**1. What is OOPS?**  
Object-oriented programming (OOP) is a programming language model organized around objects rather than "actions" and data rather than logic. Historically, a program has been viewed as a logical procedure that takes input data, processes it, and produces output data.  
  
**2. Basic Concepts of OOPs?**  
Abstraction.  
Encapsulation.   
Inheritance.   
Polymorphism.  
  
**3. What is a class?**  
A set or category of things having some property or attribute in common and differentiated from others by kind, type, or quality.  
  
**4. What is an object?**  
Objects are created from Classes, in C#, is an instance of a class that is created dynamically. Object is also a keyword that is an alias for the predefined type System.  
 **5. What is Encapsulation?**  
Encapsulation is the packing of data and functions into a single component. The features of encapsulation are supported using classes in most object-oriented programming languages, although other alternatives also exist.  
  
It allows selective hiding of properties and methods in an object by building an impenetrable wall to protect the code from accidental corruption.

**6. What is Polymorphism?**  
In programming languages and type theory, polymorphism is the provision of a single interface to entities of different types.  
A polymorphic type is a type whose operations can also be applied to values of some other type, or types.  
  
**7. What is Inheritance?**  
Inheritance is when an object or class is based on another object or class, using the same implementation (inheriting from a class) specifying implementation to maintain the same behavior (realizing an interface; inheriting behavior).  
  
It is a mechanism for code reuse and to allow independent extensions of the original software via public classes and interfaces.  
  
**8. What is Constructor?**  
A is special method of the class that will be automatically invoked when an instance of the class is created is called as constructor.  
  
Constructors are mainly used to initialize private fields of the class while creating an instance for the class.  
  
When you are not creating a constructor in the class, then compiler will automatically create a default constructor in the class that initializes all numeric fields in the class to zero and all string and object fields to null.  
  
Syntax.  
[Access Modifier] ClassName([Parameters])  
{  
}  
  
**9. Types of Constructors**  
Basically constructors are 5 types those are  
Default Constructor  
Parameterized Constructor  
Copy Constructor  
Static Constructor  
Private Constructor  
  
**10. Define Destructor?**  
A destructor is a method which is automatically invoked when the object is destroyed.  
  
Its main purpose is to free the resources (memory allocations, open files or sockets, database connections, resource locks, etc.)

**11. What is Inline function?**  
In the C and C++ programming languages, an inline function is one qualified with the keyword inline; this serves two purposes.  
Firstly, it serves as a compiler directive, which suggests (but does not require) that the compiler substitute the body of the function inline by performing inline expansion,  
The second purpose of inline is to change linkage behavior; the details of this are complicated.  
  
**12. What is operator overloading?**  
In programming, operator overloading—less commonly known as operator ad hoc polymorphism—is a specific case of polymorphism, where different operators have different implementations depending on their arguments. Operator overloading is generally defined by the language, the programmer, or both.  
  
**13. Different between method overriding and method overloading?**  
In Overriding methods it will create two or more methods with same name and same parameter in different classes.  
  
While overloading it will create more than one method with same name but different parameter in same class.  
  
**14. What is this keywords?**  
Every instance method in every object in Java receives a reference named this when the method is invoked.  The reference named this is a reference to the object on which the method was invoked.  It can be used for any purpose for which such a reference is needed.  
  
**15. What is super keyword?**  
The super keyword is a reference variable that is used to refer immediate parent class object.  
  
**16. What is an abstract class?**

An abstract class is a class that is declared abstract—it may or may not include abstract methods. Abstract classes cannot be instantiated, but they can be subclassed.  
  
**17. What is final keywords?**  
  
The final keyword can be applied with the variables, a final variable that have no value it is called blank final variable or uninitialized final variable. It can be initialized in the constructor only. The blank final variable can be static also which will be initialized in the static block only.

**18. What is difference between abstraction and abstract class?**

Abstract class a type of class that object cannot be create it contain abstract or not abstract method while abstraction is mechanism of data hiding