Java (Theory)

Name: Teresa Jency Bala
ID: 19 385 20 113

Course Code: CSE1221

Department: CSE

Campus: Imperial College of Engineering, Khulna
Affiliated by Rajshahi University (385)

Date: 17, Oct, 2019

Day: Wednesday.

1. What is Java?

Java is a high level programming language which is used to create softwares for multiple platforms. Which means java applications doesn't depend on OS i.e they are platform independent, and are easy in terms of implementation. Java applies the concept of Object Oriented programming. Java is simple and easy to learn. Java syntaxes are clearly understable. Java is highly secure, roboust (strong memory management) and portable (for creating bytecode).

2. What is Java Platform? (IDE)

Java platform is a collection of programs that help to develop and run programs written in the Java programming language. Java platform includes an execution engine, a compiler and a set of libraries. All these

are included in Java's own platform. For this java programs doesn't need any specific processor or operating-system.

3. What are the key features of Java?

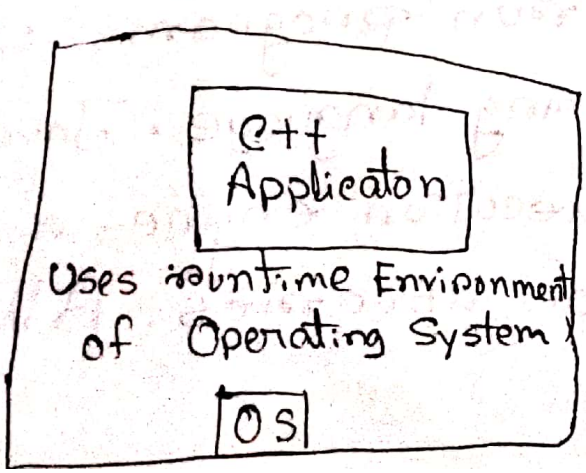
Java has some important features which make it so unique language. These are-

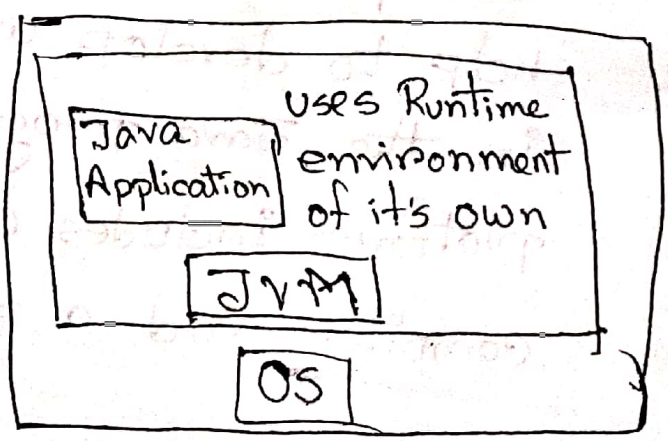
① Simple : Java codes are easy to understand If anyone read through the codes he can understand what it indicates.

② Secure : Java is best known for its security. With Java, virus-free systems can be developed.

Java is secured because :

- No explicit pointer.
- Java programs run inside a virtual machine sandbox.





③ Robust: ~~Java is strong~~ Robust simply means strong. Java because it has strong memory management. There is no use of pointers to avoid security issues. ~~and more.~~

④ Portable: Java is portable because it facilitates to carry Java bytecode to any platform. It doesn't require any implementation.

⑤ Architecture-neutral: As there are no dependency of Java on the Operating system It's architecture neutral. The sizes of primitive data are fixed.

⑥ Dynamic: Java has runtime polymorphism.

⑦ Object-Oriented: Java is O.O.P language. Every-thing in Java is an object. O.O.P means the software is organized by combination of objects that incorporates both data (value) and behavior (funtionality).

⑧ Multithreaded: ~~Java~~ It doesn't occupy memory for each thread. It shares a common memory area. Threads are important for

multi-media, web application etc.

(9) High-performance: Java bytecode was carefully designed so that it would be easy to translate directly into native machine code.

(10) Interpreted: Java enables cross-platform programs by compiling into an intermediate representation called Java bytecode.

(11) Distributed: It facilitates users to create distributed application in Java. RMI (Remote method invocation) feature invoke methods across a network (from any machine on Internet).

(12) Platform independent: After compiling Java program once it can be executed on any platform (OS) by means on JVM.

4. Where are Java used ? (Application of Java).

Java is used to create many web applications also a lot more.

(a) Android Apps: Written in Java programming language, with Google Android API (application program interface) which is similar to JDK (Java development kit).

(b) Web application: ~~like~~ Many govt. healthcare, insurance, education, defence & several other department have their web app built in Java using Servlets, JSP etc.

(c) Embedded System

(d) Smart cards

(e) Servers applications for banks and industry

(f) Games

(g) Robotics.

(5) Compare among C, C++ and Java.

Comparism topic	C	Java C++	Java C++
1. Platform dependency	Dependent	Independent	Dependent.
2. Programming approach	Procedural Programming Language	OOP, Generic Programing language	Procedural, OOP, Generic Programming language.
3. Compiled Source Code	Executable in Native Code	Compiled into Java byte code.	Executable in Native Code
4. Memory Management	Manually done	Managed using a garbage collector	Manually don
5. Pointer	Commonly used	No pointers, references are used instead	Used and some form of reference available too.
6. Inheritance	Not applicable	Single class inheritance, multiple interface implementation	Multiple class inheritance.
7. Operator Overloading	Not applicable	No	Yes.
8. Goto Statement	Yes	No.	Yes
9. Interpreter & Compiler	Compiler only ~~each time~~ used	Both Compiler & Interpreter used	Complier only user

1. What is Java?

Java is a high level programming language which is used to create software for.

Comparism topic	C	Java	C++
(10) Library called by.	#Including header files	Importing class names	Including header files
(11) Mainly Used	System programming	Application programing Window, web-based, enterprise and mobile application	For System programming

5. Difference among JDK, JRE & JVM.

Topic	JDK	JRE	JVM
(1) Stands for.	Java development kit	Java runtime environment	Java virtual Machine
(2) Defined for	It is tool necessary to compile, document and package Java program	JRE refers to a runtime environment in which Java byte-code can be executed. Contains libraries.	It is an abstract machine. It is a specification Provides runtime environment for compiling source code to bytecode and execute it.
(3) Works	Contains JRE + development tools	Implementation of JVM & physically exists	Follows: Specification, Implementation Runtime Instance

6. What is final keyword?

Final keyword is used to restrict user from make any variable, changing values of variable, method or class. unchangeable.

i.e. it makes it fixed.

Final variable's values are constant.

Final methods cannot be overridden but can be inherited.

Final class cannot be extended.

7. What is block?

A block is a group of individual Java statements enclosed in braces ({}). Each statement inside block is executed sequentially.

8. Why Java called language of internet?

Java allows object to move freely in Cyberspace Java applet is an application designed to be transmitted over internet and executed by Java compatible web browser. Java is server side as well as client side language, run at both side, so easy to make a site.

9. Difference ~~amon~~ betw Array & Arraylist

Topic	Array	Arraylist
1) Size	We need to know size of array to create it.	No need to know size.
2) Static/ Dynamic	Static	Dynamic
3) Part of	Part of core Java Programming	Part of collection of framework with other classes.
4) Datatypes	Can contains primitives or objects	Can store both objects and Generics
5) Insertion and deletion of elements	Manually write logic.	Use Done by calling methods.
6) Speed	Fast	Slow

10, How to create arraylist, add and remove elements.

Arraylist is a generic class which extends Abstractlist and implements List interface.

To create an arraylist.

```
Arraylist<String> list = new ArrayList<String>();
```

To add element,

```
list.add ("NameOfElement");
list.add (1, "SecondElement");
```

↑ Index no.

To remove an element:

```
list.remove("Name Of Element");
list.remove(1);
          ↳ Index no.
```

A simple program to of using arraylist:

```
import java.util.List;
import java.util.ArrayList;
class ArrayListDemo{
   public static void main(String arg[]){
    ArrayList<String> al = new ArrayList<String>();
   // List<Integer> al = new ArrayList<>();
    al.add("A");
    al.add("E");
    al.add("I");
    al.add(1, "E2");
    al.remove(2);
    al.remove("A");
   System.out.println("al: " + al);
   }
}
```

Output: al : E2

11. Differenciate Class and Object.

Object	No	Class
Object is an instance of a class	1	Class is a blue print from which objects are created.
Object is a real world entity such as chair, table, laptop etc	2	Class is a group of similar objects.
Object is physical entity	3	Class is a logical entity
Object is created many times as per requirement	4	Class is declared once
Object allocates memory when it is created	5	Class doesn't allocate memory when it is created
Object created through new keyword. Employee ob = new Employee();	6	Class is declared using class keyword. class Employee { }

12. Difference between ~~Or~~ Overloading and Overraiding.

Topic	Method Overloading	Method Overriding
Parimeter ①	Must be ~~same~~ different	Must be same.
Class ②	Occurs in the same class	Occurs between two classes - sub class and super class
Inheritance ③	Not involved	Inheritance involved
Hidding ④	One method does not hide another	Child method hides its parent method.
Polymorphism ⑤	Compile time polymorphism	Runtime polymorphism

13. What is static method and variable?

When we declare a method static (non-access modifier), then it's a static method. The coding advantage of static is that ther is no need to create object to invoke the static method

The static keyword placed before a variable makes it static. These are created when program starts and destroyed when program stops. They are declared Public (available to access) And are called by name.

ClassNam.VariableName.

14. What are primitive and non primitive datatypes?

Primitive: The datatype which is predefined by the language and named as keyword There are 8 major Primitive datatypes. These are - byte, short, int, long, float, double, char, boolean.
8 16 32 64 32 64 8 1 bit

Syntax example: int num=50;

Non-Primitive: Also known as reference variable are created using defined constructors of the classes. These variables are declared to be of a specific type that cannot be changed. Objects of various classes like String and array. come under the reference datatype. Syntax example:

String
array
class
interface

~~String s1 = new~~

array-var-name = new type [size].

months. = new int [12]

15. Write about access modifiers in Java.

Classes and packages are both means of encapsulating and containing variables and methods.

Packages contains classes and other subordinate packages. Classes contains data & code

Java address four categories of visibility for class memberes.

- Subclass in same package
- Non-Subclasses in same package.
- Subclass in different package.
- Classe that are neither in same package and Subclasses.

The access modifiers are Private, Protected and Public. And default state for no Modifier.

The access levels:-

Area	Private	NO Modifier	Protected	Public
Same class	Yes	Yes	Yes	Yes
Same package subclass	No	Yes	Yes	Yes
Same package non-subclass	No	Yes	Yes	Yes
Different package subclass	No	No	Yes	Yes
Different package non-subclass	No	No	No	Yes.

16. What is local class?

A local class is declared locally within a block of Java code, rather than as a member of a class. Typically it is defined within a method, but it can also be defined within a static initializer or instance of a class

17. How Java keeps memory garbage free?

~~Java handles deallocated memory auto~~
Java handles deallocation automatically. The technique that accomplish the lea releasing of memory is garbage collection. When no reference to an object exists, the object is assumed to be no longer needed, and the memory occupied by the object can be reclaimed. There is no explicit need to destroy object as in other languages.

18. Why is Java ~~OO~~ Object oriented? What are basic concepts of OOP?

~~Java~~ In Java Everything is taken as an object. We organize our software as a combination of different types of objects that incorporates both data and behavious.

Basic concepts of OOP are:

1. Object
2. Class
3. Inheritance
4. Polymorphism
5. Abstraction
6. Encapsulation.

19. What is ~~constractor?~~ default constructor?

~~The constructor is a~~

The ~~the~~ non-parameterized method whose name is ~~same~~ exactly same as the name of the class it is in is called default constructor. Default constructors are used to initialize create objects of a class upon it's creation. They are created with new operator. It has no return type, not even void. Example:

```
class Box{
    double width;
    double height;
    Box(){
    }
```

20. What is constractor Overloading?

The mathods within a same class having same name as the class but have different parameter list is known as constractor Over loading.

21. What is object?

An object is an instance of a class Objects have states and behaviours. It is a real world entity.

The three characteristics of an object:

→ state (value). → Behaviour (method)

→ Identity.

22. What are the 3 ways to initialize object?

An object can be initialized by any of the following 3 ways.

① By reference variable

② By method

③ By constructor.

23. What are the different ways to create object in Java?

The ways to create objects are-

① By new keyword.

② By newInstance() method

③ By clone() method.

④ By deserialization

⑤ By factory method

24. What are the differences between for, while and do-while loops ?

For	While	Do-While

1. Syntax -
for(initialization;
condition;
increment / decrement)
{
//body
}
	1: While (logical condition)
	{
	//body of loop
	}
		1. do{
		//statements
		//body of loop
		}
		while (condition);

2. Entry controlled loop	Entry controlled loop	Exit controlled loop
	2 loop executed after testing the logical condition	3. loop executed at least once before checking condition
	4 There is no semicolon at the end of while statement.	4. There is a semicolon at the end of while statement.

25. What are the different types of variables in Java ?

Variable: Variable is the name of reserved area allocated in memory. In

other words, it is a name of memory location. It is a combination of "vary+able" that means it's value can be changed.

Variables in Java are of 3 types.

① Local (local to method and visible in that only)

② Instance (Object level. defined in instance level)

③ Static (Class level)

① Local variable: ⓐ Declared inside body of method.
ⓑ Can be used only in that method.
ⓒ Can not be defined with "static" keyword.
ⓓ Implemented in stack level internally

② Instance variable: ⓐ Declared inside class but outside body of methods.
ⓑ It's value is instance (object) specific. and not shared among instances.
ⓒ Created ~~with~~ when an object is created with new keyword.
ⓓ Acces modifiers can be given for instance variables.
ⓔ The instance variables are visible for all methods, constructors and block in the class.
ⓕ Have default value (zero) & for boolean (false)

③ Static variable: ⓐ Variable which is declared with ~~as~~ static keyword.

ⓑ Cannot be local. Created outside method, constructor or a block.

26. How can you obtain an Array from ArrayList?

To create an array from an arraylist we have to use the toArray( ) method. As array has a fixed size. So, we need to know the size of arraylist too. A simple program is given below to obtain array from ArrayList.

```
import java.util.ArrayList;
import java.util.List;
class ArrayFromArrayList{
    public static void main(String[] args){
        List<Integer> list = new ArrayList<>();
        list.add(1);
        list.add(2);
        list.add(3);
        Integer[] intArr = new Integer[list.size()];
        list.toArray(intArr);
```

```
for (Integer. i : intArr) {
    System.out.println (i);
}
```

The command toArray has transformed the ArrayList to array.

27. What are difference between constructor & method ?

Java Constructor	Java Method
1. Used to initialize the state of an object	1. Used to expose the behavior of an object.
2. Must not have a return type	2. Must have a return type.
3. Is invoked implicitly	3. Is invoked explicitly
4. Java compiler provides a default constructor if you don't have any constructor in a class.	4. The method is not provided by the compiler in any case
5. The constructor name must be same as the class name.	5. The method name ~~may~~ can not be same as class name.

28. What is static method ?

- This method belong to the class rather than the object.
- There is no need to create the object to call the static method.
- A static method can access and change the value of the static variable.

29. Why is main method static?

Because the object is not required to call the static method. If we make the main method non-static, JVM will have to create its object first and then call main() method which will lead to the extra memory allocation.

~~main~~ Closable:

```
mainframe.addWindowListener(new WindowAdapter(){
  {
    public void windowClosing(WindowEvent e){
        System.exit(0);
    }
});
```