TypeScript Class		TypeScript Interface
Introduction	Classes are the fundamental entities used to create reusable components. It is a group of objects which have common properties. It can contain properties like fields, methods, constructors, etc.	An Interface defines a structure which acts as a contract in our application. It contains only the declaration of the methods and fields, but not the implementation.
Usage	It is used for object creation, encapsulation for fields, methods.	It is used to create a structure for an entity.
Keyword	We can create a class by using the class keyword.	We can create an interface by using the interface keyword.
Compilation	A class cannot disappear during the compilation of code.	Interface completely disappeared during the compilation of code.
Real-Time Usage	Design Pattern, Designing project Structure	Implements of defined Architectures
Instantiation	A class can be instantiated to create an object.	An interface cannot be instantiated.
Methods	The methods of a class are used to perform a specific action.	The methods in an interface are purely abstract (the only declaration, not have a body).
Access Specifier	The member of a class can be public, protected, or private.	The members of an interface are always public.
Constructor	A class can have a constructor.	An interface cannot have a constructor.
Implement/Extend	A class can extend only one class and can implement any number of the interface.	An interface can extend more than one interfaces but cannot implement any interface.

TypeScript Class		TypeScript Interface
Introduction	Classes are the fundamental entities used to create reusable components. It is a group of objects which have common properties. It can contain properties like fields, methods, constructors, etc.	An Interface defines a structure which acts as a contract in our application. It contains only the declaration of the methods and fields, but not the implementation.
Usage	It is used for object creation, encapsulation for fields, methods.	It is used to create a structure for an entity.
Keyword	We can create a class by using the class keyword.	We can create an interface by using the interface keyword.
Compilation	A class cannot disappear during the compilation of code.	Interface completely disappeared during the compilation of code.
Real-Time Usage	Design Pattern, Designing project Structure	Implements of defined Architectures
Instantiation	A class can be instantiated to create an object.	An interface cannot be instantiated.
Methods	The methods of a class are used to perform a specific action.	The methods in an interface are purely abstract (the only declaration, not have a body).
Access Specifier	The member of a class can be public, protected, or private.	The members of an interface are always public.
Constructor	A class can have a constructor.	An interface cannot have a constructor.
Implement/Extend	A class can extend only one class and can implement any number of the interface.	An interface can extend more than one interfaces but cannot implement any interface.