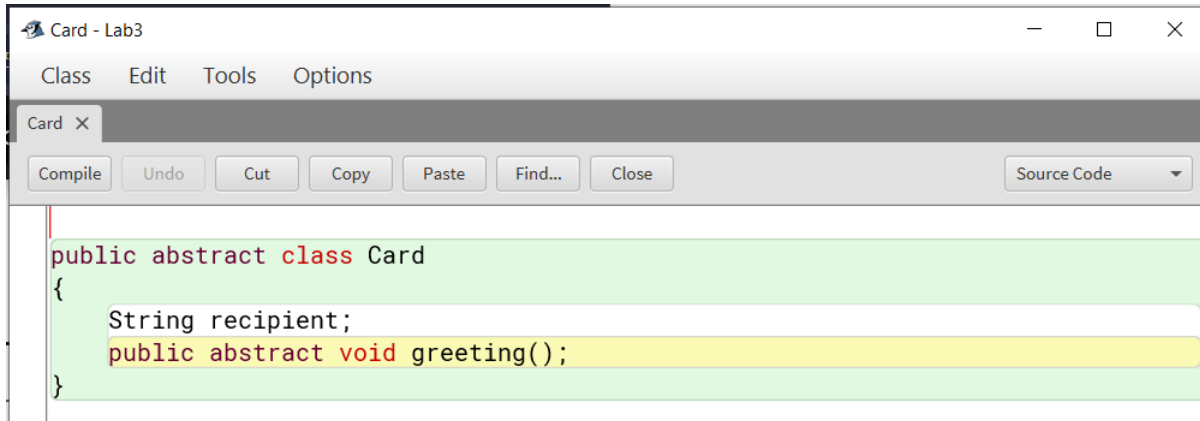


Lab 3 – Polymorphism

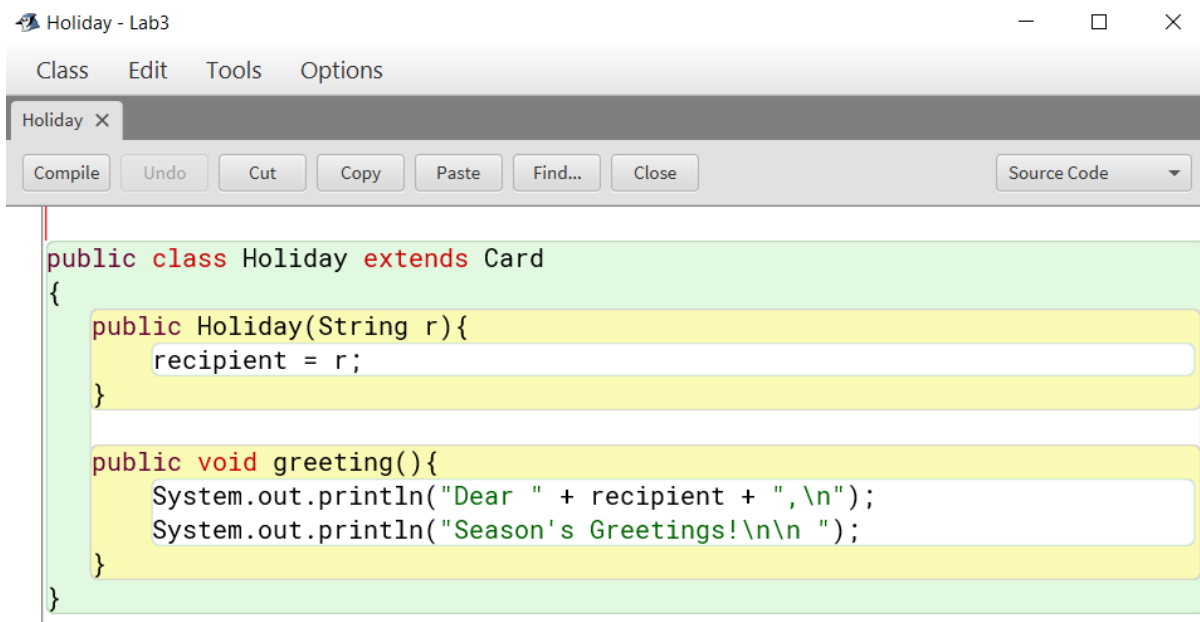
Answers:

1(a). Code and Output



```
public abstract class Card
{
    String recipient;
    public abstract void greeting();
}
```

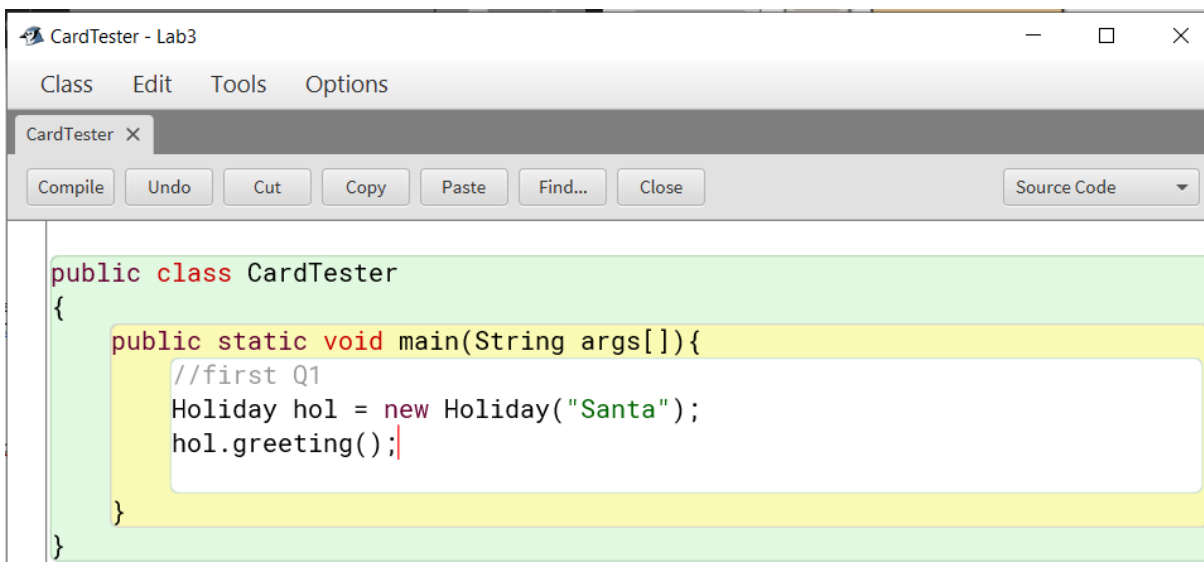
Figure 1: Card class



```
public class Holiday extends Card
{
    public Holiday(String r){
        recipient = r;
    }

    public void greeting(){
        System.out.println("Dear " + recipient + ",\n");
        System.out.println("Season's Greetings!\n\n ");
    }
}
```

Figure 2: Holiday class

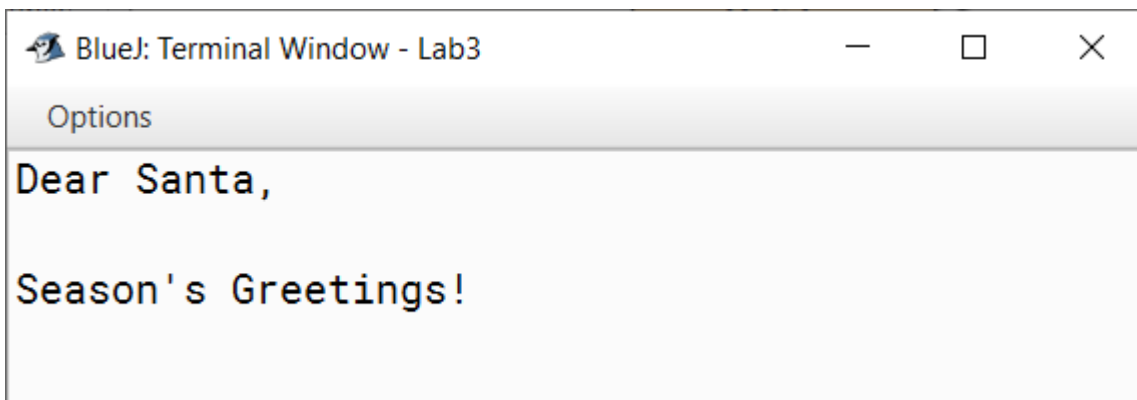


The screenshot shows the BlueJ IDE window titled "CardTester - Lab3". The menu bar includes "Class", "Edit", "Tools", and "Options". Below the menu bar is a toolbar with buttons for "Compile", "Undo", "Cut", "Copy", "Paste", "Find...", and "Close". A "Source Code" dropdown menu is also present. The main area displays the source code for the `CardTester` class, which is highlighted in green. The code defines a `main` method, which is highlighted in yellow. Inside the `main` method, there is a comment `//first Q1`, followed by the creation of a `Holiday` object named `hol` with the string `"Santa"`, and a call to `hol.greeting()`.

```
public class CardTester
{
    public static void main(String args[]){
        //first Q1
        Holiday hol = new Holiday("Santa");
        hol.greeting();
    }
}
```

Figure 3: CardTester class

The output:



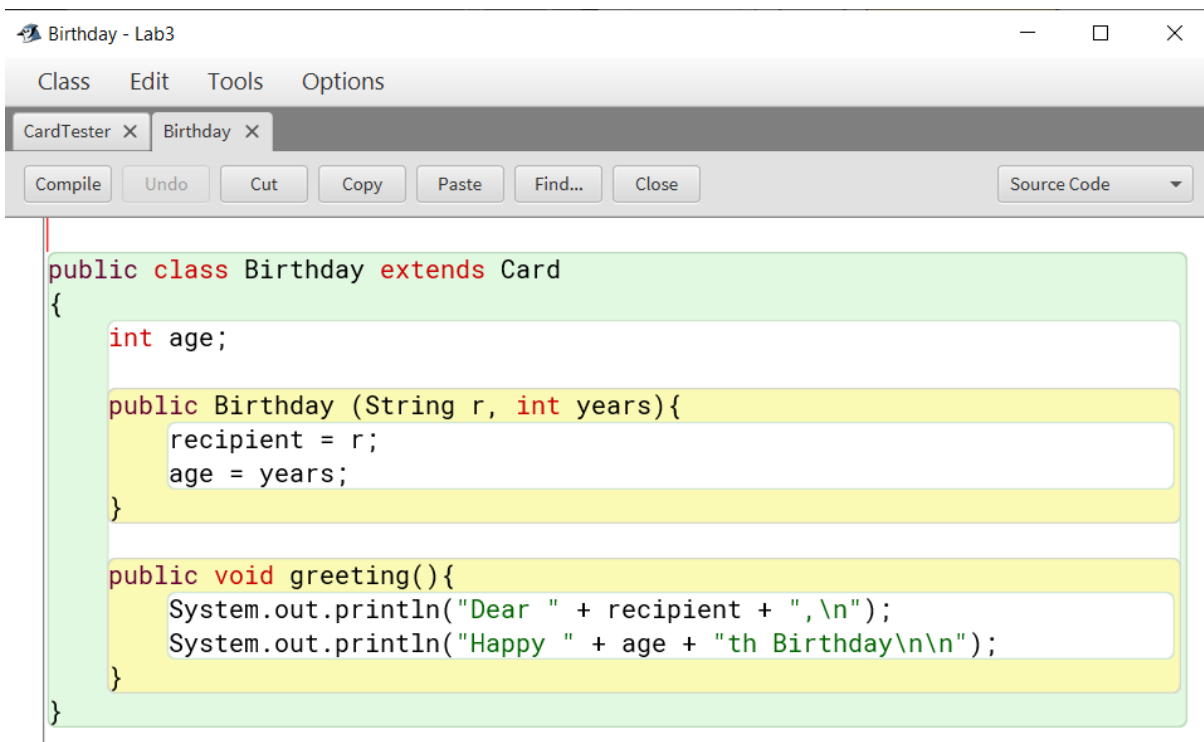
The screenshot shows the BlueJ IDE window titled "BlueJ: Terminal Window - Lab3". The menu bar includes "Options". The main area displays the output of the `CardTester` class, which is "Dear Santa," followed by "Season's Greetings!" on a new line.

```
Dear Santa,

Season's Greetings!
```

Figure 4: Output question 1 (a)

1(b) :



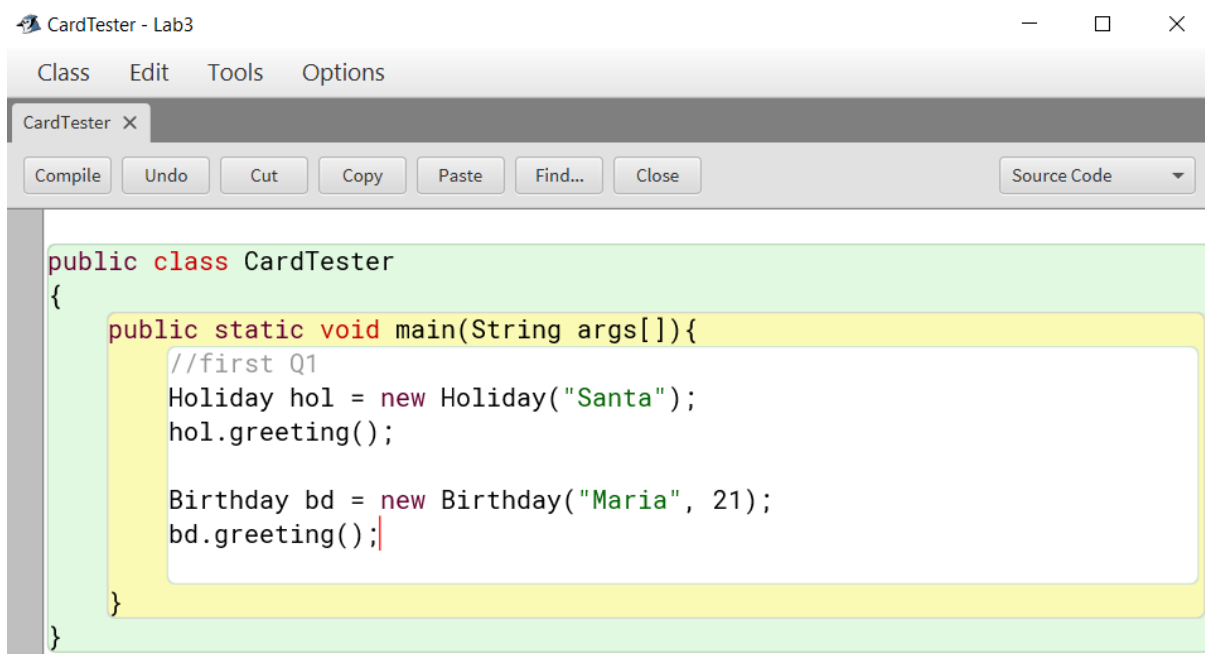
```
public class Birthday extends Card
{
    int age;

    public Birthday (String r, int years){
        recipient = r;
        age = years;
    }

    public void greeting(){
        System.out.println("Dear " + recipient + ",\n");
        System.out.println("Happy " + age + "th Birthday\n\n");
    }
}
```

Figure 5: Birthday class

1(c) and (d) :



```
public class CardTester
{
    public static void main(String args[]){
        //first Q1
        Holiday hol = new Holiday("Santa");
        hol.greeting();

        Birthday bd = new Birthday("Maria", 21);
        bd.greeting();
    }
}
```

Figure 6 :Answer for question 1(c)

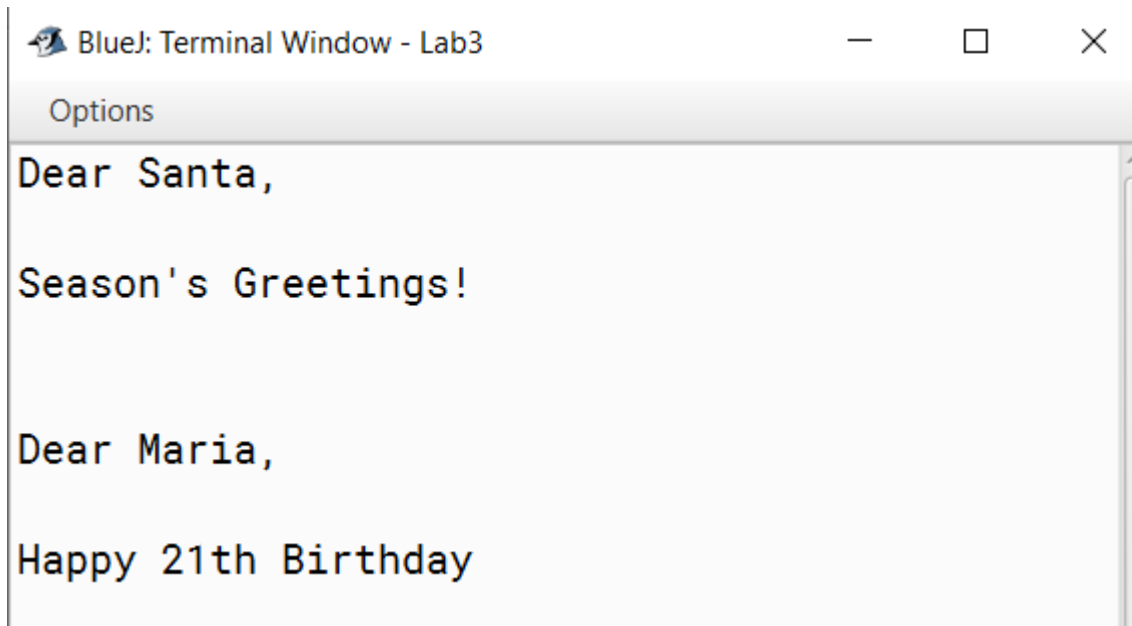


Figure 7: Answer question 1(d)

1(e):

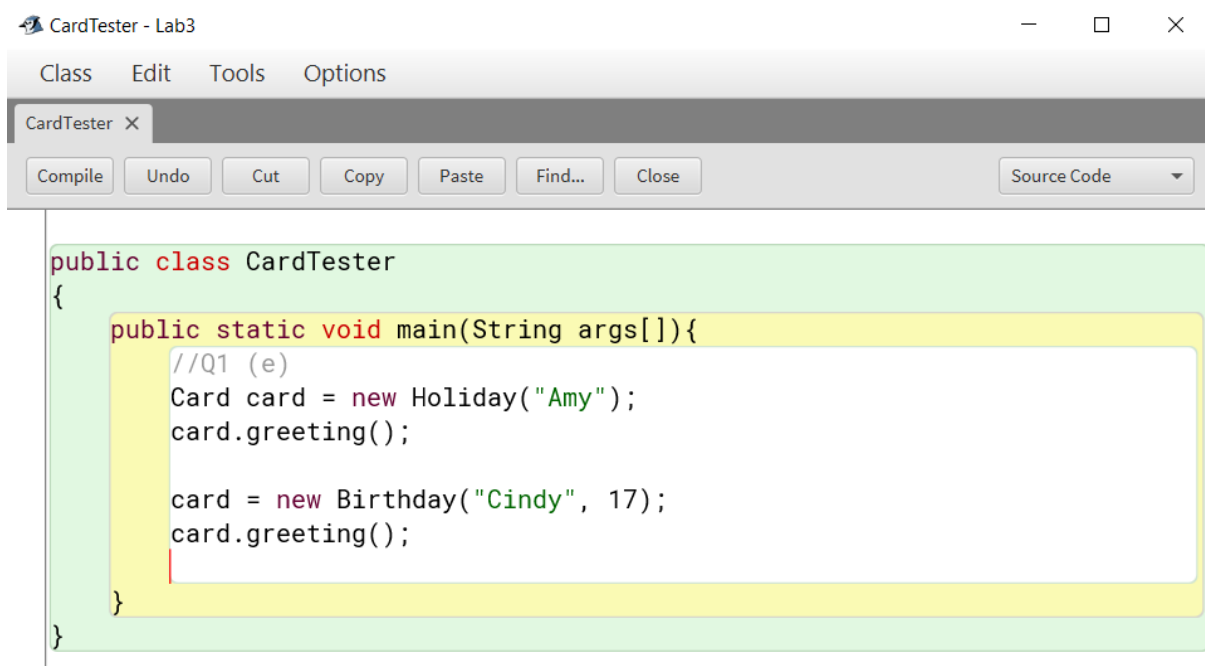


Figure 8: Rewrite the main program

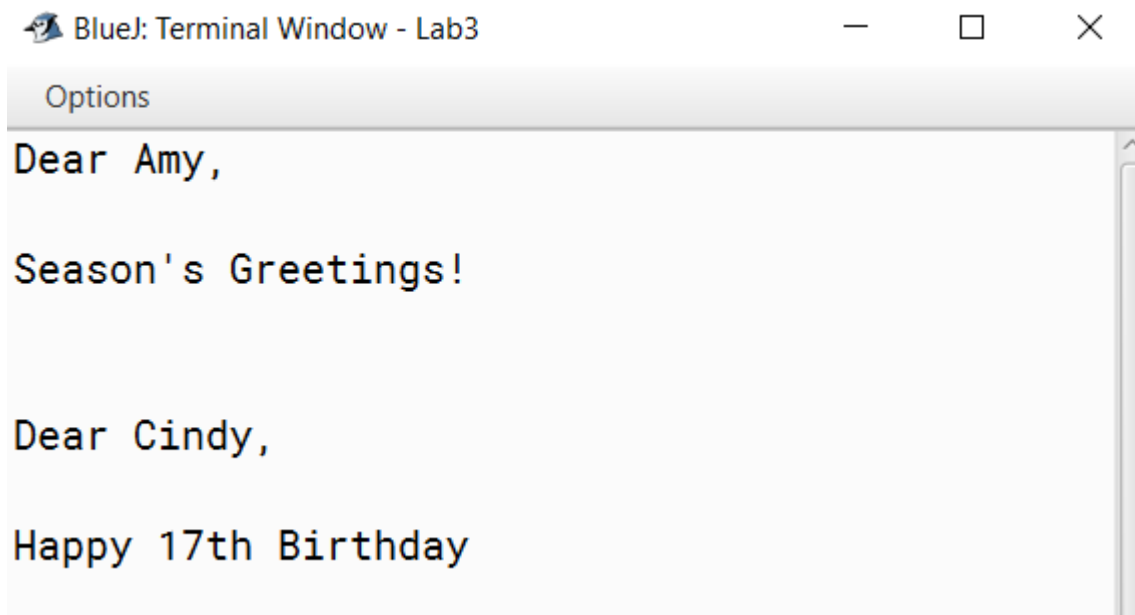


Figure 9: Output for question 1(e)

2. Give output

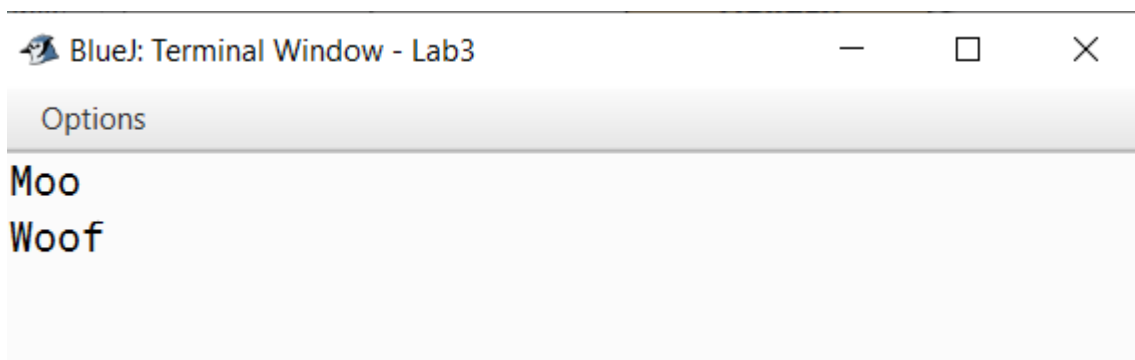
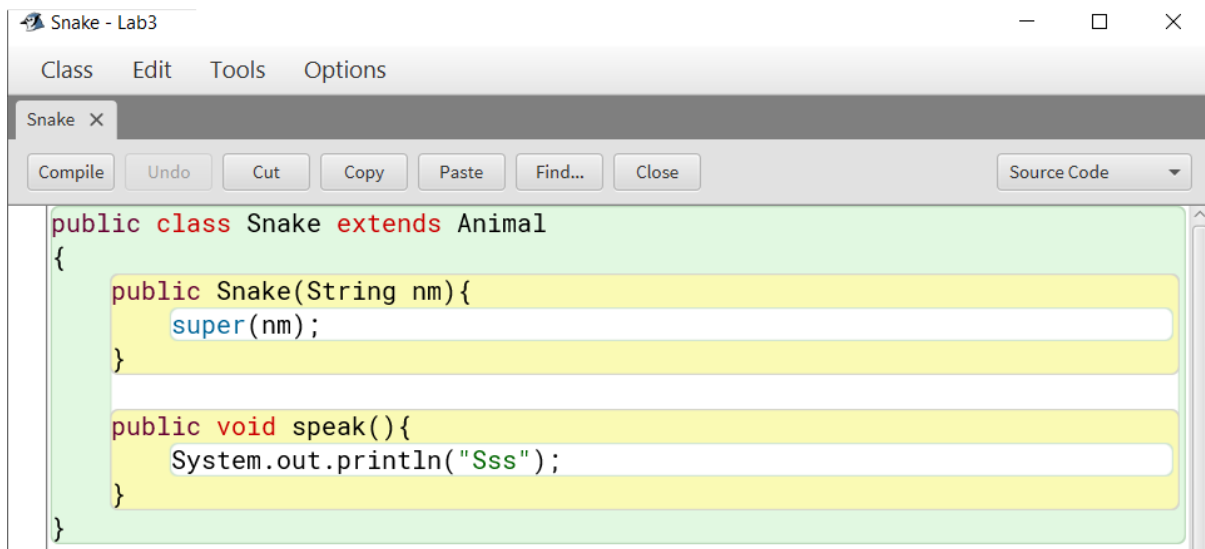


Figure 10: Output for question 2

3. Extends the program by adding 2 subclasses of Animal Snake and Cat

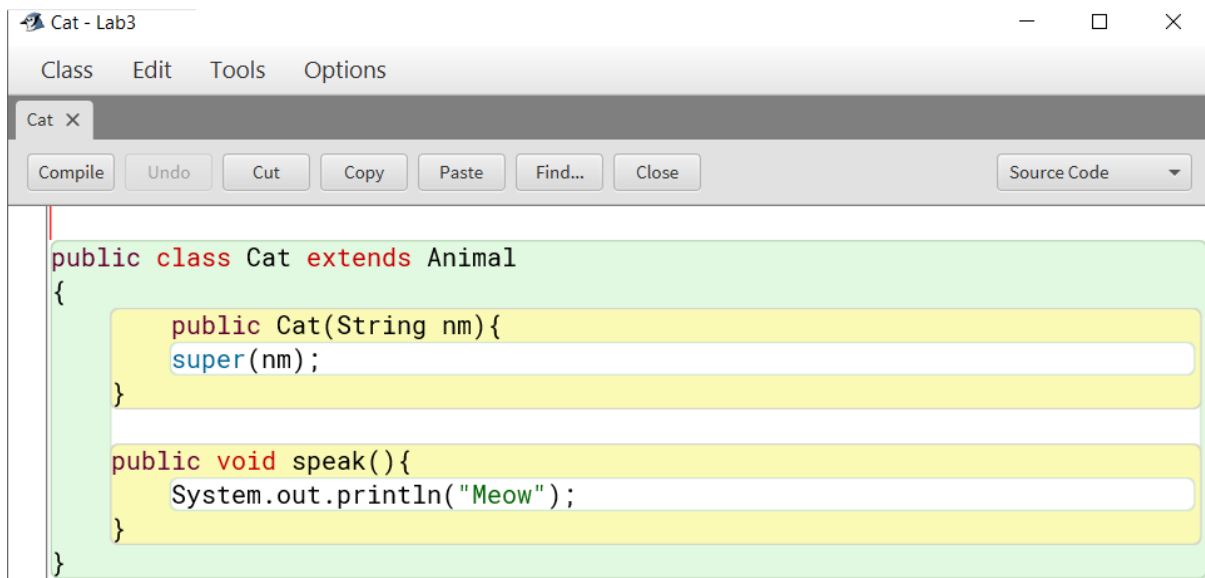


The screenshot shows an IDE window titled "Snake - Lab3". The menu bar includes "Class", "Edit", "Tools", and "Options". The toolbar contains buttons for "Compile", "Undo", "Cut", "Copy", "Paste", "Find...", and "Close", along with a "Source Code" dropdown. The code editor displays the following Java code for the Snake class:

```
public class Snake extends Animal
{
    public Snake(String nm){
        super(nm);
    }

    public void speak(){
        System.out.println("Sss");
    }
}
```

Figure 11: Snake class extends Animal



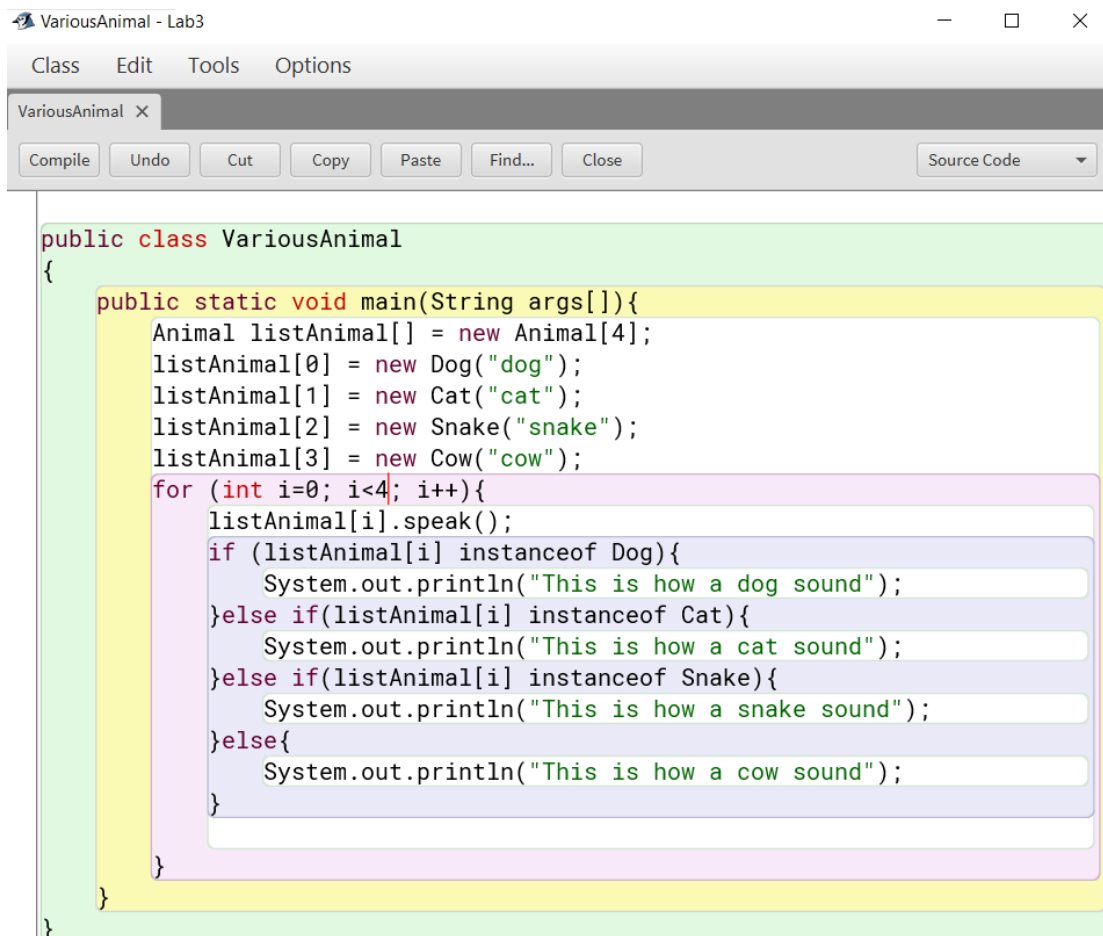
The screenshot shows an IDE window titled "Cat - Lab3". The menu bar includes "Class", "Edit", "Tools", and "Options". The toolbar contains buttons for "Compile", "Undo", "Cut", "Copy", "Paste", "Find...", and "Close", along with a "Source Code" dropdown. The code editor displays the following Java code for the Cat class:

```
public class Cat extends Animal
{
    public Cat(String nm){
        super(nm);
    }

    public void speak(){
        System.out.println("Meow");
    }
}
```

Figure 12: Cat class extends Animal

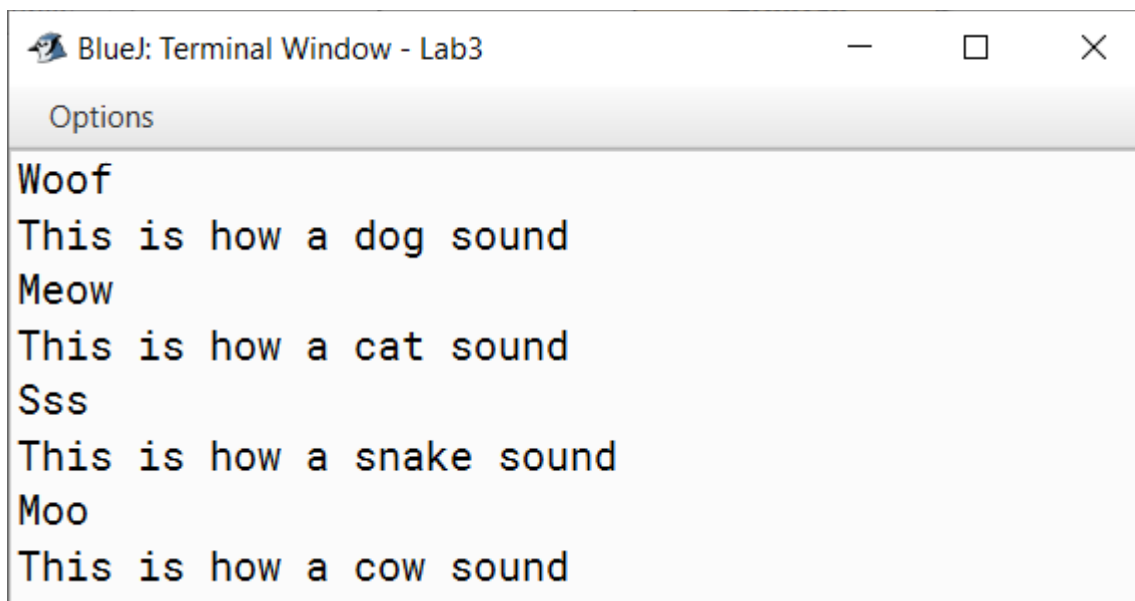
4.Modify program in exercise 3 by creating an array of superclass



The screenshot shows a Java IDE window titled 'VariousAnimal - Lab3'. The code defines a 'VariousAnimal' class with a 'main' method. In the 'main' method, an array of 'Animal' objects is created and populated with 'Dog', 'Cat', 'Snake', and 'Cow' instances. A loop iterates over this array, calling the 'speak()' method on each object. An 'if-else' statement checks the type of each object and prints a corresponding message: 'This is how a dog sound', 'This is how a cat sound', 'This is how a snake sound', or 'This is how a cow sound'.

```
public class VariousAnimal
{
    public static void main(String args[]){
        Animal listAnimal[] = new Animal[4];
        listAnimal[0] = new Dog("dog");
        listAnimal[1] = new Cat("cat");
        listAnimal[2] = new Snake("snake");
        listAnimal[3] = new Cow("cow");
        for (int i=0; i<4; i++){
            listAnimal[i].speak();
            if (listAnimal[i] instanceof Dog){
                System.out.println("This is how a dog sound");
            }else if(listAnimal[i] instanceof Cat){
                System.out.println("This is how a cat sound");
            }else if(listAnimal[i] instanceof Snake){
                System.out.println("This is how a snake sound");
            }else{
                System.out.println("This is how a cow sound");
            }
        }
    }
}
```

Figure 13: Code for question 4



The screenshot shows a terminal window titled 'BlueJ: Terminal Window - Lab3'. The output of the program is displayed as follows:

```
Woof
This is how a dog sound
Meow
This is how a cat sound
Sss
This is how a snake sound
Moo
This is how a cow sound
```

Figure 14: Output for question 4

Postlab Exercise

a) Code:

```
public abstract class ZooTaiping
{
    private String visitorName;
    private String icNumber;
    private boolean govServant;

    public ZooTaiping(){}

    public ZooTaiping(String name, String ic, boolean govServ){
        visitorName = name;
        icNumber = ic;
        govServant = govServ;
    }

    public void setVisitorName(String name){
        visitorName = name;
    }

    public void setIcNumber(String ic){
        icNumber = ic;
    }

    public void setGovServant(boolean govServ){
        govServant = govServ;
    }
}
```



```
public String getVisitorName(){  
    return visitorName;  
}
```

```
public String getIcNumber(){  
    return icNumber;  
}
```

```
public boolean getGovServ(){  
    return govServant;  
}
```

```
public abstract double totalCharges();
```

```
public String getTicketInfo() {  
    StringBuilder ticketDetails = new StringBuilder();  
  
    // Information for day visits  
    ticketDetails.append("Day Visit Pricing:\n");  
    ticketDetails.append("Adult: RM 25.00 per person\n");  
    ticketDetails.append("Children: RM 15.00 per person\n\n");  
  
    // Information for night visits  
    ticketDetails.append("Night Visit Pricing (Adults only):\n");  
    ticketDetails.append("Package A: RM 50.00 per person\n");  
    ticketDetails.append("Package B: RM 75.00 per person\n");  
    ticketDetails.append("Package C: RM 100.00 per person\n\n");  
  
    // Discount for government servants  
    ticketDetails.append("Special Discount:\n");
```

```
        ticketDetails.append("Government servants receive a 15 % discount on all  
tickets.\n");
```

```
        // Return the ticket information  
        return ticketDetails.toString();  
    }
```

```
    public String toString(){  
        return ( "\nVisitor Name : "+visitorName+  
        "\nIC number : "+icNumber+  
        "\nGoverment Servant Status : "+govServant);  
    }  
}
```

```
public class DayVisit extends ZooTaiping  
{  
    private String category;
```

```
    public DayVisit(){}
```

```
    public DayVisit(String name, String ic, boolean govServ, String category ){  
        super(name,ic,govServ);  
        this.category = category;  
    }
```

```
    public void setCategory(String category){  
        this.category = category;  
    }
```

```
public String getCategory(){  
    return category;  
}
```

```
public double totalCharges(){  
    double price;  
    if(category.equalsIgnoreCase("Adult")){  
        price = 25.00;  
        if(super.getGovServ() == true){  
            double discountPrice = price * 0.15;  
            price = price - discountPrice;  
            return price;  
        }  
        return price;  
    }else{  
        price = 15.00;  
        return price;  
    }  
}
```

```
public String toString(){  
    return (super.toString() + "\nCategory :" + category);  
}  
}
```

```
public class NightVisit extends ZooTaiping  
{  
    private char Package;
```

```
public NightVisit(){}
```

```
public NightVisit(String name, String ic, boolean govServ, char Package){  
    super(name,ic,govServ);  
    this.Package = Package;  
}
```

```
public char getPackage(){  
    return Package;  
}
```

```
public double totalCharges(){  
    double price;  
    if(super.getGovServ() == true){  
        double discount = 0.15, discountPrice = 0.00;  
        if(Package == 'A' || Package == 'a'){  
            discountPrice = 50 * discount;  
            price = 50 - discountPrice;  
        }else if(Package == 'B' || Package == 'b'){  
            discountPrice = 75 * discount;  
            price = 75 - discountPrice;  
        }else{  
            discountPrice = 100 * discount;  
            price = 100 - discountPrice;  
        }  
        return price;  
    }else{  
        if(Package == 'A' || Package == 'a'){  
            price = 50;  
        }  
    }  
}
```

```

    }else if(Package == 'B' || Package == 'b'){
        price = 75;
    }else{
        price = 100;
    }
    return price;
}
}

```

```

public String getPackageDetails(){
    StringBuilder packageDetails = new StringBuilder();

    packageDetails.append("Night Visit Pricing (Adults only):\n");
    packageDetails.append("Package A: RM 50.00 per person\n");
    packageDetails.append("Package B: RM 75.00 per person\n");
    packageDetails.append("Package C: RM 100.00 per person\n\n");

    // Discount for government servants
    packageDetails.append("Special Discount:\n");
    packageDetails.append("Government servants receive a 15 % discount on all tickets.\n");

    // Return the ticket information
    return packageDetails.toString();
}

public String toString(){
    return (super.toString()+ "\nPackage : " + Package);
}
}

```

b) Write TestZoo class

```
import java.util.*;
```

```
public class TestZoo{
```

```
    public static void main(String args[]){
```

```
        Scanner inputText = new Scanner(System.in);
```

```
        Scanner inputOption = new Scanner(System.in);
```

```
        Scanner inputNumber = new Scanner(System.in);
```

```
        char opt = ' ';
```

```
        double totalPrice = 0;
```

```
        ZooTaiping zt = new DayVisit();
```

```
        ArrayList<ZooTaiping> allList = new ArrayList<>();
```

```
        while(opt != 'E' || opt != 'e'){
```

```
            System.out.println("\nWelcome To Zoo Taiping ");
```

```
            System.out.println(zt.getTicketInfo()+"\n");
```

```
            System.out.println("Buy Ticket - (Y) or Exit - (E)\n");
```

```
            opt = inputOption.nextLine().toUpperCase().charAt(0);
```

```
            if(opt == 'Y'){
```

```
                char visitOption;
```

```
                System.out.println("\nHow many ticket do you want? ");
```

```
                int numOfTicket = inputNumber.nextInt();
```

```
                //declaration of array to store data from user – Q1
```

```
                ZooTaiping[] listVisit = new ZooTaiping[numOfTicket];
```

```
                System.out.println("You want Day visit? - (D) or Night visit? - (N)");
```

```
                visitOption = inputOption.nextLine().toUpperCase().charAt(0);
```

```
                switch(visitOption){
```

```
                    case 'D':
```

```

for(int i =0; i < listVisit.length; i++){
    System.out.println("Ticket : " + (i+1));
    System.out.println("Adult or Children \n");
    System.out.println("Please key in Adult for adult ticket or Children
for children ticket\n");
    String category = inputText.nextLine().toUpperCase();
    if(category.equalsIgnoreCase("Adult")){
        boolean govServ = false;
        System.out.println("Are you a government servant or not? (Y -
yes or N - no)");
        char gs = inputOption.nextLine().toUpperCase().charAt(0);
        if(gs == 'Y'){
            govServ = true;
        }
        System.out.println("Enter your name : ");
        String name = inputText.nextLine();
        System.out.println("Enter your IC number : ");
        String ic = inputText.nextLine();

        DayVisit dv = new DayVisit(name, ic, govServ, category);
        listVisit[i] = dv;
        allList.add(dv);
    }else{
        boolean govServ = false;
        System.out.println("Enter your name : ");
        String name = inputText.nextLine();
        System.out.println("Enter your IC number : ");
        String ic = inputText.nextLine();

        DayVisit dv = new DayVisit(name, ic, govServ, category);
        listVisit[i] = dv;
    }
}

```

```

        allList.add(dv);
    }
}

for(int i =0; i< listVisit.length;i++){
    totalPrice = totalPrice + listVisit[i].totalCharges();
}

System.out.println("Total price for the tickets is : RM "+ totalPrice +
"\n");

break;

case 'N':
    NightVisit nvd = new NightVisit();
    boolean govServ = false;
    for(int i = 0; i< listVisit.length; i++){
        String category = "Adult";
        nvd.getPackageDetails();
        System.out.println("Ticket : " + (i+1));
        System.out.println("What Package do you want? ");
        char pack = inputOption.nextLine().toUpperCase().charAt(0);
        System.out.println("Are you a government servant or not? (Y - yes
or N - no)");

        char gs = inputOption.nextLine().toUpperCase().charAt(0);
        if(gs == 'Y'){
            govServ = true;
        }else{
            govServ = false;
        }

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

```



```

        String name = inputText.nextLine();
        System.out.println("Enter your IC number : ");
        String ic = inputText.nextLine();

        NightVisit nv = new NightVisit(name, ic, govServ, pack);
        listVisit[i] = nv;
        allList.add(nv);
    }

    for(int i =0; i< listVisit.length;i++){
        totalPrice = totalPrice + listVisit[i].totalCharges();
    }

    System.out.println("Total price for the tickets is : RM "+ totalPrice +
"\n");

    break;
}

}

```

//Answers for Q2 & Q3

```

else if(opt == 'O'){
    int dayVisitor = 0, nightVisitor = 0;
    double totalIncome = 0.00;
    for(int i =0; i< allList.size();i++){
        if(allList.get(i) instanceof DayVisit){
            dayVisitor = dayVisitor + 1;
        }else if(allList.get(i) instanceof NightVisit){
            nightVisitor = nightVisitor +1;
        }
        totalIncome = totalIncome + allList.get(i).totalCharges();
    }
}

```

```
}
```

```
System.out.println("Total number of visitor for Day Visit is : "+ dayVisitor);
```

```
System.out.println("Total number of visitor for Night Visit is : "+ nightVisitor);
```

```
System.out.println("Total income for Zoo Taiping is : RM "+ totallIncome);
```

```
//Q3
```

```
for(int i = 0; i < allList.size(); i++){
```

```
    if(allList.get(i) instanceof NightVisit){
```

```
        NightVisit n = new NightVisit();
```

```
        n = (NightVisit)allList.get(i);
```

```
        if(n.getPackage() == 'C' || n.getPackage() == 'c'){
```

```
            allList.get(i).toString();
```

```
        }
```

```
    }
```

```
}
```

```
}else{
```

```
    opt = 'E';
```

```
    return ;
```

```
}
```

```
}
```

```
}
```

```
}
```