Constructor	Destructor
1. Constructors are special member functions of a class having the same name as that of the class.	1. A destructor is also special member function of a class having same name as that of the class but prefixed with symbol of tilde (~).
2. Constructor is implicitly called by the compiler as soon as the object of a class gets created	2. A destructor is also automatically called by the compiler but just before the object of the class is to be destroyed
3. The C++ compiler automatically provides 2 constructors in our class if we don't define any constructor ourselves and they are called default constructor and default copy constructor.	3. The C++ compiler automatically provides 1 destructor by default if we don't provide any destructor ourselves and that destructor is called default destructor
4. Constructors are called in the order in which objects are created and it is always left to right.	4. Destructor is called in reverse order of creation of the object and it is always right to left
5. Constructor can be parameterized.	5. Destructor can't be parameterized.
6. Since they can be parameterized we can overload them and so a class can have multiple constructors.	6. Since they don't accept any parameter or argument their overloading is not possible and so a class can have only one destructor.
7. Constructor can't be declared as **"static".**	7. A destructor also can't be declared as **"static".**
8. Constructors can't be declared as **"const".**	8. A destructor also can't be declared as **"const".**
9. Constructors are not inherited	9. Inheriting a destructor is also possible
10. Constructors can't be declared as **virtual**	10. We can declare a destructor as **virtual**