso es una buena señal.

Voy a grabar los cambios.

Regresemos al navegador web descargo y después de un segundo tenemos ok Smash por qué.

Esa es otra característica propia de las funciones de flecha.

Las funciones de flecha no modifican a lo que apunta Díaz.

Es decir aquí afuera en este bloque situado acá en este es.

Díaz apunta a Jock y dentro de la función de flecha como la función de flecha no modifica el diseño.

Sigue apuntando a Jock.

De hecho si ustedes dejan el cursor encima o visor estudio Cott junto con sus extensiones de que ya

les dicen a lo que está apuntando genial esto es algo principal de las funciones de flecha.

Ahora puede ser que esta parte de los métodos y el sea algo nuevo pero eso es lo que tendríamos que

comprender sobre este.

Este dice que simplemente nos sirve para apuntar a el objeto.

Ya no vamos a utilizar más adelante y conforme más lo usamos más nos vamos a acostumbrar a trabajar

con él.

Pero si sienten que necesitan algún reforzamiento les recomendaría que tomaran algún curso especializado

de JavaScript para entrar en más detalles del Viz porque esto podríamos tener varias clases hablando

sobre este objeto.

Pero eso es todo lo que quería cubrir en esta clase.

Voy a copiar está el PP apuntó J.V.

Te voy a pegar aquí adentro de mi Tapscott y esta lección era sobre funciones de flecha y nada más Flecha

voy a cerrarlo.

Borremos todo y saben qué podemos cambiar esto por una función anónima auto invocada pero utilizando

una función de flecha.

De esa manera qué es exactamente lo mismo.