Assignment 3

1.Create a class Publication with data members title(String) and price(int). From this class derive two classes Book and CD. Class Book adds pages(int) and CD adds Size(int). Each of these classes should have constructors and display(). Write a java program to implement this using super, this and method overriding concepts.

import java.util.\*;

class Publications

{

String title;

int price;

Publications(String title,int price)

{

this.title=title;

this.price=price;

}

void display()

{

System.out.println("The title is "+title+" price"+price);

}

}

class Book extends Publications

{

int page;

Book(String title,int price,int page)

{

super(title,price);

this.page=page;

}

void display()

{

System.out.println("The book title is "+title+" price "+price+" pages "+page);

}

}

class CD extends Publications

{

int size;

CD(String title,int price,int size)

{

super(title,price);

this.size=size;

}

void display()

{

System.out.println("The book title is "+title+" price "+price+" size "+size);

}

}

class Hierarchical

{

public static void main(String[] args)

{

Publications p=new Publications("java",2000);

p.display();

Book b=new Book("java Book",200,300);

b.display();

CD c=new CD("java CD",400,45);

c.display();

}

}

OUTPUT:

The title is java price2000

The book title is java Book price 200 pages 300

The book title is java CD price 400 size 45

2. Write a simple java program to demonstrate method overriding.

class Override{

public void print()

{

System.out.println("Hello");

}

}

class Override2 extends Override{

public void print(){

System.out.println("Hii");

}

public static void main( String args[]) {

Override2 obj = new Override2();

obj.print();

}

}

3. Write a java program to create an interface called Shape with CalculateArea(). Create three classes namely Square,Circle,Triangle which implements Shape.

interface Shape

{

void CalculateArea();

}

class Square implements Shape

{

public void CalculateArea()

{

int s=4;

System.out.println("The area of square is "+s\*s);

}

}

class Triangle implements Shape

{

public void CalculateArea()

{

int b=3,h=9;

System.out.println("The area of Triangle is "+(b\*h)/2);

}

}

class Circle implements Shape

{

public void CalculateArea()

{

float r=4.2f;

System.out.println("The area of circle is "+3.14\*r\*r);

}

}

class Interface

{

public static void main(String[]args)

{

Square s=new Square();

s.CalculateArea();

Triangle s1=new Triangle();

s1.CalculateArea();

Circle s3=new Circle();

s3.CalculateArea();

}

}

OUTPUT:

The area of square is 16

The area of Triangle is 13

The area of circle is 55.38959496917736

4. Create two packages p1 and p2. The package p1 contains class A which contains one display(). Create class B in package p2. The main method of class B invoke A’s display(). Write a java program to do this.

package p1;

public class A

{

public void display()

{

System.out.println("It is in class A of package p1");

}

}

package p2;

import p1.\*;

class B

{

public static void main(String args[])

{

A a=new A();

a.display();

}

}

OUTPUT:

It is in class A of package p1

5. Write a java program to count numbers, characters in the command line arguments using Exception handling mechanism.

public class CommandLine {

public static void main(String args[]) {

try{

int d=0,c=0;

for(int i=0;i<args.length;i++){

if(Character.isLetter(args.charAt(i)))

c++;

else

d++;

}

System.out.println("COUNT OF CHARACTERS "+c+" DIGITS "+d);

}

catch(Exception e){

System.out.println(e);

}

}

}

output:

COUNT OF CHARACTERS 6 DIGITS 6

THEORY

1. What is Inheritance?

Inheritance can be defined as the process where one class acquires the properties(methods and fields) of another.The class which inherits the properties of other is known as subclass (derived class, child class) and the class whose properties are inherited is known as superclass (base class, parent class).

2. What is Multiple Inheritance?

“Multiple Inheritance” refers to the concept of one class extending (Or inherits) more than one base class. The problem with “multiple inheritance” is that the derived class will have to manage the dependency on two base classes.In java ,mulple inheritance can be achieved using interfaces.

3. What is the use of Super keyword?

The super keyword in Java is a reference variable which is used to refer immediate parent class object.Whenever you create the instance of subclass, an instance of parent class is created implicitly which is referred by super reference variable.

Usage of Java super Keyword

1.super can be used to refer immediate parent class instance variable.

2. super can be used to invoke immediate parent class method.

3.super() can be used to invoke immediate parent class constructor.

4. What is abstract method?

A method without body (no implementation) is known as abstract method. A method must always be declared in an abstract class, or in other words you can say that if a class has an abstract method, it should be declared abstract as well.

5. What is abstract class?

A class that is declared using “abstract” keyword is known as abstract class. It can have abstract methods(methods without body) as well as concrete methods (regular methods with body).An abstract class ca not be instantiated, which means you are not allowed to create an object of it.

6. What is the use of final modifier?

The final is a modifier in Java, which can be applied to a variable, a method or a class.

• When a final modifier is used with a class then the class cannot be extended further.

This is one way to protect your class from being subclassed and often sensitive

classes are made final due to security reason.

• When the final keyword is used with a method that it cannot be overridden in

Java, which means you cannot override the logic of the method in the subclass.

• When the final keyword is used with a variable then its value cannot be changed

once assigned.

7. What is interface? Write the syntax interface.

An interface in Java is a blueprint of a class. It has static constants and abstract methods.The interface in Java is a mechanism to achieve abstraction. There can be only abstract methods in the Java interface, not method body. It is used to achieve abstraction and multiple inheritance in Java.

Syntax:

interface <interface\_name>{

// declare constant fields

// declare methods that abstract

// by default.

}

8. What is package?

A package as the name suggests is a pack(group) of classes, interfaces and other packages. In java we use packages to organize our classes and interfaces. We have two types of packages in Java: built-in packages and the packages we can create (also known as user defined package).

9. What is exception?

Exception is an error event that can happen during the execution of a program and disrupts it’s normal flow. Exception can arise from different kind of situations such as wrong data entered by user, hardware failure, network connection failure etc.Whenever any error occurs while executing a java statement, an exception object is created and then JRE tries to find exception handler to handle the exception.

10. What is the use of finally block?

The finally block will execute when the try/catch block leaves the execution, no matter what condition cause it. It always executes whether the try block terminates normally or terminates due to an exception. The main purpose of finally block is to release the system resources. The finally block follows try/catch block.