1. public class triangle {

     double l;

     double b;

     double h;

}

public triangle(){

        double l = 3;

        double b = 4;

        double h = 5;

}

public double getarea(){

    double a = (l + b + h)/2;

    double c = a \* (a - l) \* (a - b) \* (a - h);

    double area = math.sqrt(c);

    system.out.println("area   "+area);

}

public double getperimeter(){

    double perimeter = l + b + h;

    system.out.println("perimeter   "+perimeter);

}

public class areaandperimeter{

            public static void main(string[] args){

              triangle scalene = new triangle();

              scalene.getperimeter();

              scalene.getarea();

               }

}

2.import java.util.scanner;

   public class Adddistance {

     public int a;

     public int b;

}

public Adddistance(){

        int a = 0 ;

        int b = 0;

}

public int getdistance(int c){

   scanner sc = new scanner(system.in);

   system.out.println("enter distance"+c+"in feet");

   a = input.nextint();

   system.out.println("enter distance"+c+"in inches");

   b = input.nextint();

   return a,b;

}

public void addiction(adddistance d,adddistance e){

    this.a = d.a + e.a;

    this.b = d.b + e.b;

    if (this.b >= 12) {

        this.a += this.b / 12;

        this.b %= 12;

    }

   system.out.println("distance in feet is "+a);

   system.out.println("distance in inches is "+b);

}

public class Adddistance{

            public static void main(string[] args){

             int adddistance d = new adddistance();

             int adddistance e = new adddistance();

             int adddistance f = new adddistance();

             d.getdistance(1);

             e.getdistance(2);

             f.adddistance(d,e);

               }

}

3.   public class employee{

       private int id;

       private string name;

       private int age;

       private string disignation;

 public int getid(){

     return id;}

 public void setid(int id){

     this.id = id;}

public string getname(){

     return name;}

 public void setname(string name){

     this.name = name;}

public int getage(){

     return age;}

 public void setage(int age){

     this.age = age;}

public string getdesignation(){

     return designation;}

 public void setdesignation(string designation){

     this.designation = designation;}}

public class employee{

   public static void main(string[] args){

   employee details = new empolyee();

   details.setid(1);

   details.setname("sruthin");

   details.setage(22);

   details.setdestination("engineer");

   system.out.println(""+details.getid());

   system.out.println(""+details.getname());

   system.out.println(""+details.getage());

   system.out.println(""+details.getdesignation("lead engineer"));}}

4.public class rectangle {

   public int length;

   public int width;

public rectangle() {

     this.length = 0;

     this.width = 0;}

public Rectangle(int length, int width) {

     this.length = length;

     this.width = width;}

public int getlength() {

     return length;}

public void setLength(int length) {

     this.length = length;}

public int getwidth() {

     return width;}

public void setwidth(int width) {

     this.width = width;}

public void getarea() {

    int area =  length \* width;

    system.out.println(area);}}

public class rectangle{

   public static void main(string[] args){

    rectangle obj = new rectangle(2, 3);

    obj.getarea();}}

5. 1)  output : play-clean-play-

   the class toy will be created and the memory will allotted for car object. then it calls the method of play ,it will print the string "play-" . then it calls the system.gc() ,it will calls the remaining methods which is not executed rest of the main method calls till. so it calls fanilizers ,it will print the string "clean-". again the main function create a new memory for doll object . then it again calls the play method , it will print the string "play-" . it doesn't had a ln function ,so it print the output in a single line.

2) output: parent

                child

                child

   in the main function creates a memory allocation for x in parent node as a parent ,also creates a memory allocation for y in parent node as child and it aslo creates a memory allocation for z in child node as a child. when printmain method line code is executed, it takes a line to execute the printmain method function and it executes the print method line code . when print method is called for x ,it checks the condition of "it is a parent node or child node".because there are two methods having a same name "print" ,but both of them having a different class. when it confirms the node is parent, it will execute the parent class's method so it print "parent". when the execution goes to printmain(y) , it is a child in parent node,so it considered as child. so it executes the child class method ,so it prints the "child". then printmain(z) aslo had a same case but child is present in the child node ,so it executes child lass method. so it prints the "child".

3) output : play-clean-play-

   the class toy will be created and the memory will allotted for car object. then it calls the method of play ,it will print the string "play-" . then it calls the system.gc() ,it will calls the remaining methods which is not executed rest of the main method calls till. so it calls fanilizers ,it will print the string "clean-". again the main function create a new memory for doll object . then it again calls the play method , it will print the string "play-" . it doesn't had a ln function ,so it print the output in a single line.

both first and third cases are same , so i gave the same answer

6. 1) No, it is not possible .Encapsulation hides the object specifics, whereas abstraction covers the implementation details. The object is an abstraction of reality, and encapsulation is used to obscure specifics. As a result, abstraction requires encapsulation.

 2) Abstraction

   Abstraction, often known as the process of withdrawing or deleting something superfluous, is a generic notion created by extracting common elements from individual cases.

   Encapsulation

   Code and the data it manipulates are linked by a method called encapsulation, which protects both against interference from outside sources and unauthorised usage.

3) Encapsulation

   Code and the data it manipulates are linked by a method called encapsulation, which protects both against interference from outside sources and unauthorised usage.

   Polymorphism

   it is a process of saving a same data in different form,when there are several classes that share an inheritance relationship.