Q1 What is Apex Programming?

Ans: Apex is a strongly typed, object-oriented programming language that allows developers to execute flow and transaction control statements on the Lightning platform server in conjunction with calls to the Lightning Platform API.

Q2 What are the features of Apex Programming?

Ans:

- It upgrades automatically.
- Integrated with the DB which means it can access and manipulate records without the need to establish the DB connection explicitly.
- It has java like syntax and it is easy to use.
- It is easy to test as it provides built in support for executing test cases.
- Multi-Tenant Environment.
- You can save your apex code against different versions of the force.com API.
- Apex is a case-insensitive language.

Q3 What are the Similarities between apex and other programming?

Ans: A programming language is used to interact with the system or a language which is used to give instructions.

- Web scripting programming language
- Programming language [C++, JAVA]
- Cloud based programming language

Q4 How Apex programming is different from other programming languages?

Ans: Apex is different from other programming language are as follows:

- Apex run in a multitenant architecture.
- Apex performs uncontrolled program innovations.
- Apex is case-insensitive.
- Apex can be processed on any platform.

Q5 What is a Data Type?

Ans: A particular kind of data item, as defined by the values it can take, the programming language used, or the operations that can be performed on it.

Q6 What is a Variable?

Ans: In programming, a variable is **a value that can change, depending on conditions or on information passed to the program**. Typically, a program consists of instructions that tell the computer what to do and data that the program uses when it is running.

Q7 What all data types are supported in Apex programming?

Ans:

- Primitive (Integer, Double, Long, Date, Datetime, String, ID, or Boolean)
- Collections (Lists, Sets and Maps) (To be covered in Chapter 6)
- sObject
- Enums
- Classes, Objects and Interfaces

Q8 What is Sobject?

Ans: Sobjects are **standard or custom objects that stores record data in the force.com database**. There is also SObject datatype in apex that is the programmatic representation of these SObjects. Developers refers to SObject and their fields by their API names.

Q9 What are the types of Sobject?

Ans: Generic & Specific

Q10 What is the use of the Substring() method?

Ans: Returns a new String that begins with the character at the specified zero-based startIndex and extends to the end of the String.

Q11 Which method is used to get the length of a string?

Ans: length() method can be used to find the length of the String.

Q12 Which method is used to replace the value in a string?

Ans: replaceAll()

Replaces each substring of a string that matches the regular expression regExp with the replacement sequence replacement .

Q13 What are Conditional Constructs and types of conditional constructs?

Ans: Conditional constructs are used to organize basic conditions, loops, or nested conditional constructs in an If-then-elseif type of logical sequence to create complex expressions for evaluation.

- If statement.
- If-Else statement.
- Nested If-else statement.
- If-Else If ladder.
- Switch statement.

Q14 What are Looping Constructs and types of looping constructs?

Ans: Looping constructs are **used when the same set of steps has to be carried out many times**. There is usually a counter that indicates how many times the loop is executed, or a test that is made every time the loop is executed to see if it should be executed again.

There are three main types of loops: **For, While, and Do... While**

Q15 What are comments? Types of comments?

Ans: Comments are **used to make the program more readable by adding the details of the code**. It makes easy to maintain the code and to find the errors easily. The comments can be used to provide information or explanation about the variable, method, class, or any statement.

There are two types of comment:

- //
- /*….......*/

Q16 What is Enumeration? Which keyword is used to define an Enumeration?

Ans: An enumeration is a user-defined data type that consists of integral constants. To define an enumeration, keyword **enum** is used.

Q17 Which loop is also known as exit-controlled loop and why?

Ans: Do while loop is the example of Exit controlled loop.

When statements inside the loop body is executed and then the condition is checked

Q18 Why do we use the ValueOf() method?

Ans: It converts string to integer.

Q19 What is an Array?

Ans: Arrays in Apex are basically the same as Lists in Apex. There is no logical distinction between the Arrays and Lists as their internal data structure and methods are also same but the array syntax is little traditional like Java.

Q20 What is the starting index of an Array?

Ans: Each element in the array is located by index and the index value starts with **zero**.

Q21 How to calculate the last index of an Array?

Q22 What are Apex Collections?

Ans: List, Set and map are collections used in apex.

Q23 What is a List Collection?

Ans: A list is **an ordered collection of elements that are distinguished by their indices**. List elements can be of any data type—primitive types, collections, sObjects, user-defined types, and built-in Apex types.

Q24 To retrieve value from a particular index from a list?

Ans: get()

Q25 To add values of a list into another list?

Ans: Addall()

Q26 Which method is used to add a value in a list at a particular index?

Ans: add(index, value)

Q27 What is the usage of set() method in list?

Ans: This method replaces the value of particular index.

Q28 If you want to delete a value from a particular index in a list?

Ans: remove()

Q29 What is the use of the clear() method in list?

Ans: The Clear method is used at the end of the program, **to remove all items from the list**, and the Capacity and Count properties are then displayed.

Q30 How is List different from a set?

Ans: A **list** is an ordered collection of elements that are distinguished by their indices. List elements can be of any data type—primitive types, collections, sObjects, user-defined types, and built-in Apex types.

A **set** is an unordered collection of elements that do not contain any duplicates.

Q31 Tell any 3 methods of set?

Ans: add()

Size()

Contains()

Q32 What is the use Contains() method?

Ans: The contains() method checks whether a string contains a sequence of characters. Returns true if the characters exist and false if not.

Q33 What is a Map?

Ans: A map is a collection of key-value pairs where each unique key maps to a single value. Keys and values can be any data type—primitive types, collections, sObjects, user-defined types, and built-in Apex types.

Q34 Tell any 5 methods of Map?

Ans: get()

Put()

Putall()

Remove()

Size()

Q35 What is the use of values() method in a Map?

Ans: Values() method returns the Collection view of the values contained in this map.

Q36 Which method is used to get the keys of a Map?

Ans: keyset()

Q37 What is the difference between Map and Set?

Ans: A **map** is a collection of key-value pairs where each unique key maps to a single value. Keys and values can be any data type—primitive types, collections, sObjects, user-defined types, and built-in Apex types.

A **set** is an unordered collection of elements that do not contain any duplicates.

Q38 What is the use of ContainsKey() method in a Map?

Ans: Map. containsKey() method is used to check whether a particular key is being mapped into the Map or not.

Q39 To add all values of a map into another map?

Ans: putall()

Q40 What is OOP?

Ans: Object-oriented programming (OOP) is a computer programming model that organizes software design around data, or objects, rather than functions and logic.

Q41 What are the features of OOP?

Ans: The important features of Object Oriented programming are:

- Inheritance
- Polymorphism
- Data Hiding
- Encapsulation
- Overloading
- Reusability

Q42 What is a Class?

Ans: In object-oriented programming, a class is a template definition of the methods and variables in a particular kind of object.

Q43 What is an Object?

Ans: An object can be defined as an instance of a class.

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.

Q44 What is Abstraction and how is it achieved in programming?

Ans: Abstraction is a feature of OOPs. The feature allows us to hide the implementation detail from the user and shows only the functionality of the programming to the user. Because the user is not interested to know the implementation. It is also safe from the security point of view.

Abstraction is achieved by **interfaces and abstract classes**. We can achieve 100% abstraction using interfaces. Abstract classes and Abstract methods: An abstract class is a class that is declared with an abstract keyword.

Q45 What is Encapsulation and is it achieved in programming?

Ans: Encapsulation is defined as **the wrapping up of data under a single unit**. It is the mechanism that binds together code and the data it manipulates. Another way to think about encapsulation is, that it is a protective shield that prevents the data from being accessed by the code outside this shield.

A setter method is used to set or update the value of a specific variable within a class. Programmers can use access modifiers to define the visibility and accessibility of classes, along with the data and methods that they contain.

Q46 What is Inheritance and its types?

Ans: Inheritance is **the procedure in which one class inherits the attributes and methods of another class**. The class whose properties and methods are inherited is known as the Parent class. And the class that inherits the properties from the parent class is the Child class.

There are different types of inheritance:

- Single inheritance
- Multiple inheritance
- Multilevel inheritance
- hybrid inheritance
- hierarchical inheritance

Q47 What is Polymorphism and its type?

Ans: Polymorphism is the ability to process objects differently on the basis of their class and data types.

Types of Polymorphism:

- Static Binding (Compile time Polymorphism)
- Dynamic Binding (Runtime Polymorphism)

Q48 What are access modifiers?

Ans: Access modifiers are keywords used to specify the accessibility of a class (or type) and its members.

Q49 Which is the default access modifier in Apex?

Ans: The default access modifier in Apex is **private.**

Q50 What are static members?

Ans: When we declare a member of a class as static it means no matter how many objects of the class are created, there is only one copy of the static member. **A static member is shared by all objects of the class**. All static data is initialized to zero when the first object is created, if no other initialization is present.

Q51 What are Constants? Which keyword is used for declaring constants?

Ans: Constants cannot be changed once it is declared. Class constants can be useful if you need to define some constant data within a class. **A class constant is declared inside a class with the const keyword**. Class constants are case-sensitive. However, it is recommended to name the constants in all uppercase letters.

The keyword **const** is used to declare these constant variables.

Q52 What are the pillars of OOP?

Ans: Pillars of oops:

- Abstraction.
- Encapsulation.
- Inheritance.
- Polymorphism.

Q53 What is the usage of return statement?

Ans: The return statement is used for returning a value when the execution of the block is completed.

Q54 What is a Method/function?

Ans: **A method is a procedure or function in object-oriented programming**. A function is a group of reusable code which can be called anywhere in your program.

Q55 What are properties?

Ans: A property, in some object-oriented programming languages, is a special sort of class member, intermediate in functionality between a field (or data member) and a method.

Q56 How multiple inheritance is implemented in Apex?

Ans: To inherit any class, it must be defined with keyword Virtual.

Q57 What is an Abstract class?

Ans: An abstract class is a class that contains at least one abstract method. An abstract method is a method that is declared, but not implemented in the code.

Q58 What are Abstract methods?

Ans: Abstract Method is a method that has just the method definition but does not contain implementation. A method without a body is known as an Abstract Method. It must be declared in an abstract class. The abstract method will never be final because the abstract class must implement all the abstract methods.

Q59 What is an Interface?

Ans: An interface is a contract between itself and any class that implements it. Interface can have methods, properties, or events. It contains only declaration of its members and implementation of its members will be given by the class who implements the interface. Interface makes it easy to maintain the program.

Q60 What is the difference between Abstract Class and Interface?

Ans:

Abstract Class	**Interface**
An abstract class can have both abstract and non-abstract methods.	The interface can have only abstract methods.
It does not support multiple inheritances.	It supports multiple inheritances.
It can provide the implementation of the interface.	It cannot provide the implementation of the abstract class.
An abstract class can have protected and abstract public methods.	An interface can have only have public abstract methods.
An abstract class can have final, static, or static final variable with any access specifier.	The interface can only have a public static final variable.

Q61 What is a Constructor?

Ans: In OOP (Object-oriented programming) constructor is a special method. It is called whenever you create an object using **new** keyword. Constructor enables an object to initialize itself at the time of its creation without the need to make a separate call to the instance method. It looks like a method but it is different from the method in two ways as follow

- A constructor always has the same name as the class whose instance members they initialize.
- It also does not have a return type and not even void like methods. It causes the compiler to automatically call the constructor whenever an object of the class is created.

Q62 What is Function Overloading?

Ans: Function overloading is known as a function of polymorphism. The function can perform various operations best on the argument list. It differs by type or number of arguments they hold. By using a different number of arguments or different types of arguments, the function can be redefined.

Q63 What is Constructor Overloading?

Ans: Constructor Overloading have the same name (name of the class) but the different number of arguments. Depending upon the number and type of arguments passed, the corresponding constructor is called.

Q64 What is Function Overriding?

Ans: When it redefines a function of the base class in a derived class with the same signature i.e., name, return type, and parameter but with a different definition, it is called function overriding.

Q65 What is a Virtual class?

Ans: A virtual class is a nested inner class whose functions and member variables can be overridden and redefined by subclasses of an outer class.

Q66 What is the difference between Function Overloading and function overriding?

Ans: a) In overloading, there is a relationship between methods available in the same class whereas in overriding, there a is relationship between a superclass method and subclass method.

(b) Overloading does not block inheritance from the superclass whereas overriding blocks inheritance from the superclass.

(c) In overloading, separate methods share the same name whereas in overriding, subclass method replaces the superclass.

(d) Overloading must have different method signatures whereas overriding must have same signature.

Q67 What is the difference between Abstract and Virtual class?

Ans: Virtual methods have an implementation and provide the derived classes with the option of overriding it. Abstract methods do not provide an implementation and force the derived classes to override the method. So, abstract methods have no actual code in them, and (non-abstract) subclasses HAVE TO override the method

Q68 What is a Specific Sobject?

Ex:

Lead ld;

Contact con;

Account ac;

Q69 What is a Generic SObject?

Ex:

Sobject ob;

Q70 What are the ways to Initialize an Sobject variable?

Ans:

i) Using Constructor

ii) Dot notation

Q1.WAP to display a list storing 5 names of students.

Ans:

```
public class devques
{
    public static void student()
    {
        list<string> stu=new list<string>();
        stu.add('Aman');
        stu.add('Deva');
        stu.add('Manisha');
        stu.add('Amrita');
        stu.add('Vikash');
            system.debug('Show me all student name
'+stu);
    }
 }
```

Q2.WAP to find the total, percentage of marks in 5 subjects stored in array.

Ans:

```
public class devques
{
    public static void percentage(integer sub1,integer
sub2,integer sub3,integer sub4,integer sub5)

    {
         integer[] marks=new integer[5];
```

```
            double percentage, total;
        marks[0]=sub1;
        marks[1]=sub2;
        marks[2]=sub3;
        marks[3]=sub4;
        marks[4]=sub5;
        total=sub1+sub2+sub3+sub4+sub5;
        percentage=total/5;
        system.debug('Show me percentage '+percentage);
    }
}
```

Q3.WAP to store and display 10 integer values in set.

Ans:

```
public class devques
{
    public static void sets()
    {
        set<integer> value=new set<integer>();
        value.add(1);
        value.add(2);
        value.add(3);
        value.add(4);
        value.add(5);
        value.add(6);
        value.add(7);
        value.add(8);
        value.add(9);

        value.add(10);

        system.debug('Show me all values '+value);

    }

}
```

Q4.WAP to store and display student address with student id in a Map.

Ans:

```
public class devques

{

   public static void studentadd()

    {

        map<integer,string> stuadd=new
map<integer,string>();

        stuadd.put(1,'Kanpur');

        stuadd.put(2,'Uttarakhand');

        stuadd.put(3,'Kolkata');

        system.debug('Show me address and id of student
'+stuadd);

    }

}
```

Q5.WAP to create one map with values[(1,'Parrot'),(2,'Peacock')] and another with values[(3,'Sparrow'),(4,'Nightingale']. put the values of second map into first.

Ans:

```
public class devques
```

```
{
  public static void birdsname()
    {
        map<integer,string> bird1=new
map<integer,string>{1=>'Parrot',2=>'Peacock'};
         system.debug('Show me birds name '+bird1);
        map<integer,string> bird2=new
map<integer,string>{3=>'Sparow',4=>'Nightangle'};
            system.debug('Show me birds name '+bird2);
        bird1.putall(bird2);
        system.debug('Show me all birds name '+bird1);
    }
}
```

Q6.WAP to store values of map created in Q5 to a BirdsName list and keys into BirdsID set.

Ans:

```
public class devques
{
        public static void birds()
    {
        map<integer,string> birdsname=new
map<integer,string>();
        birdsname.put(1,'Parrot');
        birdsname.put(2,'Peacock');
        birdsname.put(3,'Sparrow');
        birdsname.put(4,'Nightangle');
        system.debug('Show me birdsname with id
'+birdsname);
        set<integer> birdid=new
set<integer>(birdsname.keyset());
        system.debug('Show me id of birds '+birdid);
        list<string> birdname=new
list<string>(birdsname.values());
        system.debug('Show me name of birds
'+birdname);
    }
}
```