Date _____ Page _____

Q. What is Java?

Ans:- It is portable lang. coz code written and apps developed in Java can run in any system like. macOS, linux, windows etc. by converting the code into byte.

Q. Why Java is platform independent language?

Ans:- Because it uses a virtual machine that can run on multiple system.

Q. Why Java is diff from other prog. lang?

Ans:- Java uses both compiler as well interpreter while other prog. lang. uses only one, that may be compiler or an interpreter.

→ for eg:- c and c++ use only compiler to compile the code and python uses only interpreter.

→ Java use both, it first uses compiler then interpreter

Q What are compiler and interpreter?

Ans:- Compiler translate the entire source code in a single run. An interpreter translates the entire source code line by line.

→ Compiler consumes less time than interpreter.

Q. Why pointers are not used in Java but in c++?

Ans:- Java doesn't have pointers, it has references.

Java:
→ pointer lead to confusion for a programmer.
→ It break security, using pointers harmful program like virus and other hacking programs can be created
→ It may crash the program easily.

Q. Threading in java and its type?

Ans:- It is a java feature that allows concurrent execution of two or more parts of a program for maxm utilization of cpu.

-> Each part of such program is called a thread. So

-> threads are light-weight processes within a process.

-> ~~Type of~~ states of thread:

~~i) Born~~

i) New state (Born)

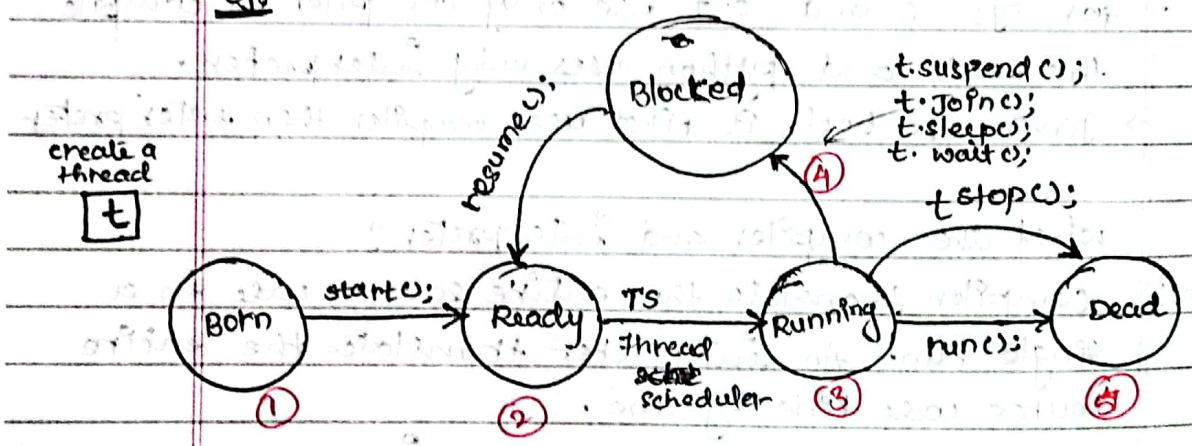
ii) Runnable state (Ready)

iii) Running state (Execution)

iv) waiting state (Blocked)

v) Dead state (Exit)

Eg:-



-> Initially thread t is in born state then it will start by using start() method then thread goes to ready state i.e. runnable state.

-> Now the thread is ready to execute but not in running state yet.

-> The thread scheduler will select the thread ~~from~~ then it will come into running state.

Date _____ Page _____

→ In running state thread t is getting executed and after successfull execution by run(); method the thread t will enter in the dead state.

→ If thread is still getting executed (ie not completely executed) and still we want to take it to dead state then we'll use t.stop() method. where t is thread name.

→ If we'll call any method among t.suspend/join/sleep/wait() then thread will go into blocked state. ie(waiting state)

→ Once thread goes into the blocked state it will again enter into ready state and thread scheduler will select the thread which are in waiting state and send it into the running state.

→ If we'll call suspend() method then thread will suspend or stop in blocked state and to start it again resume() method will be called (ie to bring thread from blocked state to ready state we need to use the resume() method.

→ In case of join/sleep/wait() method the blocked thread will automatically enter into the ready state.

→ If we'll mention the time then the thread will stop for given time and enter to the ready state.
for eg:- t.sleep(1000 m/s); the thread t will stop for 1 sec and enter into the ready state.

Q. Difference b/w extending & implementing thread?

Ans:- The major diff is when a class extends the thread class, you can't extend any other class, but by implementing the Runnable interface, it is possible to extend from another class as well, like; class ___ .
MyClass extends OtherClass implements Runnable.

→ Java focus on code simplicity, and the use of pointers can make it challenging.

C/C++ → pointers are used both in C/C++ for three main purposes: To allocate new object on the heap, To pass function to other functions, To iterate over elements in array or other data structure.

Q. What is variable and its type?

Ans:- Variable is the name of memory location.

→ It holds the value while the Java program is executed.

→ It is assigned with a datatype.

a) local variable :- variable declared inside the f^n are called local variable.

b) Global variable :- variable declared outside the f^n are called global variable.

c) static variable :- variable that are declared as static is called static variable.

Eg:-

```
public class A {
        static int m=100; // static variable
    void method() {
        int n=90; // local variable
    }

    public static void main (String args[]) {
        int data = 50; // global variable
    }
} // end of class
```

Date _____ Page _____

Q. What is class in Java?

Ans:- Class is a big entity that carry a lot of function.

eg:- class main {
void main() {
}
}

Q. Diff b/w heap memory and stack memory?

Ans:- Heap is the ~~hiera~~ hierarchical data structure whereas stack is a linear data structure.

→ Heap memory is used by all the parts of application whereas stack memory is used only by one thread of execution.

→ whenever an object is created its always stored in the heap space and stack memory contains the reference to it.

Q What is final keyword in Java?

Ans:- final ~~is a~~ keyword is a non-access modifier used for classes, attributes and ~~classes~~ methods that makes them non-changable. (impossible to inherit & override)

→ final keyword is ~~usefull~~ useful when you want to a variable to always store the same value like π (3.14).

→ It is also called "modifier".

→ final can be, variable, class & method.

Q What is super keyword in Java?

Ans:- It refers to the immediate parent of class and is used inside the subclass method definition for calling a method defined in the superclass.

→ It refers to superclass objects.

→ Used to call superclass methods, and to access the superclass constructor.

-> static member don't deal with ~~static~~ object in Java they are called by class name directly.

Date _____ Page _____

Q. what is keyword ?

Ans:- fixed value defined in the dictionary.

Q. Use of break and continue in Java. ?

Ans:- break statement is used to jump out of switch statement and also be used to jump out of loop.

eg:-
```
for (int i=0; i<10; i++) {
    if (i==4) {
        break; }
    syso (i);
}
```

-> Continue:- used to end the current iteration in a for loop (or a while loop) and continues to the next iteration

-> or to pass the control to next iteration of nearest for loop.

eg:-
```
for (int i=0; i<10; i++) {
    if (i==4) {
        continue; }
    syso(i); }
```

Q. what is JIT compiler ?

Ans:- JIT stands for Just-in-time compiler.

-> It is a component of Java Runtime environment that improves the performance of Java application at runtime.

Q. What is JVM and its use ?

Ans:-> Java virtual machine loads, verifies and executes Java bytecode. It is known as Interpreter or the core of Java prog. lang. coz it executes Java program.

-> platform independent & doesn't include software development tools.

abbreviation = summerization

Date _____ Page _____

Q. What is JRE and its use?

Ans:- Java runtime environment is software that Java programs require to run correctly.

→ JRE is the underlying technology that communicates between the Java program and the operating system.

→ ~~platf~~ platform dependent.

Q. What is JDK and its use?

Ans:- JDK in Java is a abbreviation for Java Development kit. It is a bundle of software development tools and supporting libraries combined with Java runtime Environment (JRE) and Java virtual machine (JVM)

→ It is platform dependent ie, for different platforms different JDK is required.

Q. What is string?

Ans:- It is predefined class. All predefined ~~class first~~ class's first alphabet is capital letter.

Q.

```
class A
{
    public static void main (String args[]) {
    }
}
```

→ public is access modifier in Java.

→ Reason why main method is public ~~is that~~, so that we can access the main method from outside the class.

→ JVM access the main method from outside the class while we execute the program.

→ If the main method will not be public then JVM can't call the main method and hence program can't be executed.

→ Main method is static so that we can access the main method without creating the object of class.

→ static member don't deal with object.

→ If it will not be static then we need to access the main method from object.

→ so as JVM call the main method then it will (static fn) be called automatically and it will not have to be dependent on any class.

→ void are datatype that is written before the function and if void is written before any function as return type i.e we don't have to return any value from that function.

→ args[] :- It is a variable of array type and called as object of string. we pass args[] from main method as an arguments, so that we can pass the information from main method.

→ If we don't pass any info then also we have to write this because it is the part of syntax.

Q. What is constructor?

Ans:- In java constructor is a special method that is used to initialize objects.

→ Constructor is called when an object of class is created

→ It is special type of function with no return type.

a). Default constructor (no-arg constructor)

b) parameterized constructor

a) → Default constructor :- when constructor doesn't have any parameter.

→ <class name>() { }

→ It is used to provide default value to the object like, 0, null etc., depending on the type.

Date _____ Page _____

eg:- // Java program to create and call default constructor //

```
class Bike1 {
// Creating a default constructor //
Bike1 () { syso ("Bike is created") ; }
// main method //
public static void main (String args[]) {
// calling a default constructor //
Bike1 b = new Bike1 ();
}
}
```

o/p : Bike is created

→ If there is no constructor in class, compiler automatically create a default constructor.

b) <u>Parameterized constructor</u> :- A constructor which has a specific number of parameters.

→ Used to provide diff value to distinct objects. However we can provide the same value also.

eg:- //

Q. What is method overloading in Java?

Ans:- If class has multiple methods having same name but different in parameters, it is known as method Overloading. It is eg of static polymorphism.

→ It increase the reliability of the program.

→ Two way to overload the method in Java.

a) <u>changing no. of arguments</u> :-

eg:-

```
class Adder {
    static int add (int a, int b) { return a+b ; }
    static int add (int a, int b, int c) { return a+b+c ; }
}
class TestOverloading1 {
    public static void main (String args[]) {
        syso (Adder.add (11, 11)) ;
        syso (Adder.add (11, 11, 11)) ;
    }
}
```

o/p : 22
33

→ compile time error is better than run time error. so, Java compiler renders (donor) compiler time if you declare the same method having same parameters.

Date _____ Page _____

b) <u>changing data type of arguments</u> :-

eg:-
```
class Adder{
  static int add(int a, int b){return a+b;}
  static double add(double a, double b){return a+b;}
}
  ~~class TestOverloading~~
  class TestOverloading2{
    public static void main(String args[]){
    syso(Adder.add(11,11));
    syso(Adder.add(12.3,12.6));
    }
}
```

o/p :- 22
24.9

→ we have created 2 method that differ in datatype. the first add method receives two integer args and 2nd receives two double args.

Q. why method overloading is not possible by changing the return type of method only ?

Ans:- Because of Ambiguity it is not possible.
(Ambiguity :- something that can be understood in more than one way)

eg:-
```
class Adder{
  static int add(int a, int b){return a+b;}
  static double add(double a, double b){return a+b;}
}
class TestOverloading3{
  public static void main(String args[]){
  syso(Adder.add(11,11)); // ambiguity
}}
```

o/p:- compile time error. method add(int, int) is already defined in class Adder.

→ syso(Adder.add(11,11)); // Here how Java determine which func method should be called ? //

→ overriding and overloading are two types of polymorphism.

Date _____ Page _____

Q. Can we overload Java main method?

Ans:- Yes, by method overloading. We can have any number of main methods in a class by method overloading.

→ But JVM calls main() method which receives string arrays as arguments only.

Eg:- class Overloading {

public static void main(String args[]){syso(" main with String[]");}

public s. v m (String args[]){syso ("main with String[]")}

~~public s v m (String args[]){syso("main with String[]");}~~

public static void main(){syso ("main without String[]");}

}

o/p:- main with String[]

Q. What is method overriding in Java?

Ans:- In Java method overriding occurs when a subclass (child class) has the same method as the parent class.

→ Used to provide the specific implementation of the method which is already provided by its superclass.

→ It is an example of dynamic polymorphism.

→ Java method overriding is ~~mainly~~ mostly used in Runtime polymorphism.

Eg → Rules for Java overriding :-

a) The method must have same name as in the parent class.

b) The method must have the same parameter as in the parent class.

c) There must be an IS-A relationship (Inheritance).

Eg:- In this ex. we have defined the run ~~out~~ method in subclass as defined in the parent class but it has some specific implementation. The name and the parameter of the method are the same, and there is IS-A relationship b/w the classes, so there is a method overriding.

```
// Java program to illustrate the use of method overriding //
// Creating a parent class //
class Vehicle {
// defining a method //
void run(){ sysO("Vehicle is running "); }
}

//creating a child class //
class Bike2 extends Vehicle{
// defining the same method as in the parent class //
void run(){ sysO("Bike is running safely"); }

public static void main (String args[]){
Bike2 obj = new Bike2(); // creating object //
obj.run(); // Calling method //
}
}
```

Q: What is compile time and run-time error in Java?

Ans:- A compile time error generally refer to errors that corresponds to the semantics or syntax.

→ It happens when you violet the rules of writing syntax.

→ This compiler error indicate something that must be fixed before the code can be compiled.

→ A class can't be private or ~~proote~~ protected except nested class.

Date _____ Page _____

Imp

→ The compile time is the period when the code is converted to the machine code and runtime is the period ~~when~~ of time when a program is running and generally occurs ~~when~~ after compile time.

→ The runtime error refers to the error that we encounter during the code execution during runtime.

→ We can easily fix a compile-time error during the development of code.

→ A compiler can't identify a runtime error.

→ Insufficient memory is eg. of runtime error.

→ Runtime error is a software and hardware problem that ~~event~~ prevent internet explorer from working correctly.

Q. What is access modifier in Java?

Ans:- It is keyword that can be used to control the visibility of fields, methods and constructors in a class.

→ The private keyword in Java is an access modifier.

→ It's used to say that a method or variable can only be accessible in the class where it's declared.

a) → Private:- It's access level is only within the class. It can't be accessed from outside the class.

b) ~~Default:- It's access level is only within the package.~~

eg:-
```
class A {
    private A() {} // private constructor //
    void msg() {System.out.("Hello Java");}
}
public class Simple {
    public static void main(String args[]) {
        A obj = new A(); // compile Time error //
    }
}
```

b) **Default:-** It's access level is only within the package.

→ Can't accessed from outside the package. If you don't specify any access level, it will be default.

eg:-

```
// Save by A.java //
package mypack;
import pack;
class A {
}
```

```
// Save by A.Java //
package pack;
class A {
    void msg() {System.out.("Hello");}
}
```

```
//Save by B.java//
package mypack;
import pack;
class B {
    public static void main (String args[]) {
        A obj = new A(); // Compile Time error //
        obj.msg(); // Compile Time error //
    }
}
```

In the given eg we have created two package, pack & mypack. we're accessing A class from outside its package, since A class is not public, so it can't be accessed from outside the package.

→ In the above eg, the scope of class A and its method msg() is default so it can't be accessed from outside package.

c) **protected:-** It's access level is within the package and outside the package through child class.

→ If you don't make a child class, it can't be accessed from ~~the~~ outside the package.

→ Provide more accessibility than default modifier.

Date _____ Page _____

Eg: we have created 2 package pack and mypack. The 'A' class pack package is public, so can be accessed from outside the package. But msg method of this package is declared as protected, so it can be accessed from outside the class only through inheritance.

```
//Save by A.Java//
package pack;
public class A {
protected void ~~main~~ msg()
    {sys0("Hello")}
}
```

```
//Save by B.Java//
package mypack;
import pack.*;
class B extends A{
public static void main (String args[])
{
  B obj = new B();
  obj.msg();
}
}
```

o/p Hello

d) public :- It is accessible everywhere. It has widest scope among all other modifiers.

Eg:- //Save by A.Java//

```
package pack;
public class A {
public void msg(){sys0("Hello")}
}
```

```
//Save by B.Java//
package mypack;
import pack.*;
class B {
public static void main (String args[]) {
  A obj = new A();
  obj.msg();
}
}
```

o/p Hello

Date _____ Page _____

Q. What is string in Java?

Ans:- Sequence of characters that exists as an object of class Java.lang.

→ Java strings are created and manipulated through the string class.

→ Once created, a string is immutable -- its value can't be changed.

Q. What is sorting and its type in Java?

Ans:- Sorting is the process of arranging data into the meaningful order so that we can analyze it more effectively.

→ Three type of sorting:-

a). Bubble sort.

b). Selection sort.

c) Insertion sort.

Q. Explain class and object with eg. in Java?

Ans:- Scanner sc = new Scanner (System.in);

① class, ② object

this keyword is used to define the object scanner ②.

→ So, Scanner is non-premitive datatype in Java.

Q. What is oops in Java and its principle?

Ans:- OOPS stands for object oriented programming structure.

→ It is a computer programming model that organizes software design around data, or objects, rather than functions and logic.

→ An object can be defined as ~~unique~~ data field that has unique attribute and behaviours.

Date _____ Page _____

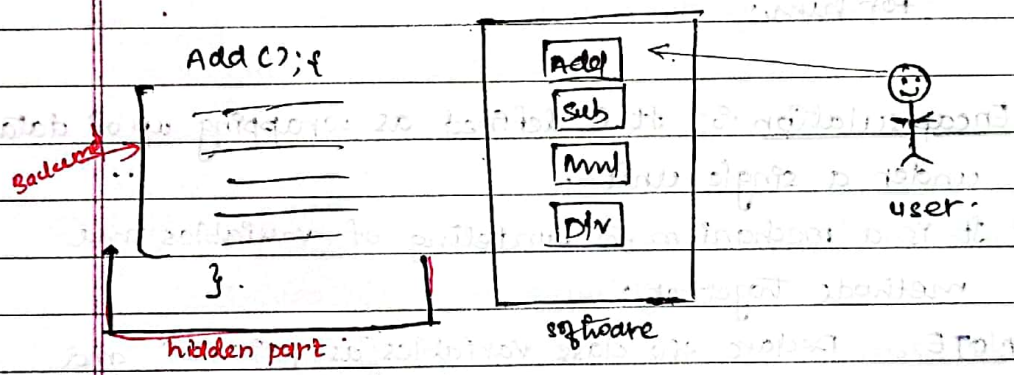
→ OOP is used to deal with real-world entities using the programming language.

→ It ~~an~~ aims is to construct a program using classes and objects.

→ It helps the programmer control and access the data in program easily.

a) **Abstraction** :- It is the process of hiding the ~~implementation~~ (implementation) unnecessary details from the users.

eg:-



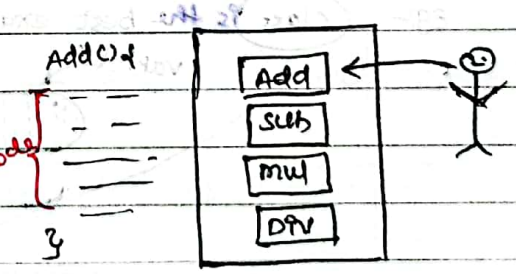
The user doesn't even know in which lang the code is written

→ **Advantage** :-

i) security

ii) Enhancement

eg:-



we can replace the code.

→ The developer knows that in which lang code is written but not the user.

→ If code is written in Java and it is taking more time to execute then we can replace the Java code with Python / C++ / C etc; and user will get good experience

→

-> Method is a block of code which only runs when it is called
-> we can pass the data, known as parameter into method.
-> methods are also known as function.

Date _____ Page _____

-> two way of abstraction :-

i) Abstract class (0 to 100%).

-> suppose we know the 20% of work then we will get 80% of the work done by someone else.

ii) Interface (100%) :- the total requirement is of client.

-> Its 100% implementation will be done by other user.

ie client don't have any knowledge about software devlopement he will tell his requirement to developer and developer will create the software for him.

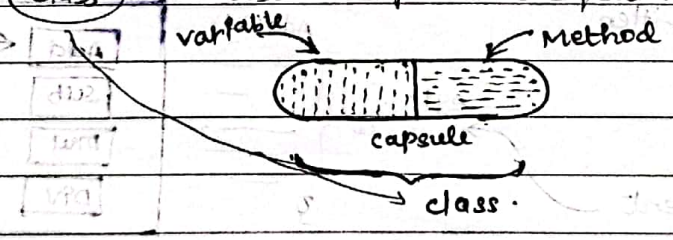
b) Encapsulation :- It is defined as wrapping up of data under a single unit.

-> It is a mechanism to bundeling of variables and methods together.

NOTE :- Declare the class variables as private and

-> Declare the class methods as public.

eg:- class is the best example of encapsulation.



-> Encapsulation provides the control over data and are easy to test so, it is better for unit testing.

```
-> public class EncapTest {
    private String name;
    private String idNum;
    private int age;
```

note:- get value देता और set value को set करके देता।

→ obj should be madeup of subclass coz it uses the all features of superclass.

Date _____ Page _____

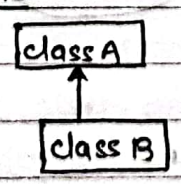
```
public int getAge () {
  return age ; }
public string getName () {
  return name ; }
public string getIdNum () {
  return idNum ; }
public void setAge ( int newAge ) {
  age = newAge ; }
public void setName ( String newName ) {
  name = newName ; }
public void setIdNum ( String newId ) {
  idNum = newId ; } }.
```

c) **Inheritance** :- It is a mechanism where a new class is driven from an existing class.

→ A class driven from another class is called subclass, ~~and~~ whereas the class from which a subclass is derived is called a superclass.

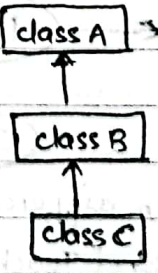
→ Method overriding is possible and it ~~provides~~ is used for code reusability.

i) ~~single~~



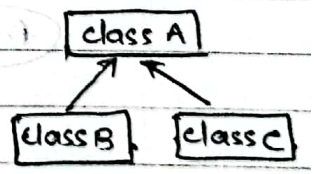
single

ii)



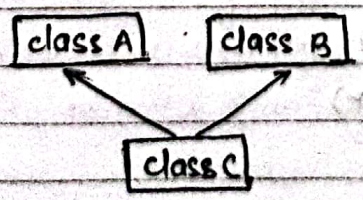
multilevel

iii)



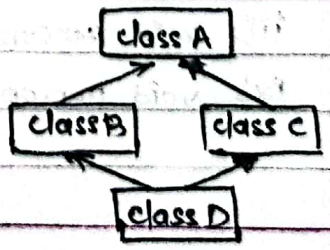
Hierarchical

iv)



Multiple

v)



Hybrid.

Date _____ Page _____

Note:- In Java extends keyword is used to perform inheritance.

→ It provide code reusability.

→ we can't access private members of class through inheritance.

→ A subclass contains all the features of superclass so, we should create the object ~~the~~ of subclass.

→ **syntax:**

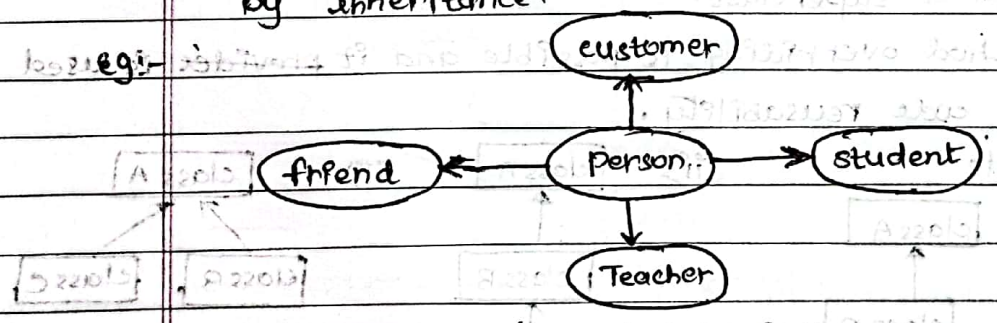
```
class A {                    // superclass
    ========
    ========   // +, - //
    ========
}

class B extends A {          // subclass
    ========
    ========   +, -
    ========
    ========
    ========   // *, /, % //
    ========
}
```

class B can all features of class A and add them with new features and can launch.

⇒ +, -, *, /, %

d) **Polymorphism:-** means "many forms", and it occurs when we have many classes that are related to each other by inheritance.

eg:-



same person different behaviour.

i) void person (Teacher)

ii) void person (student)

iii) void person (friend)

iv) void person (customer)

→ Two type of polymorphism :-

i) compile time polymorphism is a type of polymorphism that is resolved during compilation process.

→ It is obtained through the method overloading. which allows us to have more than one eg method with same name.

ii) Runtime polymorphism is also known as Dynamic method Dispatch, is a process ~~that~~ in which call is not resolved by the compiler. Ex:- method overriding.

→ It depends on the runtime not on compile time.

eg:-
```
class Bike {
        void run() {syso("running");}
    }class Splendor extends Bike {
        void run() {syso("running safely with 60km");}
        public static void main(String args[]){
            Bike b = new Splendor();  //upcasting//
            b.run();
        }
    }
```

o/p :- running safely with 60km.

Q. What is packages in Java?

Ans:- package is a group of similar type of classes, interface and sub-packages.

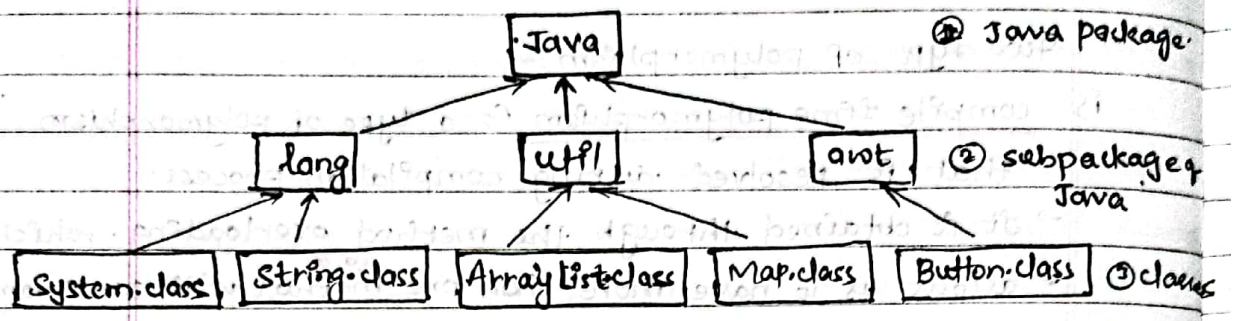
→ It can be categorized ~~by~~ into two forms, built in package and user-defined package.

→ Advantage:-

→ Used to categorized the classes and interfaces so that they can be easily maintained.

→ Java package provide access protection.

→ It removes naming collision.



Eg:- Java Package:-

→ Package keyword is used to create a package in Java.

// save as simple.Java //

```
Package.mypack ;
public class simple {
    public static void main (String args[]) {
        syso ("Welcome to package") ;
    }
}
```

→ Use the below syntax

→ To compile:- Java c-d . simple.Java ; (directory, filename)

→ To run :- Java mypack.Simple

o/p Welcome to package.

→ -d is switch which tells compiler where to put class file, i.e. it represents the destination.

→ Three ways to access package from outside the package.

a) import. package.* ;

b) import package. classname;

c) fully qualified name.

~~a) Import. package.* :~~ :-

a) Using package name :- If you use package.*, then all the classes and interfaces of this package will be accessible but not subpackages.

→ Import keyword is used to make classes and interfaces of another package accessible to the current package.

eg:

```
//save by A.Java//
package pack;
public class A{
Public void msg(){sysO("Hello");}
}
```

```
//save by B.Java//
Package mypack;
Import pack.*;
class B{
public s v m (String args[]){
A obj = new A();
obj.msg();}
}
```

o/p: Hello.

b) Using packagename.classname :- If you import the package.classname then only declared class of ~~these~~ this package will be accessible.

eg:-

```
//Saved by A.Java//
Package pack;
public class A{
public void msg(){sysO("Hello");}
}
```

```
// saved by B.Java //
Package mypack;
import pack.A;
class B{
public s v m (String args[]){
A obj = new A();
obj.msg();}}.
```

o/p:- Hello.

c) Using fully qualified name:- If you use fully qualified name then only declared class of this package will be accessible. Now there is no need to import. But you need to use fully qualified name every time when you are accessing the class and interface.

→ It is ~~normally~~ generally used when two packages have same class name, eg:- Java.util and Java.sql packages contains Data class.

Eg:-

```
// save by A.java //
package pack;
public class A {
public void msg() {sysO("Hello");}
}
```

```
// save by ~~B.java~~ B.Java //
Package mypack;
class B {
public s v m (String args[]) {
Pack.A obj = new pack A();  //using fully qualified name//
obj.msg();
}
}
```

O/P:- Hello.

Q. What is subpackage?

Ans:- package inside the package is called subpackage.

→ It should be created to categorized the package further.

Q. Why override annotation is used?

Ans:- It indicates that the child class method is over-writing its base class method.

→ It extracts the warning from the compiler ~~if the~~ the annotation method doesn't actually override anything.

→ It can improve the ~~reliability of~~ readability of the source code.

Q. What is instanceof operator in Java?

Ans:- → It is a binary operator we use to test if an object is of a given type. The result of this operator is either true or false.

→ It is also known as a type comparison operator because it compares the instance with the type.

→ Before casting an unknown object, the instanceof check should always be used.

Date _____ Page _____

Q. What do you mean by interface in Java?

Ans:- In Java an interface is a blueprint of a class.

→ It is the mechanism to achieve abstraction.

→ To access the interface methods, the interface must be implemented by another class with the implements keyword.

Q. What is super key and its use in Java?

Ans:- The super keyword refers to superclass (parent) objects.

→ It is used to call superclass methods, and to access the superclass constructor.

→ The most use of it is, to eliminate the confusion between superclasses and subclasses that have methods with the same name.

Q. Why pointer is not used in Java?

Ans:- Because using pointer the memory area can be directly accessed, which is security issue. pointers need so much memory spaces at the runtime and to reduce the uses of memory spaces Java don't use the pointer.

Q What is the use of GET and POST method?

Ans:- GET → It request the data from specified resource. GET method is used to append form data to URL in name or value pair. If you use GET, the length of URL will remain limited. It helps user to submit the bookmark the result.

POST :- It submit the processed data to a specified resources.

CamScanner