¿Qué es un TypeScript Class Expression? .Normalmente cuando trabajamos con un lenguaje de programación orientado a objetos estamos muy acostumbrados a generar clases . Vamos a ver un ejemplo con la clase Telefono en TypeScript.

```
class Telefono {

  modelo:string;
  precio:number;
  marca:string;


  constructor(modelo:string, precio:number,marca:string) {

    this.modelo=modelo;
    this.precio=precio;
    this.marca=marca;
  }
}
```

Recordemos que en typeScript las variables son públicas por defecto. Si creamos un programa main para crear una persona será tan sencillo como:

```
import {Telefono} from "./03Telefono/Telefono";

let telefono= new Telefono ("iphone",1000,"apple")
```

```
console.log(telefono.marca);
console.log(telefono.modelo);
console.log(telefono.precio);
```

El resultado se imprime en la consola:



Usando TypeScript Class Expression

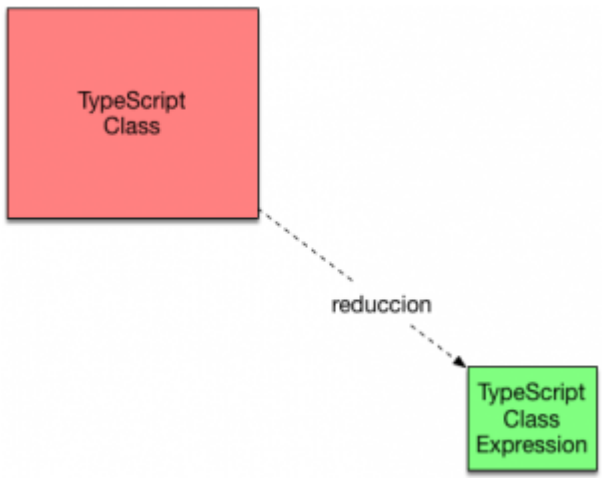
Sin embargo TypeScript es un lenguaje más moderno y soporta una sintaxis más compacta a la hora de declarar la clase . Podríamos declara la clase como un TypeScript Class Expression con lo cual el código quedaría de la siguiente forma:

```
export class Telefono {
  constructor(public modelo:string, public precio:number,public
marca:string) {
  }
}
```

Este código es mucho más compacto y genera el mismo resultado cuando lo ejecutamos por la consola:

```
apple
iphone
1000
```

Acabamos de simplificar sobremanera el programa construido:



TypeScript añade muchos nuevos conceptos a los lenguajes de programación clásicos:

1. TypeScript interface utilizando Angular DTOs
2. TypeScript CallBack ,funciones y sintaxis
3. TypeScript Generics y su funcionamiento
4. TypeScript