Scala cannot have static members

Bydefault function parameters are **Val**

Single Ton Object with no Companion object is called Standalone Object

Some times you need Multiple Constructors in a class. In Scala Contructors other than Primary Constructors are called Auxiliary Constructors

Auxillary Constructors Start with **def this..()**

1.Abstract class cannot be instantiated

2. In a Class if any field or Method is Abstract then Class needs Abstract Keyword

3.One class can extend only one super Class. A Class can have Constructor parameters .

Traits can mixin many classes

Trait Dont have Constructor parameters

4. Classes themselves may be abstract, and traits are by definition abstract, but neither of these are what are referred to as abstract types in Scala. An abstract type in Scala is always a member of some class or trait, such as type T in trait Abstract.

5.Only **"val"** can be lazy

6.**Traits** Dont have Constructors

7> Case Class implictly provide 3 functions Equals,Hascode and toString

u may leave out new Keyword

8> In Scala the primary constructor does not have an explicit definition it is defined implicitly by the class parameters and body. The private parameters can be accessed only within class and within Companion class

9> Companion classes take no parameters and starts with Keyword Object ,instead of Class

companion object can be instantiated without "new"

10> private constructors and Private members are one way to hide the initialisation of class

private contructor parameters can be initialised by **this** or factory **method apply**

11**> trait Que[T]** Here Que is trait , But not a Type....Que is not a Type beacuse it take Type Parameter...Que[T] is trait and Que[String] is type

in Example **trait Que[+T]** prefixing the type parameter with a + indicated the subtyping is covariant

Tuples ,List are Immutable