Q1.

Code(cat):

|  |
| --- |
| ***package Q\_01;  public class cat extends pet {  @Override  public String speak(){  return "Meow!";  } }*** |

Code(dog):

|  |
| --- |
| ***package Q\_01;  public class dog extends pet {  @Override  public String speak(){  return "Woof!!";  } }*** |

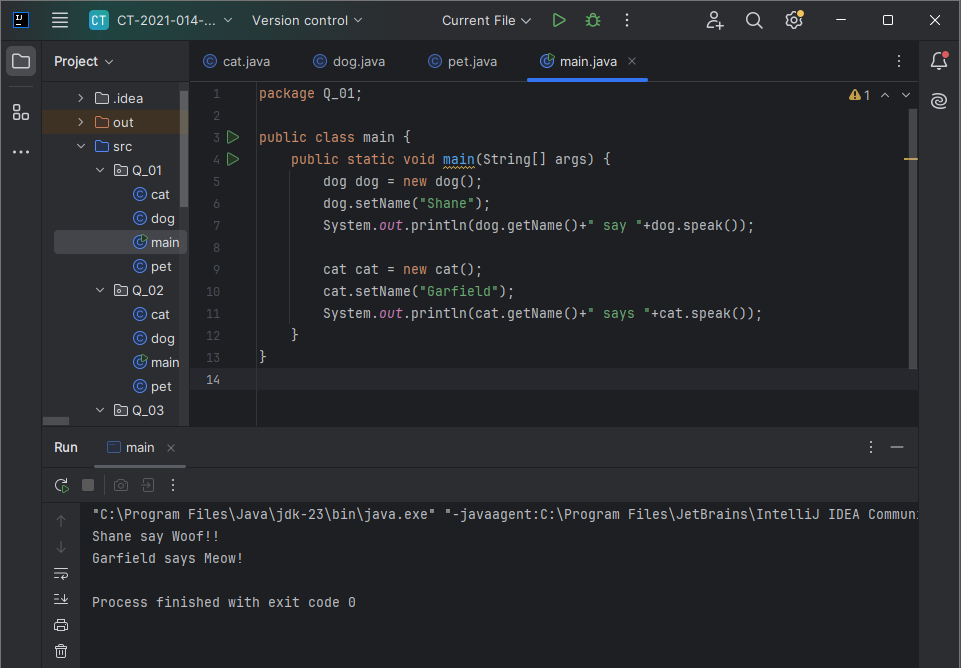
Code(pet):

|  |
| --- |
| ***package Q\_01; public class pet {  private String name;   public String getName( ) {  return name;  }  public void setName(String petName) {  name = petName;  }  public String speak( ) {  return "I'm your cuddly little pet.";  } }*** |

Code(main):

|  |
| --- |
| ***package Q\_01;  public class main {  public static void main(String[] args) {  dog dog = new dog();  dog.setName("Shane");  System.out.println(dog.getName()+" say "+dog.speak());   cat cat = new cat();  cat.setName("Garfield");  System.out.println(cat.getName()+" says "+cat.speak());  } }*** |

Output:



Q2.

Code(cat):

|  |
| --- |
| ***package Q\_02;  public class cat extends pet{ }*** |

Code(dog):

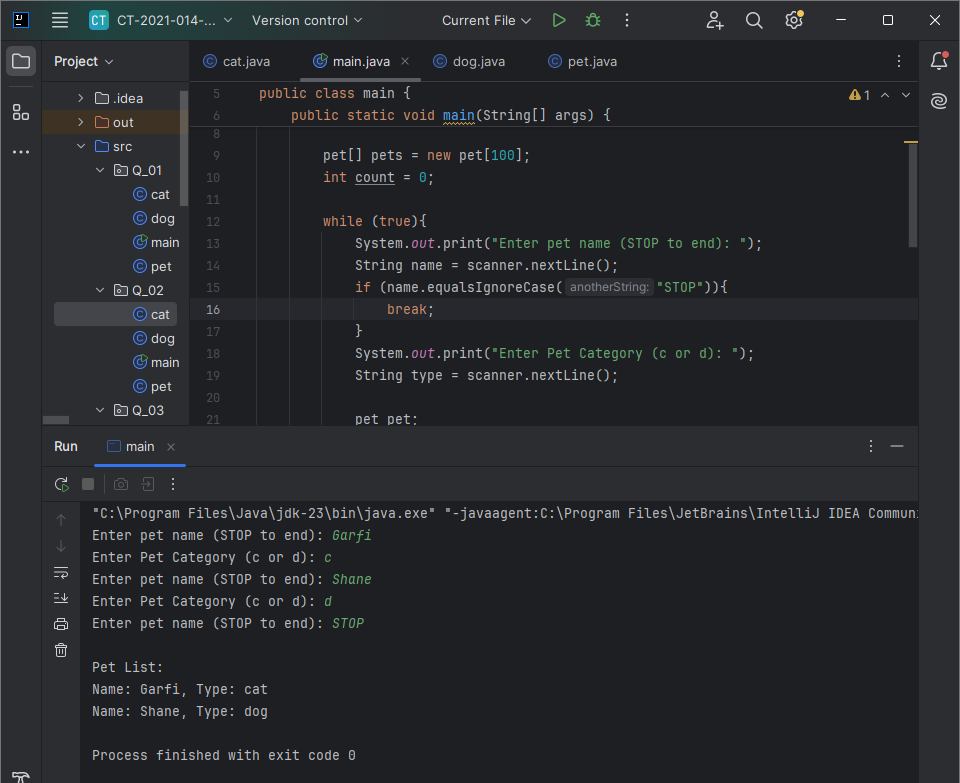
|  |
| --- |
| ***package Q\_02;  public class dog extends pet{ }*** |

Code(pet):

|  |
| --- |
| ***package Q\_02;  public class pet {  private String Name;   public void setName(String name) {  Name = name;  }   public String getName() {  return Name;  } }*** |

Code(main):

|  |
| --- |
| ***package Q\_02;  import java.util.Scanner;  public class main {  public static void main(String[] args) {  Scanner scanner=new Scanner(System.in);   pet[] pets = new pet[100];  int count = 0;   while (true){  System.out.print("Enter pet name (STOP to end): ");  String name = scanner.nextLine();  if (name.equalsIgnoreCase("STOP")){  break;  }  System.out.print("Enter Pet Category (c or d): ");  String type = scanner.nextLine();   pet pet;  if (type.equals("c")){  pet = new cat();  } else if (type.equals("d")) {  pet = new dog();  }else {  System.out.println("Invalid type!! Use 'c' or 'd'");  continue;  }  pet.setName(name);  pets[count] = pet;  count++;  }  System.out.println("\nPet List: ");  for (int i = 0; i< count; i++){  pet pet = pets[i];  String type = (pet instanceof cat) ? "cat" : "dog";  System.out.println("Name: "+pet.getName()+", Type: "+type);  }  } }*** |

Output: 

Q3.

Code(cat):

|  |
| --- |
| ***package Q\_03;  public class cat extends pet{ }*** |

Code(dog):

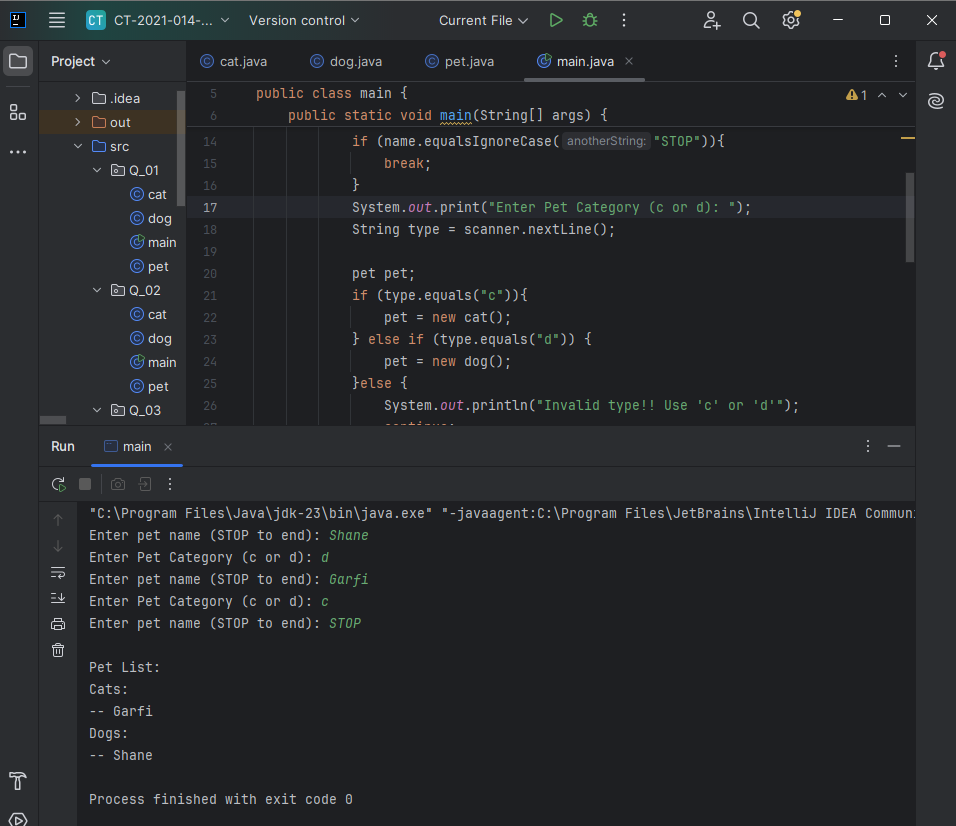
|  |
| --- |
| ***package Q\_03;  public class dog extends pet{ }*** |

Code(pet):

|  |
| --- |
| ***package Q\_03;  public class pet {  private String Name;   public String getName() {  return Name;  }   public void setName(String name) {  Name = name;  } }*** |

Code(main):

|  |
| --- |
| ***package Q\_03;  import java.util.Scanner;  public class main {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  pet[] pets = new pet[100];  int count = 0;   while (true){  System.out.print("Enter pet name (STOP to end): ");  String name = scanner.nextLine();  if (name.equalsIgnoreCase("STOP")){  break;  }  System.out.print("Enter Pet Category (c or d): ");  String type = scanner.nextLine();   pet pet;  if (type.equals("c")){  pet = new cat();  } else if (type.equals("d")) {  pet = new dog();  }else {  System.out.println("Invalid type!! Use 'c' or 'd'");  continue;  }  pet.setName(name);  pets[count] = pet;  count++;  }  System.out.println("\nPet List: ");   System.out.println("Cats: ");  for (int i =0; i< count;i++){  if (pets[i] instanceof cat){  System.out.println("-- "+ pets[i].getName());  }  }  System.out.println("Dogs: ");  for (int i = 0;i<count; i++){  if (pets[i] instanceof dog){  System.out.println("-- "+pets[i].getName());  }  }  } }*** |

Output: 

Q4.

Code(cat):

|  |
| --- |
| ***package Q\_04;  public class cat extends pet{  private String coatColor;   public String getCoatColor() {  return coatColor;  }   public void setCoatColor(String coatColor) {  this.coatColor = coatColor;  } }*** |

Code(dog):

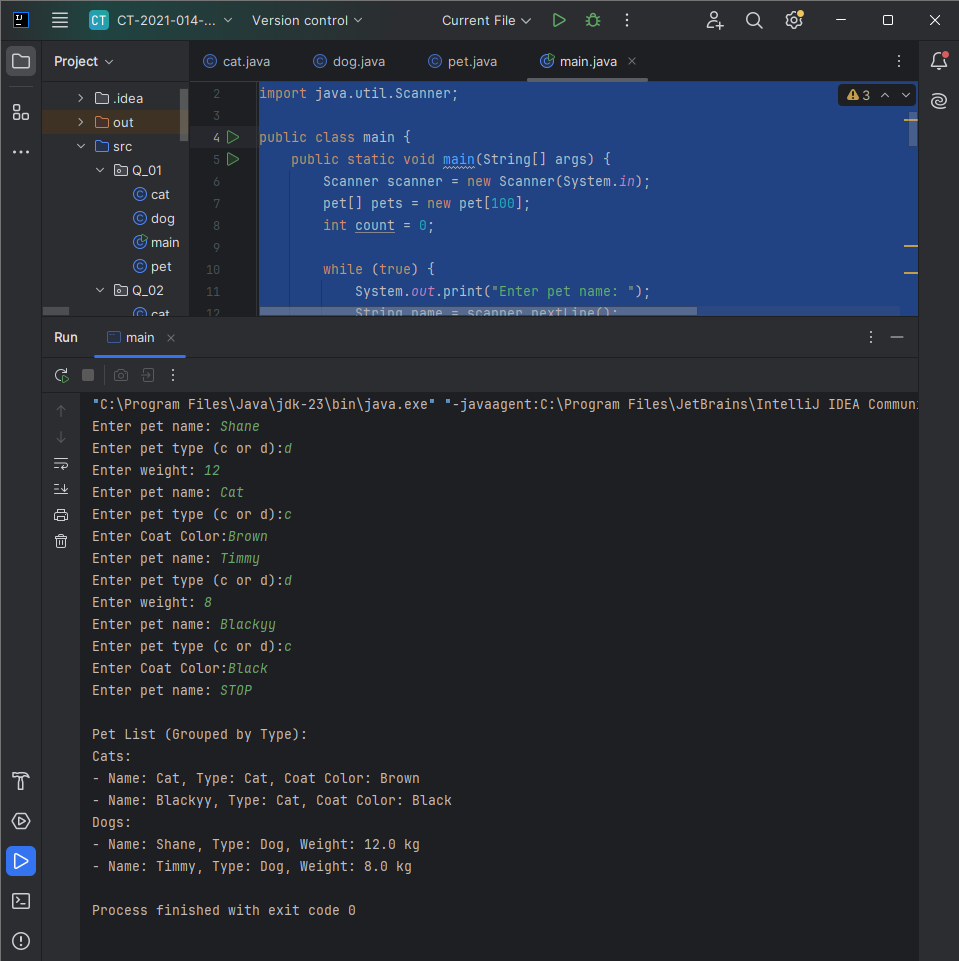
|  |
| --- |
| ***package Q\_04;  public class dog extends pet{  private double weight;   public double getWeight() {  return weight;  }   public void setWeight(double weight) {  this.weight = weight;  } }*** |

Code(pet):

|  |
| --- |
| ***package Q\_04;  public class pet {   private String Name;   public String getName() {  return Name;  }   public void setName(String name) {  Name = name;  } }*** |

Code(main):

|  |
| --- |
| ***package Q\_04; import java.util.Scanner;  public class main {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  pet[] pets = new pet[100];  int count = 0;   while (true) {  System.out.print("Enter pet name: ");  String name = scanner.nextLine();   if (name.equalsIgnoreCase("STOP")) {  break;  }  System.out.print("Enter pet type (c or d):");  String type = scanner.nextLine().toLowerCase();   pet pet;  if (type.equals("c")) {  cat cat = new cat();  System.out.print("Enter Coat Color:");  String color = scanner.nextLine();  cat.setCoatColor(color);  pet = cat;  } else if (type.equals("d")) {  dog dog = new dog();  System.out.print("Enter weight: ");  double weight = Double.parseDouble(scanner.nextLine());  dog.setWeight(weight);  pet = dog;  } else {  System.out.println("Invalid type!! Enter c or d:");  continue;  }  pet.setName(name);  pets[count] = pet;  count++;  }  System.out.println("\nPet List (Grouped by Type):");   // Cats first  System.out.println("Cats:");  for (int i = 0; i < count; i++) {  if (pets[i] instanceof cat) {  cat cat = (cat) pets[i];  System.out.println("- Name: " + cat.getName() + ", Type: Cat, Coat Color: " + cat.getCoatColor());  }  }   // Dogs next  System.out.println("Dogs:");  for (int i = 0; i < count; i++) {  if (pets[i] instanceof dog) {  dog dog = (dog) pets[i];  System.out.println("- Name: " + dog.getName() + ", Type: Dog, Weight: " + dog.getWeight() + " kg");  }    }  } }*** |

Output: 

Q5.

Code(cat):

|  |
| --- |
| ***package Q\_05;  public class cat extends pet {  private String coatColor;   public String getCoatColor() {  return coatColor;  }   public void setCoatColor(String coatColor) {  this.coatColor = coatColor;  } }*** |

Code(dog):

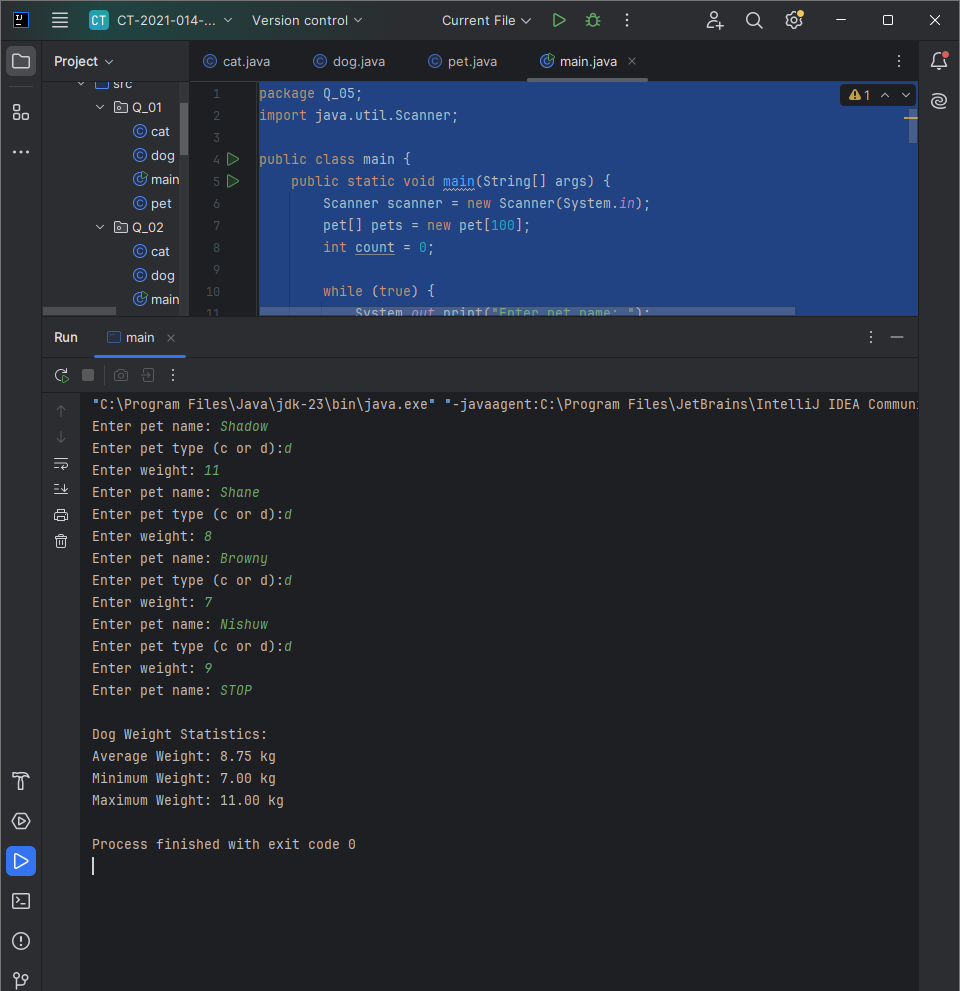
|  |
| --- |
| ***package Q\_05;  public class dog extends pet {  private double weight;   public double getWeight() {  return weight;  }   public void setWeight(double weight) {  this.weight = weight;  } }*** |

Code(pet):

|  |
| --- |
| ***package Q\_05;  public class pet {  private String Name;   public String getName() {  return Name;  }   public void setName(String name) {  Name = name;  } }*** |

Code(main):

|  |
| --- |
| ***package Q\_05; import java.util.Scanner;  public class main {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  pet[] pets = new pet[100];  int count = 0;   while (true) {  System.out.print("Enter pet name: ");  String name = scanner.nextLine();   if (name.equalsIgnoreCase("STOP")) {  break;  }  System.out.print("Enter pet type (c or d):");  String type = scanner.nextLine().toLowerCase();   pet pet;  if (type.equals("c")) {  cat cat = new cat();  System.out.print("Enter Coat Color:");  String color = scanner.nextLine();  cat.setCoatColor(color);  pet = cat;  } else if (type.equals("d")) {  dog dog = new dog();  System.out.print("Enter weight: ");  double weight = Double.parseDouble(scanner.nextLine());  dog.setWeight(weight);  pet = dog;  } else {  System.out.println("Invalid type!! Enter c or d:");  continue;  }  pet.setName(name);  pets[count] = pet;  count++;  }  dog[] dogArray = new dog[count];  int dogCount = 0;   for (int i = 0; i < count; i++) {  if (pets[i] instanceof dog) {  dogArray[dogCount] = (dog) pets[i];  dogCount++;  }  }   dog[] actualDogs = new dog[dogCount];  System.arraycopy(dogArray, 0, actualDogs, 0, dogCount);   if (dogCount > 0) {  double totalWeight = 0;  double minWeight = actualDogs[0].getWeight();  double maxWeight = actualDogs[0].getWeight();   for (int i = 0; i < dogCount; i++) {  double w = actualDogs[i].getWeight();  totalWeight += w;   if (w < minWeight) {  minWeight = w;  }  if (w > maxWeight) {  maxWeight = w;  }  }  double averageWeight = totalWeight / dogCount;   System.out.println("\nDog Weight Statistics:");  System.out.printf("Average Weight: %.2f kg\n", averageWeight);  System.out.printf("Minimum Weight: %.2f kg\n", minWeight);  System.out.printf("Maximum Weight: %.2f kg\n", maxWeight);  } else {  System.out.println("\nNo dogs were entered, so no weight statistics available.");  }   } }*** |

Output: 

Q6.

Code(cat):

|  |
| --- |
| ***package Q\_06;  public class cat extends pet{  private String coatColor;   public String getCoatColor() {  return coatColor;  }   public void setCoatColor(String coatColor) {  this.coatColor = coatColor;  }  }*** |

Code(dog):

|  |
| --- |
| ***package Q\_06;  public class dog extends pet{  private double weight;   public double getWeight() {  return weight;  }   public void setWeight(double weight) {  this.weight = weight;  }  }*** |

Code(pet):

|  |
| --- |
| ***package Q\_06;  public class pet {  private String name;   public String getName() {  return name;  }   public void setName(String petName) {  name = petName;  } }*** |

Code(main):

|  |
| --- |
| ***package Q\_06;  import java.util.ArrayList; import java.util.Iterator; import java.util.Scanner;  public class main {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  ArrayList<pet> petList = new ArrayList<>();  ArrayList<cat> catList = new ArrayList<>();  ArrayList<dog> dogList = new ArrayList<>();   while (true) {  System.out.println("\n--- Pet Menu ---");  System.out.println("1. Add Cat");  System.out.println("2. Add Dog");  System.out.println("3. Remove Cat");  System.out.println("4. Remove Dog");  System.out.println("0. Quit");  System.out.print("Enter your choice: ");  String choice = scanner.nextLine();   switch (choice) {  case "1":  System.out.print("Enter cat name: ");  String catName = scanner.nextLine();  if (petExists(petList, catName)) {  System.out.println("A pet with that name already exists.");  break;  }  System.out.print("Enter coat color: ");  String coatColor = scanner.nextLine();   cat cat = new cat();  cat.setName(catName);  cat.setCoatColor(coatColor);   petList.add(cat);  catList.add(cat);   System.out.println("Cat added.");  break;  case "2":  System.out.print("Enter dog name: ");  String dogName = scanner.nextLine();  if (petExists(petList, dogName)) {  System.out.println("A pet with that name already exists.");  break;  }  System.out.print("Enter weight (kg): ");  double weight = Double.parseDouble(scanner.nextLine());  dog dog = new dog();  dog.setName(dogName);  dog.setWeight(weight);   petList.add(dog);  dogList.add(dog);   System.out.println("Dog added.");  break;  case "3":  System.out.print("Enter cat name to remove: ");  String removeCatName = scanner.nextLine();  removePetByName(removeCatName, petList, catList);  break;  case "4":  System.out.print("Enter dog name to remove: ");  String removeDogName = scanner.nextLine();  removePetByName(removeDogName, petList, dogList);  break;  case "0":  System.out.println("Goodbye!");  scanner.close();  return;  default:  System.out.println("Invalid choice. Try again.");  }  }  }   private static boolean petExists(ArrayList<pet> petList, String catName) {  for (pet p : petList) {  if (p.getName().equalsIgnoreCase(catName)) {  return true;  }  }  return false;  }   private static void removePetByName(String name, ArrayList<pet> petList, ArrayList<? extends pet> typeList) {  boolean found = false;   // Remove from main list  Iterator<pet> petIter = petList.iterator();  while (petIter.hasNext()) {  if (petIter.next().getName().equalsIgnoreCase(name)) {  petIter.remove();  found = true;  break;  }  }  } }*** |

Output: 