**TYPESCRIPT ABSTRACT CLASSES**

* The typescript **abstract** keyword is used to define the abstract classes as well as the abstract method declaration.
* The abstract class cannot be instantiated directly, it needs to be extended.
* The abstract classes enforce all the extending/inheriting classes to have a certain method whose declaration is provided in the base/parent class.
* The abstract method simply tells the child classes to have a particular method whose type definition is given in the base/parent class.
* //? The abstract classes.
* //\* The abstract keyword is used to define abstract classes.4
* //\*  An abstract class cannot be instantiated directly it can only be extended.
* //\* The abstract classes are used to enforce that all the inheriting/extending classes must have a certain method.
* //\* The abstract keyword is also used to provide the method declaration in the base class.
* abstract *class* Player {
* *get* playerNethWorth() { return this.netWorth }
* *constructor*(public *name*: *string*, public *age*: *number*, protected *netWorth*: *string*, public *sports*: *string*) { }
* abstract getPlayerInfo(): *string* //? All the extending/inheriting class must have the getPlayerInfo method defined.
* }
* //! const yuvi = new Player('yuvraj', 26, '2cr', 'cricket'); Error: Abstract classes cannot be instantiated.
* *class* SoccerPlayer extends Player {
* *constructor*(*name*: *string*, *age*: *number*, *netWorth*: *string*, public *jerseyNumber*: *number*, private *status*: 'active' | 'inactive') {
* super(*name*, *age*, *netWorth*, 'soccer')
* }
* getPlayerStatus() {
* return this.status
* }
* //\* Abstract method defination enforced by the abstract parent class.
* getPlayerInfo() {
* return `The ${this.sports} player named ${this.name} aged ${this.age} has a net-worth of ${this.netWorth} wears the jersey number ${this.jerseyNumber} with a ${this.status} status.`
* }
* }
* *const* shelby = new SoccerPlayer('Thomas Shelby', 23, '1.2cr', 7, 'active');
* console.log(shelby.getPlayerInfo());