**TYPESCRIPT INTERFACES AND CLASSES OPTIONAL PARAMETERS AND PROPERTIES**

* We can mark the parameters, properties and methods of a class and interface as optional.
* The **?** is used to mark the properties, methods and parameters as optional.
* The **?** is used just after the properties, method or parameter name to mark it as optional in class or interface.
* //? Optional Parameters and Properties.
* //\* The ? is used to define optional parameters and properties in classes and interfaces.
* *interface* Animal {
* name: *string*,
* worth?: *string* // optional property
* getWorth?(*worth*: *string*): *void*;  //optional method
* }
* *class* Dog implements Animal {
* *constructor*(public *name*: *string*, public *breed*?: *string*) { }
* getWorth(*worth*: *string*): *void* {
* console.log(`The dog's worth is ${*worth*}`);
* }
* }
* *const* bully = new Dog('Cyber');
* *const* americanPitbull = new Dog('Buddy', 'American Pitbull');
* console.log('The instance bully and americanPitbull => ', bully, americanPitbull)