91. Can static members use non static members? Give reasons.
Ans.--> 1) A static method is a method that belongs to a class,
but it does not belong to an instance of that class and this method can be called without the instance or object of tha t class.
2) Every methods in java defaults to non-static method without static keyword preceding it.
non-static methods can access any static method and static variable also, without using the object of the class.
3) Static method uses compile time or early binding.
4) Non-static method can be overridden because of runtime binding.
==========

92. Define different ways a method can be overloaded?
Ans.--> 1) Method overloading can be achieved by the following:
By changing the number of parameters in a method.
By changing the order of parameters in a method.
By using different data types for parameters.
==========

93. Can we have an abstract class without having any abstract method?
Ans.--> 1) Yes we can have an abstract class without Abstract Methods as both are independent concepts.
Declaring a class abstract means that it can not be instantiated on its own and can only be sub classed.
Declaring a method abstract means that Method will be defined in the subclass.
==========

94. Explain the default access modifier of a class?
Ans.--> Default access modifier means we do not explicitly declare an access modifier for a class, field, method, etc.

A variable or method declared without any access control modifier is available to any other class in the same packa ge.
==========

95. Can function overriding be explained in same class?
Ans.--> While overriding −

1)Both methods should be in two different classes and, these classes must be in an inheritance relation.

2)Both methods must have the same name, same parameters and, same return type else they both will be treated as d ifferent methods.

3)The method in the child class must not have higher access restrictions than the one in the superclass.
If you try to do so it raises a compile-time exception.

4)If the super-class method throws certain exceptions,
the method in the sub-class should throw the same exception or its subtype (can leave without throwing any excepti on).

Therefore, you cannot override two methods that exist in the same class, you can just overload them.
==========

96. Does function overloading depends on Return Type?
Ans.--> No,It does not depend on Return Type. Because if return type is different and function name as well as para meter is also same
The return type of a function does not create any effect on function overloading.
Same function signature with different return type will not be overloaded.
==========

97. Define different ways to declare an array?
Ans.-->1) A Java array variable can also be declared like other variables with [] after the data type.
The variables in the array are ordered and each have an index beginning from 0.
Java array can be also be used as a static field, a local variable or a method parameter.
2) The usual way of declaring an array is to simply line up the type name, followed by a variable name, followed by a size in brackets, as in this line of code: int Numbers[10]; This code declares an array of 10 integers. The first element gets index 0, and the final element gets index 9.

==========

98. Can abstract class have a constructor?
Ans.--> 1)As we all know abstract classes also do have a constructor.
So if we do not define any constructor inside the abstract class then
JVM (Java Virtual Machine) will give a default constructor to the abstract class.
2)It must be declared with an abstract keyword. It can have a constructor, static method.

==========

99. Define rules of Function overloading and function overriding?
Ans.--> 1) Function Overloading happens in the same class when we declare same functions with different argument s in the same class.
Function Overriding is happens in the child class when child class overrides parent class function.
2) In java, method overloading can't be performed by changing return type of the method only. Return type can be s ame or different in method overloading.
But you must have to change the parameter.
3) Return type must be same or covariant in method overriding.

==========

100. Explain the concept of Pure Virtual Functions?
Ans.--> 1)A pure virtual function or pure virtual method is a virtual function that is required to be implemented by a derived class if the derived class is not abstract.
Classes containing pure virtual methods are termed "abstract" and they cannot be instantiated directly.
2)Pure Virtual Function. A virtual function for which we are not required implementation is considered as pure virtu al function.
For example, Abstract method in Java is a pure virtual function.

==========