CS6501- INTERNET PROGRAMMING

UNIT I

PART –A

1. Write a java program that displays the result of floating point division.

```
import java.io.*;

class sample

{

public static void main(String args[])

{

float a=4.2,b=2.1;

System.out.println("Division:a/b="+(a/b));

}

}
```

OUTPUT:

Division:a/b=2.0

2. Differentiate packages and interface.

BASIS FOR COMPARISON	**PACKAGES**	**INTERFACES**
Basic	Packages is a group of classes and/or interfaces together.	Interfaces is a group of abstract methods and constant fields.
Keyword	Packages are created using "Package" keyword.	Interface are created using "Interface" keyword.
Syntax	package package_name; public class	interface interface_name{ variable declaration;

	class_name{	method declaration;

BASIS FOR COMPARISON	PACKAGES	INTERFACES
	. (body of class) . }	}
Access	A package can be imported	An interface can be extended by another interface and implemented by the class.
Access keyword	Packages can be imported using "import" keyword.	Interfaces can be implemented using "implement" keyword.

3. Define an abstract class. Give example.

A super class that does nothing, but lists the features of the other classes are called abstract class. It is a class that cannot be instantiated. Abstract keyword is used. It may contain abstract methods. If a class has at least one abstract method, then that class must be declared abstract.

Example:

abstract class sample

{

abstract void meth1();

void meth2()

{

System.out.println(“Method2”);

} }

4. List the unique features of Java.

- Simple and object oriented
- Platform independent, Robust and secure
- Architecture neutral and portable
- High performance
- Interpreted, threaded, dynamic

5. Give the different types of casting in java with examples.

Assigning a value of one type to a variable of another type is known as **Type Casting**.

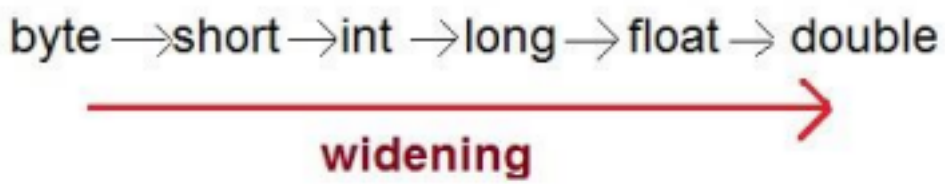
Example :

int x = 10;

byte y = (byte)x;

In Java, type casting is classified into two types,

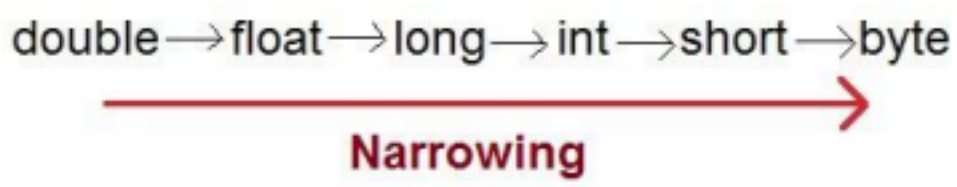
- Widening Casting(Implicit)



Example:
int a=10;

float b=a; // implicit conversion

- Narrowing Casting(Explicitly done)



Example:

double d=100.04;

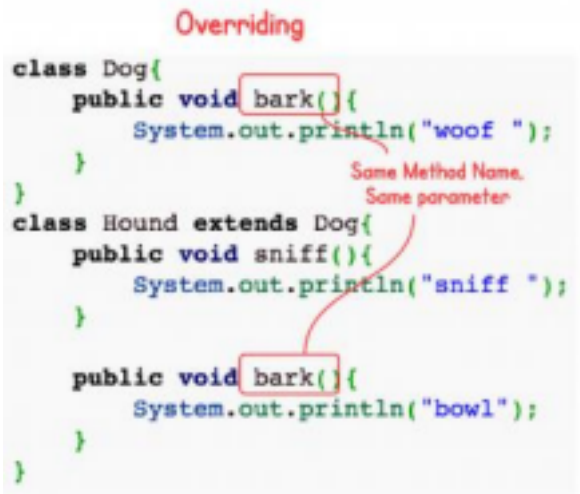
int a=(int)d; // explicit type conversion

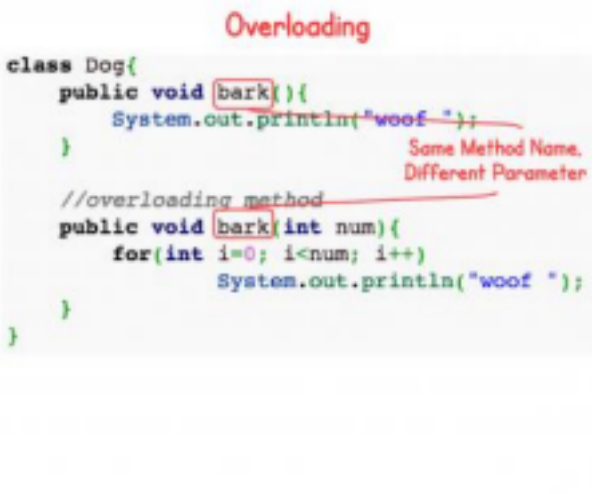
6. Mention the way in which the current state of a thread can be obtained.

In Java, to get the current state of the thread, use **Thread.getState()** method to get the current state of the thread.

7. Differentiate overloading and overriding of methods in java.

- Using the same method number of times in the same class but with different parameters is known as method overloading.
- If the super class and subclass have the same method, we say, the super class method is **overridden** by subclass. In overriding, super class and subclass will have the **same method with same parameters and same return type.**





8. Give an example of a chained exception in java.

Chained Exceptions allows to relate one exception with another exception, i.e one exception describes cause of another exception.

- **getCause()** method returns the actual cause associated with current exception.
- **initCause()** set an underlying cause(exception) with invoking exception.

```
import java.io.IOException;
public class ChainedException
 {
 public static void divide(int a, int b)
 {
 if(b==0)
 {
 ArithmeticException ae = new ArithmeticException("top layer");
ae.initCause( new IOException("cause") );
 throw ae;
 }
 else
 {
 System.out.println(a/b);
 }
 }


 public static void main(String[] args)
 {
 try {
 divide(5, 0);
 }
 catch(ArithmeticException ae) {
 System.out.println( "caught : " +ae);
 System.out.println("actual cause: "+ae.getCause());
 }
 }
```

```
}
```

OUTPUT:

caught:java.lang.ArithmeticException: top layer

actual cause: java.io.IOException: cause

9. Write a java program that reads an integer number as input from the user and determines whether it is palindrome or not.

```
        import java.io.*;
class sample
{
public static void main(String args[])
{
int a,r,temp,rev;
System.out.println(“Enter a
number:”); Scanner in=new
Scanner(System.in); a=in.nextInt();
temp=a;
while(temp!=0)
{
r=temp%10;
rev=rev*10+r;
temp=temp/10;
}
if(rev==a)
        System.out.println(“Palindrome”);
else
        System.out.println(“Not a
```

Palindrome"); }

}

Output:

Enter a number:

343

Palindrome

10. Give the differences between abstract class and interface.

Abstract class	**Interface**
1) Abstract class can **have abstract and non-abstract**methods.	Interface can have **only abstract** methods. Since Java 8, it can have **default and static methods** also.
2) Abstract class **doesn't support multiple inheritance**.	Interface **supports multiple inheritance**.
3) Abstract class **can have final, non-final, static and non-static variables**.	Interface has **only static and final variables**.
4) Abstract class **can provide the implementation of interface**.	Interface **can't provide the implementation of abstract class**.
5) The **abstract keyword** is used to declare abstract class.	The **interface keyword** is used to declare interface.
6) An **abstract class** can extend another Java class and implement multiple Java interfaces.	An **interface** can extend another Java interface only.
7) An **abstract class**can be extended using keyword ?extends?.	An **interface class**can be implemented using keyword ?implements?.
8) A Java**abstract class**can have class members like private, protected, etc.	Members of a Java interface are public by default.

9)**Example:**
public abstract class Shape{
public abstract void draw();
}
	Example:
	public interface Drawable{
	void draw();
	}

11. What is the use of break and continue statements?

Break:

- Break statement is used to break the execution of the current executing loop or block.
- Statements after the break statement will not be executed.

Continue:

- Continue statement is used to skip the current iteration of the current executing loop or block.
- Statements after the continue statement will be executed.

Break	**Continue**
for(int i = 0; i < 5; i++)	for(int i = 0; i < 5; i++)
{	{
if (i = = 3) break;	if (i = = 3) continue;
System.out.println(i);	System.out.println(i);
}	}
O/p:-	**O/p:-**
0	0
1	1
2	2
	4

12. What is an exception? Bring out its types.

Exceptions are events that occur during the execution of programs that disrupt the normal flow of instructions (e.g. divide by zero, array access out of bound, etc.).

- **Checked exception**: an exception that occurs at the compile time, cannot simply be ignored Ex: IOException, SQLException
- **Unchecked exception**: an exception that occurs at the time of execution including programming bugs. Ex: NullPointerException, ArrayIndexOutOfBoundsException
- **Error:** It is irrecoverable. It arises beyond the control of the user

13. What is the use of finally keyword in Java?

"finally" block provides assurance of execution of some important code that must be executed after the try block. If an exception happens at try block, its execution gets break off. "finally" block executes finally at the end. It is sometimes called as "clean-up" code.

Eg.

```
class sample

{

void meth1()

{
…

try

{ }

finally

{

 }

}
```

14. What are the OOP principles?

The OOP principles are:

•Objects

•Classes

•Data abstraction and encapsulation

•Inheritance

•Polymorphism

•Dynamic binding

•Message passing

15. Define – Class and Object

Class is defined as a template for a set of objects that share a common structure and a

common

behaviour. Object is an instance of a class. It has state, behaviour and identity. It is also called as an instance of a class.

16.What is meant by instance?

An instance has state, behaviour and identity. The structure and behaviour of similar classes are defined in their common class. An instance is also defined in their called as an object.

17.Define – Inheritance

Inheritance is defined as a relationship among classes, wherein one class shares the structure or behaviour defined in another class. This is called Single Inheritance. If a class shares the structure or behaviour from multiple classes, then it is called Multiple Inheritance. Inheritance defines “is-a” hierarchy among classes in which one subclass inherits from one or more generalised super classes.

18. Define – Polymorphism

Polymorphism literally means taking more than one form. Polymorphism is defined as characteristic of being able to assign a different behaviour or value in a subclass, to something that was declared in a parent class.

19.What is meant by interface?

Interface is similar to a class which may contain methods signature only but not bodies and it is a formal set of method and constant declarations that must be defined by the class that implements it.

Syntax:

interface interfacename

{

//declaration of variables and methods

}

20.What are the different states of a thread?

The different states of a thread are

•New

•Runnable

•Waiting

•TimedWaiting

•Terminated

21. List out the motivation needed in generic programming.

The motivation needed in generic programming are :

•To motivate generic methods, that contains three overloaded print Array methods

•This methods print the string representations of the elements of an integer array, double array and character array

•Choose to use arrays of type Integer, double and character to setup our generic methods

22. What is meant by adapter class?

Adapter class is one which contains null body definition for the methods which are inheriting from appropriate listener.

An adapter class provides the default implementation of all methods in an event listener interface. Adapter classes are very useful when you want to process only few of the events that are handled by a particular event listener interface. Define a new class by extending one of the adapter classes and implement only those events relevant.

23. What is meant by synchronization and why is it important?

Synchronization is the capability to control the access of multiple threads to share resources without synchronization, it is possible for one thread to modify a share, the object while another thread is in the process of using of updating that object value this often leads to significant errors.

24. How is throw exception created in exception handling?

Throw method is used to throw an exception manually. It has to declare the exceptions thrown in the method signature and then include a throw statement in the method

throw new ExceptionName (“Value”);

25. Differentiate error from exception.

An Error indicates that a non-recoverable condition has occurred that should not be caught .Error is a subclass of throwable is intended for drastic problem, such as OutOfMemoryError,an exception is an event, which occurs during the execution of a program ,that disrupts the normal flow of the program.

26.What are the classes of exceptions caught by a catch clause?

A Throwble class is a super class .It includes all the classes whatever is covered from

catch class.

27.How is custom exception created?

By extending the exception class or one of its subclasses

Example

```
Class MyException extends Exception
{
Public MyException(){super();}
Public MyException(String s){super(s);}
}
```

28.What are the different ways to handle exceptions?

There are two ways to handle Exceptions

• Wrapping the desire code in a try block followed by a catch block to catch the exception

• List the desired exception in the throws clause of the method and let the caller of the method handle those exceptions.

29. What is meant by multithreading?

Multithreading extends the idea of multitasking into applications where you can subdivide specific operations within a single application into individual threads. Each of the threads can run in parallel. The OS divides processing time not only among different applications, but also among each thread within an application.

30.What method must be implemented by all threads?

All tasks must implement the run() method, whether they are a subclass of thread or implement the runnable interface. It can be called using the start() method.

.