**Name – Mikias Dabessa**

**ID -ATE/6856/11**

**OOP 2 CONCEPTUAL QUESTION ANSWERS**

1. In [object-oriented programming (OOP)](https://searchapparchitecture.techtarget.com/definition/object-oriented-programming-OOP), objects are the things you think about first in designing a program and they are also the units of code that are eventually derived from the process

2. Encapsulation in OOP Meaning: In object-oriented computer programming languages, the notion of encapsulation (or OOP Encapsulation) refers to the bundling of data, along with the methods that operate on that data, into a single unit.

3. Abstraction occurs when a programmer hides any irrelevant data about an object or an instantiated class to reduce complexity and help users interact with a program more efficiently.

4. Access modifiers and specifies are keywords (private, public, internal, protected and protected internal) to specify the accessibility of a type and its members.

5. Inheritance is the procedure in which one class inherits the attributes and methods of another class.

6. In Multiple inheritances, one class can have more than one superclass and inherit features from all its parent classes

7. The derived class doesn't "inherit" the private members of the base class in any way - it can't access them, so it doesn't "inherit" them.

8. Polymorphism the ability of objects of different types to provide a unique interface for different implementations of methods.

9. Method overloading is the ability to redefine a function in more than one form.

10. If we need to do the same kind of the operation in different ways

11. Method Overriding is a technique that allows the invoking of functions from another class (base class) in the derived class.

12. A constructor is a special method of the class which gets automatically invoked whenever an instance of the class is created.

13. Constructor of a class must have the same name as the class name in which it resides

• A constructor cannot be abstract, final, and synchronized.

• Within a class, you can create only one static constructor.

14.  Private constructor is a special instance constructor. It is generally used in classes that contain static members only.

15. No, object of a class having private constructor cannot be instantiated from outside of the class.

16. Private constructors are used to prevent creating instances of a class when there are no instance fields or methods

17. Static constructor is a special constructor that gets called before the first object of the class is created.

18. A Destructor is automatically invoked when an object is finally destroyed.

19. Namespace allows creating a system to organize the code.

20. Virtual is used to modify a method, property, indexer, or event declared in the base class and allows it to be overridden in the derived class.

Override is used to extend or modify a virtual/abstract method, property, indexer, or event of the base class into the derived class.

New is used to hide a method, property, indexer, or event of the base class into the derived class.

21. Structs are value types, allocated either on the stack or inline in containing types. While, classes are reference types allocated on the heap and garbage-collected.

22. An interface looks like a class, but it has no implementation.

23. Because it allows us to easily interchange one component for another which is using the same interface.

24. What is implicit interface implementation?

24. Interfaces are implemented implicit by declaring a public member in the class with the same signature of the method as defined in the interface and the same return type.

25. An explicit interface implementation is a class member that is only called through the specified interface.

26. An abstract class is a class that is declared abstract —it may or may not include abstract methods.

27.  Abstract classes cannot be instantiated, but they can be sub classed.

28. Abstraction is the method of hiding the unwanted information. Whereas encapsulation is a method to hide the data in a single entity or unit along with a method to protect information from outside

29. Abstract classes cannot be declared sealed

30. Yes, an abstract class can have a constructor.

31. No, abstract method can’t be private

32. Yes, abstract class can have Static Methods.

33. No it doesn’t support multiple interfaces

34. False

35. An abstract class is used if you want to provide a common, implemented functionality among all the implementations of the component.

36. The sealed modifier prevents a class from being inherited and the abstract modifier requires a class to be inherited.

37. The abstract keyword enables you to create classes and class members that are incomplete and must be implemented in a derived class.

38. Operator overloading Is a specific case of polymorphism, where different operators have different implementations depending on their arguments

39. Yes it is possible by using the static variable

40. No you can’t inherit.

41. You can use extension methods to extend a class or interface

42. No, you can't return multiple values from a function

43. Constants are immutable values which are known at compile time and do not change for the life of the program.

44. They are known at compile and run time and do not change their values at run time like in any function for the life of application till the application is running.

45. They are known at compile and run time and do not change their values at run time like in any function for the life of application till the application is running.

46. Static Readonly type variable value can be assigned at runtime or at compile time and can be changed at runtime.