Esta va a ser la última clase que vamos a tener antes de un examen teórico y uno práctico.

Les voy a pedir que abramos nuevamente el proyecto de Scripps y el que nosotros teníamos así básico

lo vamos a dejar caer dentro de Visual Studio Catt y ãramos teníamos aquí un archivo de funciones de

flechado donde tenemos varias funciones porque tenemos varios varias funciones acá voy a cerrarlo simplemente

quiero expandir algo que se nos había olvidado o que no habíamos tocado porque no teníamos el tema de

las interfaces y otras cosas haremos el p.ej. ese y podemos borrar todo lo que teníamos acá aunque todo

esto lo vamos a borrar y vamos a empezar a explicar sobre el tipado de un retorno de una función voy

a crear una constante llamada sumar sumar es una función que va a recibir dos argumentos voy a poner

una fecha cerró llaves y voy a hacer un retorno.

Voy a recibir a este tipo number pero sirve de tipo number me faltó una coma y el rector no va a ser

a más puntico.

Si yo dejo el cursor encima vamos a ver qué es lo que dice acá dice que sumar va a regresar un número.

Pero cómo podemos saber de que efectivamente es un número con esto.

Bueno porque obviamente la función me lo está diciendo pero si yo necesito especificar el tipo.

Entonces después de definir los argumentos voy a poner dos puntos y voy a escribir number one.

De esta manera queda aún más claro de que esta función regresa un número.

Ahora ustedes pueden definir primero la firma.

O sea todo lo que es la definición de los tipado y su retorno para evitar accidentalmente poner acá

un error como digamos que accidentalmente yo lo puse como un string.

Entonces si dejo el cursor encima me dice que el tipo jamás voy a todo ese string.

No puede ser asignarle al tipo number de esa manera yo puedo agregar condiciones extra para evitar errores

a la hora de los ratones o viceversa.

Si yo necesito que esto sea un estreno y estoy mandando un número entonces aquí me toca a mí me va a

marcar el error pero dejémoslo como nombre.

Ahora todo este código se puede resumir simplemente a más.

Como ustedes saben puntico.

Y ahí tenemos exactamente lo mismo aunque en este caso es sumamente sencillo saber por qué.

Porque la función ya es la puede inferir taches y efectivamente todavía lo encuentra ok number pero

voy a poner los dos puntos simplemente para que quede como referencia.

Ahora hay veces donde nosotros tenemos casos que no es capaz de inferir el tipo pero ya voy a llegar

a eso.

Simplemente quiero quedarme otra constante llamado nombre en función de flecha orillero llaves.

Y lo único que va a hacer es un retorno de Hola Mundo o Hola Fernando.

Punto y Coma si dejo el cursor encima del nombre van a ver que regresa un string.

Si yo le quiero poner el tipo.

Entonces después de los paréntesis dos puntos Schenk y de esta manera queda sumamente claro de que regreso

Mistery.

Ahora sí hagamos un ejemplo en el cual TIC's no puede inferir el tipo de una constante llamada obtener

salario obtener salario esto va a ser igual a una función de flechado desarrolladas aquí adentro es

el retorno de una promesa parentesis Punto y Coma dentro de mi promesa Voy a crearme las dos funciones

que es el rizó como Willett aquí adentro.

Función de flecha.

Y nuevamente abro y cierro llaves Ok este obtener salario es una función que regresa una promesa pero

no sé qué tipo de salida es.

Por consecuencia si yo intentara usar esa función voy a poner aquí obtener salario.

Voy a poner el punto.

Yo recibiría un argumento voy a ponerle a ese argumento y voy a ser Impression en consola de ese argumento.

Si de algo soy a es imposible para mí saber qué tipo de dato es el que estoy recibiendo acá lo que puede

llevar a ciertos errores.

Por ejemplo si yo necesitara imprimir a pasarlo todo mayuscula tu Opeth qué van a ver que no aparece.

Por consecuencia si yo no regresar a un número acá puede darme ciertos errores o llevarme a ciertos

errores que yo no quiero.

Entonces para definirle el tipo de la promesa Voy a hacer acá voy a poner dos puntos después de los

argumentos y voy a decir que se regresa su promesa con la P mayúscula y entre mayor y menor voy a especificar

el tipo de respuesta.

En este caso voy a decir que esta promesa va a resolver un Strength ok aquí no me está marcando un error

todavía porque la promesa está regresando en fin aquí ya debería de marcarme por lo menos algo aquí

adentro de la promesa Voy a escribir rizó y si yo pongo un uno van a notar que el 1 me marca el error

debería de marcarlo antes.

Si me pregunta a mí por lo menos debería marcarlo antes tal vez cuando ustedes lo hagan o lo estén viendo

ya les marqué el error pero si dejo el cursor encima alguno me dice que el argumento tipo 1 no puede

ser asignarle al tipo string excelente.

Entonces puedo cambiarlo y poner aquí Fernando al hacerlo así dejó el cursor encima de.

Ya sabe que es un estreno por consecuencia ahora sí puedo hacer el punto tu Hopper Keys parentesis y

perfectamente eso es básicamente todo.

Con esta forma o de esta forma ustedes pueden especificar el tipo de retorno de las funciones lo vamos

a dejar hasta este punto.