**Question No:1**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,

C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.

Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.\*;

class HotelRoom

{

String hotelName;

int noOfSqClaw;

int ratePerSqClaw;

boolean hasWifi;

boolean hasAC;

public HotelRoom(String name,int area,int rate,String wifi,String ac)

{

this.hotelName=name;

this.noOfSqClaw=area;

this.ratePerSqClaw=rate;

if(wifi.equals("Yes"))

{

this.hasWifi=true;

}

else

{

this.hasWifi=false;

}

if(ac.equals("Yes"))

{

this.hasAC=true;

}

else

{

this.hasAC=false;

}

}

}

class DeluxeRoom extends HotelRoom

{

public DeluxeRoom(String name,int area,int rate,String wifi,String ac)

{

super(name,area,rate,wifi,ac);

}

void display()

{

System.out.println(hotelName);

if(hasWifi==true)

{

System.out.println("Amount:"+(noOfSqClaw\*ratePerSqClaw)+10);

}

else

{

System.out.println("Amount:"+(noOfSqClaw\*ratePerSqClaw));

}

}

}

class DeluxeACRoom extends DeluxeRoom

{

String hotelName;

int noOfSqClaw;

int ratePerSqClaw;

boolean hasWifi;

boolean hasAC;

public DeluxeACRoom(String name,int area,int rate,String wifi,String ac)

{

super(name,area,rate,wifi,ac);

this.hotelName=name;

this.noOfSqClaw=area;

this.ratePerSqClaw=rate;

if(wifi.equals("Yes"))

{

this.hasWifi=true;

}

else

{

this.hasWifi=false;

}

if(ac.equals("Yes"))

{

this.hasAC=true;

}

else

{

this.hasAC=false;

}

}

void display()

{

if(hasAC==true)

{

System.out.println("Amount:"+(noOfSqClaw\*ratePerSqClaw)+50+10);

}else

{

System.out.println("Amount:"+(noOfSqClaw\*ratePerSqClaw)+10);

}

}

}

class SuiteACRoom extends HotelRoom

{

public SuiteACRoom(String name,int area,int rate,String wifi,String ac)

{

super(name,area,rate,wifi,ac);

}

void display()

{

System.out.println(hotelName);

System.out.println("Amount: "+(noOfSqClaw\*ratePerSqClaw));

}

}

public class Main

{

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

System.out.println("1. Deluxe Room");

System.out.println("2. Deluxe AC Room");

System.out.println("3. Luxury Suite");

int choice=s.nextInt();

int area,rate;

String wifi,ac;

String name;

switch(choice)

{

case 1:

System.out.println("Enter hotel name: ");

name=s.next();

System.out.println("Enter room area: ");

area=s.nextInt();

System.out.println("Enter rate/sq claw: ");

rate=s.nextInt();

System.out.println("Hotel has Wifi? ");

wifi=s.next();

System.out.println("Hotel has AC?");

ac=s.next();

DeluxeRoom dl=new DeluxeRoom(name,area,rate,wifi,ac);

dl.display();

break;

case 2:

System.out.println("Enter hotel name: ");

name=s.next();

System.out.println("Enter room area: ");

area=s.nextInt();

System.out.println("Enter rate/sq claw: ");

rate=s.nextInt();

System.out.println("Hotel has Wifi? ");

wifi=s.next();

System.out.println("Hotel has AC?");

ac=s.next();

DeluxeACRoom dc=new DeluxeACRoom(name,area,rate,wifi,ac);

dc.display();

break;

case 3:

System.out.println("Enter hotel name: ");

name=s.next();

System.out.println("Enter room area: ");

area=s.nextInt();

System.out.println("Enter rate/sq claw: ");

rate=s.nextInt();

System.out.println("Hotel has Wifi? ");

wifi=s.next();

System.out.println("Hotel has AC?");

ac=s.next();

SuiteACRoom su=new SuiteACRoom(name,area,rate,wifi,ac);

su.display();

}

}

}

**Question No:2**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,

C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.

Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.\*;

class Dinosaur

{

String species;

String consumptionType;

public Dinosaur (String species, String consumption)

{

this.species = species;

this.consumptionType = consumption;

}

public void display1()

{

System.out.println ("Dino Report");

System.out.println ("Base Dino Properties");

System.out.println (species + " is a "+consumptionType);

}

}

class LavaDinosaur extends Dinosaur

{

private char canBreatheFire;

private char hasHeatResistance;

public LavaDinosaur(String species,String consumption,char fire,char heat)

{

super(species,consumption);

this.canBreatheFire=fire;

this.hasHeatResistance=heat;

}

void display()

{

System.out.println(" Dino Report ");

System.out.println(" Lava Dino Properties ");

if(canBreatheFire=='y'&&hasHeatResistance=='y')

{

System.out.println(species+" breathes fire ");

System.out.println(" It is resistant to heat ");

}else if(canBreatheFire=='n'&&hasHeatResistance=='y')

{

System.out.println(species+" dosn 't breathes fire");

System.out.println("It is resistant to heat");

}else if(canBreatheFire=='y'&&hasHeatResistance=='n')

{

System.out.println(species+"breathes fire");

System.out.println("It dosn' t resist to heat ");

}else

{

System.out.println(species+" dosn 't breathes fire");

System.out.println("It dosn' t resist to heat ");

}

System.out.println(" Inherited Base Dino Properties ");

System.out.println(species+" is a "+consumptionType);

}

}

class DragonDinosaur extends Dinosaur

{

private int numberOfWings;

private char canBreatheFire;

private char hasHeatResistance;

private int numberOfScales;

public DragonDinosaur (String species, String consumption, char fire,char heat, int wing, int scales)

{

super (species, consumption);

this.canBreatheFire=fire;

this.hasHeatResistance=heat;

this.numberOfWings=wing;

this.numberOfScales=scales;

}

void display()

{

System.out.println("Dino Report");

System.out.println("Dragon Dino Properties");

System.out.println("Flies with "+numberOfWings+" wings");

System.out.println("Has "+numberOfScales+" scales");

System.out.println("Inherited Dino Properties");

if(canBreatheFire=='y'&&hasHeatResistance=='y')

{

System.out.println(species+" breathes fire ");

System.out.println(" It is resistant to heat ");

}else if(canBreatheFire=='n'&&hasHeatResistance=='y')

{

System.out.println(species+" dosn 't breathes fire");

System.out.println("It is resistant to heat");

}else if(canBreatheFire=='y'&&hasHeatResistance=='n')

{

System.out.println(species+"breathes fire");

System.out.println("It dosn' t resist to heat ");

}else

{

System.out.println(species+" dosn 't breathes fire");

System.out.println("It dosn' t resist to heat ");

}

System.out.println(" Inherited Base Dino Properties ");

System.out.println(species+" is a "+consumptionType);

}

}

public class Main

{

public static void main (String[]args)

{

Scanner s = new Scanner (System.in);

System.out.println ("1. Dinosaur");

System.out.println ("2. Lava Dinosaur");

System.out.println ("3. Dragon Dinosaur");

System.out.println ("Enter your choice");

String species, consumption;

int ch = s.nextInt ();

char fire, heat;

int wing, scales;

switch (ch)

{

case 1:

System.out.println ("Enter the species");

species = s.next ();

System.out.println ("Enter consumption: ");

consumption = s.next ();

Dinosaur d = new Dinosaur (species, consumption);

d.display1();

break;

case 2:System.out.println ("Enter the species");

species = s.next ();

System.out.println ("Enter consumption: ");

consumption = s.next ();

System.out.println ("Breathes fire? (y/n)");

fire = s.next ().charAt (0);

System.out.println ("Is heat resistant? (y/n)");

heat = s.next ().charAt (0);

LavaDinosaur lava = new LavaDinosaur (species, consumption, fire, heat);

lava.display();

break;

case 3:System.out.println ("Enter the species");

species = s.next ();

System.out.println ("Enter consumption: ");

consumption = s.next ();

System.out.println ("Breathes fire? (y/n)");

fire = s.next ().charAt (0);

System.out.println ("Is heat resistant? (y/n)");

heat = s.next ().charAt (0);

System.out.println ("Enter no. of wings:");

wing = s.nextInt ();

System.out.println ("Enter no. of scales: ");

scales = s.nextInt ();

DragonDinosaur drag =new DragonDinosaur (species, consumption, fire, heat,wing, scales);

drag.display();

}

}

}

**Question No:3**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,

C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.

Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.\*;

class LavaDinosaur

{

char canBreatheFire;

char hasHeatResistance;

public LavaDinosaur(char fire,char heat)

{

this.canBreatheFire=fire;

this.hasHeatResistance=heat;

}

}

class DragonDinosaur extends LavaDinosaur

{

int numberOfWings;

int numberOfScales;

char pricanFly;

public DragonDinosaur(int numberOfWings,int numberOfScales,char pricanFly,char fire,char heat)

{

super(fire,heat);

this.numberOfWings=numberOfWings;

this.numberOfScales=numberOfScales;

this.pricanFly=pricanFly;

}

}

class SharkDinosaur extends LavaDinosaur

{

int numberOfFins;

int numberOfGills;

public SharkDinosaur(int numberOfFins,int numberOfGills,char fire,char heat)

{

super(fire,heat);

this.numberOfFins=numberOfFins;

this.numberOfGills=numberOfGills;

}

}

class LizardDinosaur extends LavaDinosaur

{

int numberOfClaws;

int numberOfLegs;

char canRun;

public LizardDinosaur(int numberOfClaws,int numberOfLegs,char canRun,char fire,char heat)

{

super(fire,heat);

this.numberOfClaws=numberOfClaws;

this.numberOfLegs=numberOfLegs;

}

}

class DragonDinosaurv2 extends DragonDinosaur

{

String abilityName;

String abilityDescription;

public DragonDinosaurv2(String abilityName,String abilityDescription,int numberOfWings,int numberOfScales,char pricanFly,char fire,char heat)

{

super(numberOfWings,numberOfScales,pricanFly,fire,heat);

this.abilityName=abilityName;

this.abilityDescription=abilityDescription;

}

}

class SharkDinosaurv2 extends SharkDinosaur

{

String abilityName;

String abilityDescription;

public SharkDinosaurv2(String abilityName,String abilityDescription,int numberOfFins,int numberOfGills,char fire,char heat)

{

super(numberOfFins,numberOfGills,fire,heat);

this.abilityName=abilityName;

this.abilityDescription=abilityDescription;

}

}

class LizardDinosaurv2 extends LizardDinosaur

{

String abilityName;

String abilityDescription;

public LizardDinosaurv2(String abilityName,String abilityDescription,int numberOfClaws,int numberOfLegs,char canRun,char fire,char heat)

{

super(numberOfClaws,numberOfLegs,canRun,fire,heat);

this.abilityName=abilityName;

this.abilityDescription=abilityDescription;

}

}

class DragonDinosaurv3 extends DragonDinosaurv2

{

String superPowerName;

String superPowerDescription;

public DragonDinosaurv3(String superpower,String superPowerDescription,String abilityName,String abilityDescription,int numberOfWings,int numberOfScales,char pricanFly,char fire,char heat)

{

super(abilityName,abilityDescription,numberOfWings,numberOfScales,pricanFly,fire,heat);

this.superPowerName=superpower;

this.superPowerDescription=superPowerDescription;

}

void display()

{

System.out.println("Dino report");

System.out.println("superPower"+superPowerName);

System.out.println("superPowerDescription"+superPowerDescription);

System.out.println("ability type"+abilityName);

System.out.println("abilityDescription"+abilityDescription);

System.out.println("numberOfWings"+numberOfWings);

System.out.println("numberOfScales"+numberOfScales);

if(canBreatheFire=='y'&&hasHeatResistance=='y')

{

System.out.println(superPowerName+"can breathe fire");

System.out.println(superPowerName+"has resistent to heat");

}else if(canBreatheFire=='y'&&hasHeatResistance=='n')

{

System.out.println(superPowerName+"can breathe fire");

System.out.println(superPowerName+"has no resistent to heat");

}else if(canBreatheFire=='n'&&hasHeatResistance=='y')

{

System.out.println(superPowerName+"can not breathe fire");

System.out.println(superPowerName+"has resistent to heat");

}else

{

System.out.println(superPowerName+"can not breathe fire");

System.out.println(superPowerName+"has no resistent to heat");

}

}

}

class SharkDinosaurv3 extends SharkDinosaurv2

{

String superPowerName;

String superPowerDescription;

public SharkDinosaurv3(String superpower,String superPowerDescription,String abilityName,String abilityDescription,int numberOfFins,int numberOfGills,char fire,char heat)

{

super(abilityName,abilityDescription,numberOfFins,numberOfGills,fire,heat);

this.superPowerName=superpower;

this.superPowerDescription=superPowerDescription;

}

void display()

{

System.out.println("Dino report");

System.out.println("superPower"+superPowerName);

System.out.println("superPowerDescription"+superPowerDescription);

System.out.println("ability type"+abilityName);

System.out.println("abilityDescription"+abilityDescription);

System.out.println("Number of fins"+numberOfFins);

System.out.println("numberOfGills"+numberOfGills);

if(canBreatheFire=='y'&&hasHeatResistance=='y')

{

System.out.println(superPowerName+"can breathe fire");

System.out.println(superPowerName+"has resistent to heat");

}else if(canBreatheFire=='y'&&hasHeatResistance=='n')

{

System.out.println(superPowerName+"can breathe fire");

System.out.println(superPowerName+"has no resistent to heat");

}else if(canBreatheFire=='n'&&hasHeatResistance=='y')

{

System.out.println(superPowerName+"can not breathe fire");

System.out.println(superPowerName+"has resistent to heat");

}else

{

System.out.println(superPowerName+"can not breathe fire");

System.out.println(superPowerName+"has no resistent to heat");

}

}

}

class LizardDinosaurv3 extends LizardDinosaurv2

{

String superPowerName;

String superPowerDescription;

public LizardDinosaurv3(String superpower,String superPowerDescription,String abilityName,String abilityDescription,int numberOfClaws,int numberOfLegs,char canRun,char fire,char heat)

{

super(abilityName,abilityDescription,numberOfClaws,numberOfLegs,canRun,fire,heat);

this.superPowerName=superpower;

this.superPowerDescription=superPowerDescription;

}

void display()

{

System.out.println("Dino report");

System.out.println("superPower"+superPowerName);

System.out.println("superPowerDescription"+superPowerDescription);

System.out.println("ability type"+abilityName);

System.out.println("abilityDescription"+abilityDescription);

System.out.println("numberOfClaws"+numberOfClaws);

System.out.println("numberOfLegs"+numberOfLegs);

if(canBreatheFire=='y'&&hasHeatResistance=='y')

{

System.out.println(superPowerName+"can breathe fire");

System.out.println(superPowerName+"has resistent to heat");

}else if(canBreatheFire=='y'&&hasHeatResistance=='n')

{

System.out.println(superPowerName+"can breathe fire");

System.out.println(superPowerName+"has no resistent to heat");

}else if(canBreatheFire=='n'&&hasHeatResistance=='y')

{

System.out.println(superPowerName+"can not breathe fire");

System.out.println(superPowerName+"has resistent to heat");

}else

{

System.out.println(superPowerName+"can not breathe fire");

System.out.println(superPowerName+"has no resistent to heat");

}

}

}

public class Main

{

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

System.out.println("1.DragonDinosaur");

System.out.println("2.SharkDinosaur");

System.out.println("3.LizardDinosaur");

int ch;

String abilityName;

String abilityDescription;

int numberOfWings,numberOfFins,numberOfGills;

int numberOfScales;

char pricanFly,fire,heat;

int numberOfClaws;

int numberOfLegs;

char canRun;

String superpower,superPowerDescription;

System.out.println("Enter your choice");

ch=s.nextInt();

switch(ch)

{

case 1:

System.out.println("Enter super power name");

superpower=s.next();

System.out.println("Enter super Power Description");

superPowerDescription=s.next();

System.out.println("Enter ability type");

abilityName=s.next();

System.out.println("Enter ability discription");

abilityDescription=s.next();

System.out.println("Enter number of wings");

numberOfWings=s.nextInt();

System.out.println("Enter number of scales");

numberOfScales=s.nextInt();

System.out.println("Can fly(y/n");

pricanFly=s.next().charAt(0);

System.out.println("can breathe fire");

fire=s.next().charAt(0);

System.out.println("can resist heat");

heat=s.next().charAt(0);

DragonDinosaurv3 drag=new DragonDinosaurv3(superpower,superPowerDescription,abilityName,abilityDescription,numberOfWings,numberOfScales,pricanFly,fire,heat);

drag.display();

break;

case 2:

System.out.println("Enter super power name");

superpower=s.next();

System.out.println("Enter super Power Description");

superPowerDescription=s.next();

System.out.println("Enter ability type");

abilityName=s.next();

System.out.println("Enter ability discription");

abilityDescription=s.next();

System.out.println("Enter Number of fins ");

numberOfFins=s.nextInt();

System.out.println("Enter number of gills");

numberOfGills=s.nextInt();

System.out.println("can breathe fire");

fire=s.next().charAt(0);

System.out.println("can resist heat");

heat=s.next().charAt(0);

SharkDinosaurv3 shark=new SharkDinosaurv3(superpower,superPowerDescription,abilityName,abilityDescription,numberOfFins,numberOfGills,fire,heat);

shark.display();

break;

case 3:

System.out.println("Enter super power name");

superpower=s.next();

System.out.println("Enter super Power Description");

superPowerDescription=s.next();

System.out.println("Enter ability type");

abilityName=s.next();

System.out.println("Enter ability discription");

abilityDescription=s.next();

System.out.println("Enter numberOfClaws");

numberOfClaws=s.nextInt();

System.out.println("Enter number of legs");

numberOfLegs=s.nextInt();

System.out.println("can run (y/n)");

canRun=s.next().charAt(0);

System.out.println("can breathe fire");

fire=s.next().charAt(0);

System.out.println("can resist heat");

heat=s.next().charAt(0);

LizardDinosaurv3 lizard=new LizardDinosaurv3(superpower,superPowerDescription,abilityName,abilityDescription,numberOfClaws,numberOfLegs,canRun,fire,heat);

lizard.display();

}

}

}

**Question No:4**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,

C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.

Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.\*;

class TicketBooking

{

String source;

String destination;

String dateOfTravel;

int numberOfPassengers;

public TicketBooking(String source,String destination,String dateOfTravel,int numberOfPassengers)

{

this.source=source;

this.destination=destination;

this.dateOfTravel=dateOfTravel;

this.numberOfPassengers=numberOfPassengers;

}

}

class BusBooking extends TicketBooking

{

char isAC;

char isSleeper;

char hasWifi;

int totalcost;

public BusBooking(String source,String destination,String dateOfTravel,int numberOfPassengers,char isAC,char isSleeper,char hasWifi)

{

super(source,destination,dateOfTravel,numberOfPassengers);

this.isAC=isAC;

this.isSleeper=isSleeper;

this.hasWifi=hasWifi;

}

void display()

{

System.out.println("source"+source);

System.out.println("destination"+destination);

System.out.println("dateOfTravel"+dateOfTravel);

if(isAC=='y'&&isSleeper=='y'&&hasWifi=='y')

{

totalcost=300;

}else

if(isAC=='y'&&isSleeper=='n'&&hasWifi=='n'||isAC=='y'&&isSleeper=='y'&&hasWifi=='n'||isAC=='n'&&isSleeper=='y'&&hasWifi=='n')

{

totalcost=100;

}else if(isAC=='y'&&isSleeper=='y'&&hasWifi=='n'||isAC=='y'&&isSleeper=='n'&&hasWifi=='y'||isAC=='n'&&isSleeper=='y'&&hasWifi=='y')

{

totalcost=200;

}else

{

totalcost=0;

}

System.out.println("Total amount"+(numberOfPassengers\*500)+totalcost);

}

}

class TrainBooking extends TicketBooking

{

String coach;

int mealsOpted;

int cost;

public TrainBooking(String source,String destination,String dateOfTravel,int numberOfPassengers,String coach,int mealsOpted)

{

super(source,destination,dateOfTravel,numberOfPassengers);

this.coach=coach;

this.mealsOpted=mealsOpted;

}

void display()

{

System.out.println("source"+source);

System.out.println("destination"+destination);

System.out.println("dateOfTravel"+dateOfTravel);

if(coach.equals("A1"))

{

cost=500;

}else if(coach.equals("A2"))

{

cost=300;

}else

{

cost=100;

}

System.out.println("Total Amount: "+(numberOfPassengers\*300)+cost+(mealsOpted\*100));

}

}

class FlightBooking extends TicketBooking

{

String class1;

float luggageWeight;

public FlightBooking(String source,String destination,String dateOfTravel,int numberOfPassengers,String class1,float luggageWeight)

{

super(source,destination,dateOfTravel,numberOfPassengers);

this.class1=class1;

this.luggageWeight=luggageWeight;

}

void display()

{

System.out.println("source"+source);

System.out.println("destination"+destination);

System.out.println("dateOfTravel"+dateOfTravel);

System.out.println("Total amount: "+(numberOfPassengers\*1500)+(luggageWeight\*200));

}

}

public class Main

{

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

System.out.println("1.BusBooking");

System.out.println("2.TrainBooking");

System.out.println("3.FlightBooking");

System.out.println("Enter the choice");

String source,destination,dateOfTravel,coach,class1;

int mealsOpted;

float luggageWeight;

int numberOfPassengers;

int ch=s.nextInt();

char isAC,isSleeper,hasWifi;

switch(ch)

{

case 1:

System.out.println("Enter the source");

source=s.next();

System.out.println("Enter the destination");

destination=s.next();

System.out.println("dateOfTravel");

dateOfTravel=s.next();

System.out.println("numberOfPassengers");

numberOfPassengers=s.nextInt();

System.out.println("AC bus(y/n)");

isAC=s.next().charAt(0);

System.out.println("sleeper bus (y/n)");

isSleeper=s.next().charAt(0);

System.out.println("has wifi (y/n)");

hasWifi=s.next().charAt(0);

BusBooking bus=new BusBooking(source,destination,dateOfTravel,numberOfPassengers,isAC,isSleeper,hasWifi);

bus.display();

case 2:

System.out.println("Enter the source");

source=s.next();

System.out.println("Enter the destination");

destination=s.next();

System.out.println("dateOfTravel");

dateOfTravel=s.next();

System.out.println("numberOfPassengers");

numberOfPassengers=s.nextInt();

System.out.println("Enter the coach");

coach=s.next();

System.out.println("Number of meals opted");

mealsOpted=s.nextInt();

TrainBooking train=new TrainBooking(source,destination,dateOfTravel,numberOfPassengers,coach,mealsOpted);

train.display();

break;

case 3:

System.out.println("Enter the source");

source=s.next();

System.out.println("Enter the destination");

destination=s.next();

System.out.println("dateOfTravel");

dateOfTravel=s.next();

System.out.println("numberOfPassengers");

numberOfPassengers=s.nextInt();

System.out.println("Enter the class");

class1=s.next();

System.out.println("Enter the luggageWeight");

luggageWeight=s.nextFloat();

FlightBooking flight=new FlightBooking(source,destination,dateOfTravel,numberOfPassengers,class1,luggageWeight);

flight.display();

}

}

}

**Question No:5**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,

C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.

Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.\*;

abstract class Planet

{

String name;

public Planet(String name)

{

this.name=name;

}

abstract double calculateVolume();

}

class CublicalPlanet extends Planet

{

float side;

public CublicalPlanet(String name,float side)

{

super(name);

this.side=side;

}

double calculateVolume()

{

//side=length\*breadth\*heigth;

return side\*side\*side;

}

}

class SpericalPlanet extends Planet

{

float radius;

SpericalPlanet(String name,float radius)

{

super(name);

this.radius=radius;

}

double calculateVolume()

{

return (4\*3.14\*radius\*radius\*radius)/3f;

}

}

class ConicalPlanet extends Planet

{

float radius,heigth;

public ConicalPlanet(String name,float radius,float height)

{

super(name);

this.radius=radius;

this.heigth=height;

}

double calculateVolume()

{

return (3.14\*radius\*radius\*heigth);

}

}

class cuboidalPlanet extends Planet

{

float length,breadth,heigth;

public cuboidalPlanet(String name,float length,float breadth,float heigth)

{

super(name);

this.length=length;

this.breadth=breadth;

this.heigth=heigth;

}

double calculateVolume()

{

return length\*breadth\*heigth;

}

}

public class Main

{

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

System.out.println("1.Cubical Planet");

System.out.println("2.Spherical Planet");

System.out.println("3.Conical Planet");

System.out.println("4.Cuboial Planet");

System.out.println("Enter choice: ");

int ch=s.nextInt();

float length,breadth,heigth,radius,side;

String name;

switch(ch)

{

case 1:

System.out.println("Enter name: ");

name=s.next();

System.out.println("Enter side: ");

side=s.nextFloat();

CublicalPlanet cube=new CublicalPlanet(name,side);

System.out.println("Volume of cube:"+cube.calculateVolume()+"km3");

break;

case 2:

System.out.println("Enter name: ");

name=s.next();

System.out.println("Enter the radius");

radius=s.nextFloat();

SpericalPlanet sp=new SpericalPlanet(name,radius);

System.out.println("volume of sphere:"+sp.calculateVolume()+"km3");

break;

case 3:

System.out.println("Enter name: ");

name=s.next();

System.out.println("Enter radius: ");

radius=s.nextFloat();

System.out.println("Enter the height");

heigth=s.nextFloat();

ConicalPlanet cone=new ConicalPlanet(name,radius,heigth);

System.out.println("Volume of cone: "+cone.calculateVolume()+"km3");

break;

case 4:

System.out.println("Enter name: ");

name=s.next();

System.out.println("Enter length: ");

length=s.nextFloat();

System.out.println("Enter breadth: ");

breadth=s.nextFloat();

System.out.println("Enter height: ");

heigth=s.nextFloat();

cuboidalPlanet cuboid=new cuboidalPlanet(name,length,breadth,heigth);

System.out.println("volume of cuboid: "+cuboid.calculateVolume());

}

}

}

**Question No:6**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,

C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.

Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.\*;

import java.lang.\*;

abstract class Door

{

String doorName;

Door(String name)

{

this.doorName=name;

}

abstract void unlock();

}

class StarDoor extends Door

{

int numberOfStars;

public StarDoor(int numberOfStars,String name)

{

super(name);

this.numberOfStars=numberOfStars;

}

public void unlock()

{

int i,j,k;

int num=numberOfStars;

for(i=1;i<=numberOfStars;i++)

{

for(j=1;j<=num;j++)

{

System.out.print("\*");

}

System.out.println("");

num=num-1;

}

for(i=1;i<=numberOfStars;i++)

{

for(j=1;j<=i;j++)

{

System.out.print("\*");

}

System.out.println("");

}

}

}

class GasDoor extends Door

{

String nameOfGas;

public GasDoor(String name,String gasname)

{

super(name);

this.nameOfGas=gasname;

}

void unlock()

{

char[] a=nameOfGas.toCharArray();

int len=a.length;

char[] b=new char[len];

int j=len-1;

int i,flag=0;

for(i=0;i<len;i++)

{

b[j]=a[i];

j--;

}

for(i=0;i<len;i++)

{

if(a[i]==b[i])

{

flag=1;

}else

{

flag=0;

break;

}

}

if(flag==1)

{

System.out.println("Never say never. Palindrome, forever.");

}else

{

System.out.println("We cannot fish. Palindrome? You wish.");

}

}

}

class LiquidDoor extends Door

{

int number;

public LiquidDoor(String name,int number)

{

super(name);

this.number=number;

}

void unlock()

{

int i,k=2;

for(i=1;i<=number;i++)

{

System.out.println(Math.pow(2,k)-1);

k++;

}

}

}

public class Main

{

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

System.out.println("1.Star Door");

System.out.println("2.Gas Door");

System.out.println("3.Liquid Door");

System.out.println("Enter choice: ");

int ch=s.nextInt();

String name;

String gasname;

int numberOfStars,flow;

switch(ch)

{

case 1:

System.out.println("Enter the name");

name=s.next();

System.out.println("Enter numberOfStars: ");

numberOfStars=s.nextInt();

StarDoor star=new StarDoor(numberOfStars,name);

star.unlock();

break;

case 2:

System.out.println("Enter the name");

name=s.next();

System.out.println("Enter gas name");

gasname=s.next();

GasDoor gas=new GasDoor(name,gasname);

gas.unlock();

break;

case 3:

System.out.println("Enter the name");

name=s.next();

System.out.println("Enter flow");

flow=s.nextInt();

LiquidDoor liquid=new LiquidDoor(name,flow);

liquid.unlock();

}

}

}

**Question No:7**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,

C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.

Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.\*;

class Dinosaur

{

private String species;

private String locomotiontype;

public Dinosaur(String species,String land)

{

this.species=species;

this.locomotiontype=land;

}

public void display()

{

System.out.println("Dino details");

System.out.println("Species "+species);

if(locomotiontype.equals("Land"))

{

System.out.println("Travels by land");

}else

{

System.out.println("Travels by Air");

}

}

}

class LavaDino extends Dinosaur

{

private String heat;

public LavaDino(String species,String land,String heat)

{

super(species,land);

this.heat=heat;

}

public void heatResistence()

{

if(heat.equals("Yes"))

{

System.out.println("Lives on underground");

}else

{

System.out.println("Did not survive");

}

}

}

public class Main

{

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

System.out.println("Enter species: ");

String species=s.nextLine();

System.out.println("Enter type: ");

String land=s.nextLine();

System.out.println("Is it heat resistant? :");

String heat=s.next();

LavaDino dino=new LavaDino(species,land,heat);

dino.display();

dino.heatResistence();

}

}

**Question No:8**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,

C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.

Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.\*;

class LavaDino

{

private String species;

private String locomotionType;

int speed;

public LavaDino(String species,String type,int speed)

{

this.species=species;

this.locomotionType=type;

this.speed=speed;

}

public void displayDino()

{

System.out.println("Dino Details");

System.out.println("Species: "+species);

}

}

class AirLavaDino extends LavaDino

{

private int numberofwings;

private float speedperwing;

private float ashresistence;

public AirLavaDino(String species,String type,int speed,int wings,float speedperwing,float ashresistence)

{

super(species,type,speed);

this.numberofwings=wings;

this.speedperwing=speedperwing;

this.ashresistence=ashresistence;

}

public void displaySpeed()

{

System.out.println("Travels by air at a speed of "+(speed+(numberofwings\*speedperwing)-ashresistence));

}

}

class WateLavaDino extends LavaDino

{

private int numberoffins;

private float numberoftails;

private float lavamultiplier;

public WateLavaDino(String species,String type,int speed,int numberoffins,float numberoftails,float lavamultiplier)

{

super(species,type,speed);

this.numberoffins=numberoffins;

this.numberoftails=numberoftails;

this.lavamultiplier=lavamultiplier;

}

public void displaySpeed()

{

System.out.println("Travels by water at a speed of "+(speed+(numberoffins+numberoftails)\*lavamultiplier));

}

}

class LandLavaDino extends LavaDino

{

private int numberoflegs;

public LandLavaDino(String species,String type,int speed,int numberoflegs)

{

super(species,type,speed);

this.numberoflegs=numberoflegs;

}

public void displaySpeed()

{

System.out.println("Travels by land at a speed of "+(speed\*numberoflegs));

}

}

public class Main

{

public static void main(String[] args) {

Scanner s=new Scanner(System.in);

System.out.println("1. Land Dino");

System.out.println("2. Aqua Dino");

System.out.println("3. Aerial Dino");

String species,type;

int speed,wings,numberoflegs;

float speedperwing,ashresistence;

int numberoffins;

float numberoftails;

float lavamultiplier;

int choice=s.nextInt();

switch(choice)

{

case 1:

System.out.println("Enter species: ");

species=s.next();

System.out.println("Enter Type: ");

type=s.next();

System.out.println("Enter speed: ");

speed=s.nextInt();

System.out.println("Enter number of feets: ");

numberoflegs=s.nextInt();

LandLavaDino land1=new LandLavaDino(species,type,speed,numberoflegs);

land1.displayDino();

land1.displaySpeed();

break;

case 2:

System.out.println("Enter species: ");

species=s.next();

System.out.println("Enter Type: ");

type=s.next();

System.out.println("Enter speed: ");

speed=s.nextInt();

System.out.println("Enter no. of fins: ");

numberoffins=s.nextInt();

System.out.println("Enter no. of tails: ");

numberoftails=s.nextFloat();

System.out.println("Enter lava multiplier: ");

lavamultiplier=s.nextFloat();

WateLavaDino water=new WateLavaDino(species,type,speed,numberoffins,numberoftails,lavamultiplier);

water.displayDino();

water.displaySpeed();

break;

case 3:

System.out.println("Enter species: ");

species=s.next();

System.out.println("Enter Type: ");

type=s.next();

System.out.println("Enter speed: ");

speed=s.nextInt();

System.out.println("Enter no. of wings: ");

wings=s.nextInt();

System.out.println("Enter speed/wing: ");

speedperwing=s.nextFloat();

System.out.println("Enter ash resistance: ");

ashresistence=s.nextFloat();

AirLavaDino air=new AirLavaDino(species,type,speed,wings,speedperwing,ashresistence);

air.displayDino();

air.displaySpeed();

break;

}

}

}