Q1 Code: Pet.java

package Q1;  
  
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.";  
 }  
}

Cat.java

package Q1;  
  
public class Cat extends Pet {  
 @Override  
 public String speak() {  
 return "";  
 }  
}

Dog.java

package Q1;  
  
public class Dog extends Pet{  
 @Override  
 public String speak() {  
 return "";  
 }  
}

Main.java

package Q1;  
  
public class Main {  
 public static void main(String[] args) {  
 Pet p1=new Pet();  
 p1.setName("Tommy");  
  
 Pet p2=new Cat();  
 p2.setName("Milo");  
  
 Pet p3=new Dog();  
 p3.setName("Baster");  
  
 Cat p4=new Cat();  
 p4.setName("Charlie");  
  
 Dog p5=new Dog();  
 p5.setName("Blacky");  
  
 System.*out*.println("Pet 1 name "+p1.getName()+", says: "+p1.speak());  
 System.*out*.println("Pet 2 name "+p2.getName()+", says: "+p2.speak());  
 System.*out*.println("Pet 3 name "+p3.getName()+ ", says: "+p3.speak());  
 System.*out*.println("Pet 4 name "+p4.getName()+ ", says: "+p4.speak());  
 System.*out*.println("Pet 5 name "+p5.getName() +", says: "+p5.speak());  
 }  
  
}

Output of Q1:

A screenshot of a computer program

AI-generated content may be incorrect.

Q1 Method 2 Returning String

Pet.java

package Q1\_2;  
  
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.";  
 }  
}

Cat.java

package Q1\_2;  
  
public class Cat extends Pet {  
 @Override  
 public String speak() {  
 return "Meow!";  
 }  
}

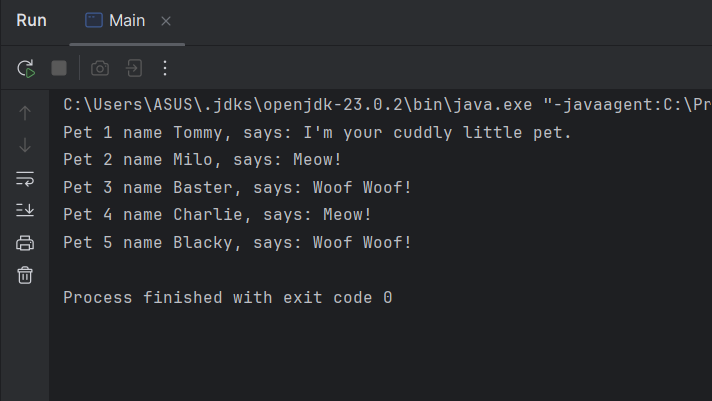
Dog.java

package Q1\_2;  
  
public class Dog extends Pet{  
 @Override  
 public String speak() {  
 return "Woof Woof!";  
 }  
}

Main.java

package Q1\_2;  
  
public class Main {  
 public static void main(String[] args) {  
 Pet p1=new Pet();  
 p1.setName("Tommy");  
  
 Pet p2=new Cat();  
 p2.setName("Milo");  
  
 Pet p3=new Dog();  
 p3.setName("Baster");  
  
 Cat p4= new Cat();  
 p4.setName("Charlie");  
  
 Dog p5=new Dog();  
 p5.setName("Blacky");  
  
 System.*out*.println("Pet 1 name "+p1.getName()+", says: "+p1.speak());  
 System.*out*.println("Pet 2 name "+p2.getName()+", says: "+p2.speak());  
 System.*out*.println("Pet 3 name "+p3.getName()+ ", says: "+p3.speak());  
 System.*out*.println("Pet 4 name "+p4.getName()+ ", says: "+p4.speak());  
 System.*out*.println("Pet 5 name "+p5.getName() +", says: "+p5.speak());  
 }  
}

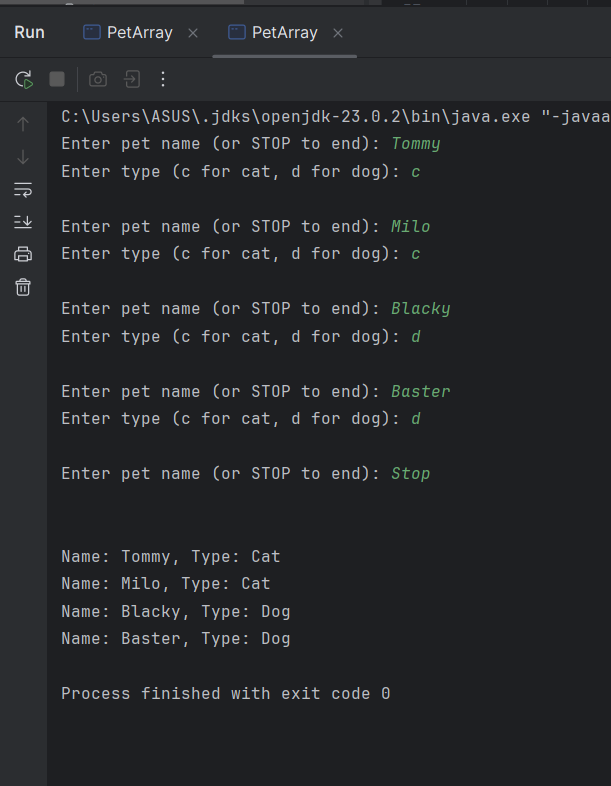
Q1 Method 2 Output



Q2 Code Method 1:

package Q2;  
import Q1.Cat;  
import Q1.Dog;  
import Q1.Pet;  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class PetArray {  
 public static void main(String[] args) {  
 ArrayList<Pet> pets = *collectPets*();  
 *displayPets*(pets);  
  
 }  
 private static ArrayList<Pet> collectPets(){  
 Scanner input = new Scanner(System.*in*);  
 ArrayList<Pet> petList = new ArrayList<>();  
  
 while (true) {  
 System.*out*.print("Enter pet name (or STOP to end): ");  
 String name = input.nextLine();  
 if (name.equalsIgnoreCase("STOP")) break;  
  
 System.*out*.print("Enter type (c for cat, d for dog): ");  
 String type = input.nextLine();  
  
 Pet pet = type.equalsIgnoreCase("c") ? new Cat() : new Dog();  
 pet.setName(name);  
 petList.add(pet);  
 System.*out*.println();  
 }  
 System.*out*.println("\n");  
 return petList;  
 }  
 public static void displayPets(ArrayList<Pet> pets) {  
 for (Pet p : pets) {  
 System.*out*.println("Name: " + p.getName() + ", Type: " + p.getClass().getSimpleName());  
 } } }

Output:



Q2 Method 2:

package Q2\_2;  
  
import Q1.Cat;  
import Q1.Dog;  
import Q1.Pet;  
  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class PetArray2 {  
 public static void main(String[] args) {  
 Scanner input= new Scanner(System.*in*);  
 ArrayList<Pet> pets= new ArrayList<>();  
  
 //Input Loop  
 while(true) {  
 System.*out*.println("Enter pet name or STOP to end: ");  
 String name = input.nextLine();  
 if(name.equalsIgnoreCase("STOP")){  
 break;  
 }  
 System.*out*.println("Enter type ( c for cat & d for dog) :");  
 String type= input.nextLine();  
  
 Pet pet;  
 if(type.equalsIgnoreCase("c")){  
 pet=new Cat();  
 }else if(type.equalsIgnoreCase("d")){  
 pet=new Dog();  
 }else{  
 System.*out*.println("Invalid Entry!");  
 continue;  
 }  
 pet.setName(name);  
 pets.add(pet);  
 }

//Output loop  
 System.*out*.println("\n--- Pet List ---");  
 for (Pet p : pets) {  
 String type = (p instanceof Cat) ? "Cat" : "Dog";  
 System.*out*.println("Name: " + p.getName() + ", Type: " + type);  
 }  
 }  
}

Output:

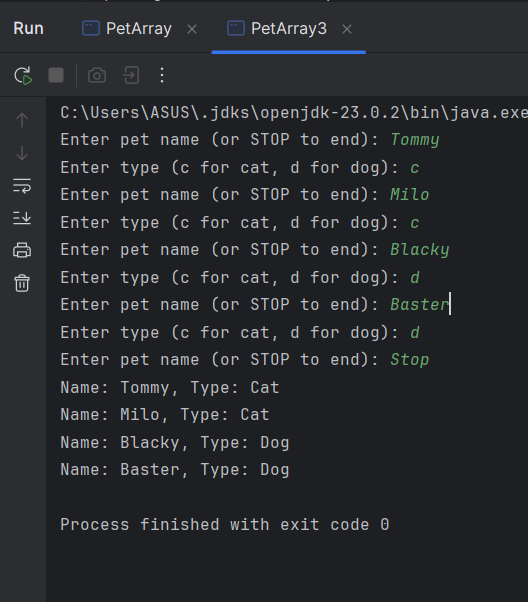
A screenshot of a computer program

AI-generated content may be incorrect.

Q2 Code Method 3:

package Q2\_3;  
import Q1.Cat;  
import Q1.Dog;  
import Q1.Pet;  
import java.util.Scanner;  
  
public class PetArray3 {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.*in*);  
 Pet[] pets = new Pet[100]; // maximum 100 pets  
 int count = 0;  
  
 while (true) {  
 System.*out*.print("Enter pet name (or STOP to end): ");  
 String name = input.nextLine();  
 if (name.equalsIgnoreCase("STOP")) break;  
  
 System.*out*.print("Enter type (c for cat, d for dog): ");  
 String type = input.nextLine();  
  
 Pet pet;  
 if (type.equalsIgnoreCase("c")) {  
 pet = new Cat();  
 } else {  
 pet = new Dog();  
 }  
  
 pet.setName(name);  
 pets[count++] = pet;  
 }  
  
 for (int i = 0; i < count; i++) {  
 System.*out*.println("Name: " + pets[i].getName() + ", Type: " + pets[i].getClass().getSimpleName());  
 }  
 }  
}

Output:

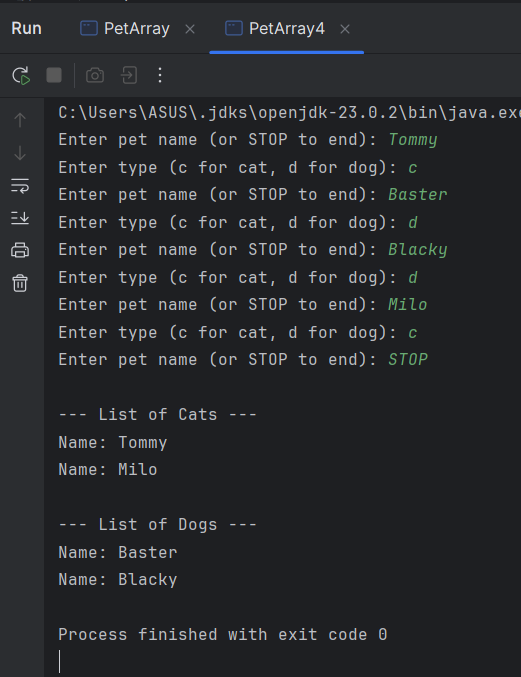


Q3

package Q3;  
import Q1.Cat;  
import Q1.Dog;  
import Q1.Pet;  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class PetArray4 {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.*in*);  
 ArrayList<Pet> pets = new ArrayList<>();  
  
 while (true) {  
 System.*out*.print("Enter pet name (or STOP to end): ");  
 String name = input.nextLine();  
 if (name.equalsIgnoreCase("STOP")) break;  
  
 System.*out*.print("Enter type (c for cat, d for dog): ");  
 String type = input.nextLine();  
  
 Pet pet = type.equalsIgnoreCase("c") ? new Cat() : new Dog();  
 pet.setName(name);  
 pets.add(pet);  
 }

System.*out*.println("\n--- List of Cats ---");  
 for (Pet p : pets) {  
 if (p instanceof Cat) {  
 System.*out*.println("Name: " + p.getName());  
 }  
 }  
 System.*out*.println("\n--- List of Dogs ---");  
 for (Pet p : pets) {  
 if (p instanceof Dog) {  
 System.*out*.println("Name: " + p.getName());  
 }  
 }  
 }}

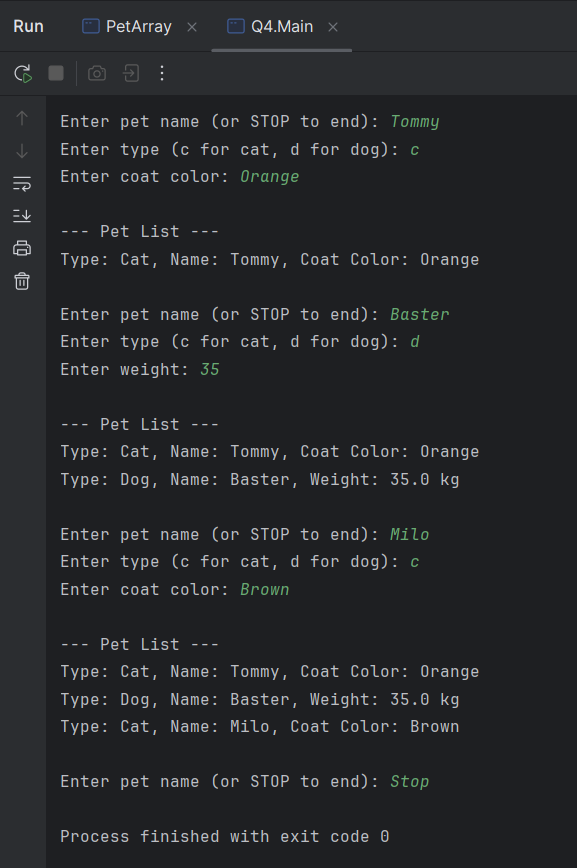
Output:



Q4 Code Method 1:

package Q4;  
  
import Q1.Pet;  
  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner input=new Scanner(System.*in*);  
 ArrayList<Pet> pets= new ArrayList<>();  
  
 while(true){  
 System.*out*.print("\nEnter pet name (or STOP to end): ");  
 String name = input.nextLine();  
 if (name.equalsIgnoreCase("STOP")) break;  
  
 System.*out*.print("Enter type (c for cat, d for dog): ");  
 String type = input.nextLine();  
  
 if (type.equalsIgnoreCase("c")) {  
 Cat2 cat = new Cat2();  
 cat.setName(name);  
 System.*out*.print("Enter coat color: ");  
 cat.setCoatColor(input.nextLine());  
 pets.add(cat);  
 } else if (type.equalsIgnoreCase("d")) {  
 Dog2 dog = new Dog2();  
 dog.setName(name);  
 System.*out*.print("Enter weight: ");

try {  
 dog.setWeight(Double.*parseDouble*(input.nextLine()));  
 } catch (NumberFormatException e) {  
 System.*out*.println("Invalid weight, setting to 0.");  
 dog.setWeight(0);  
 }  
 pets.add(dog);  
 } else {  
 System.*out*.println("Invalid type! Skipping entry.");  
 }  
  
 // Display results  
 System.*out*.println("\n--- Pet List ---");  
 for (Pet p : pets) {  
 if (p instanceof Cat2 cat) {  
 System.*out*.println("Type: Cat, Name: " + cat.getName() + ", Coat Color: " + cat.getCoatColor());  
 } else if (p instanceof Dog2 dog) {  
 System.*out*.println("Type: Dog, Name: " + dog.getName() + ", Weight: " + dog.getWeight() + " kg");  
 }  
 }  
 }  
 }  
}

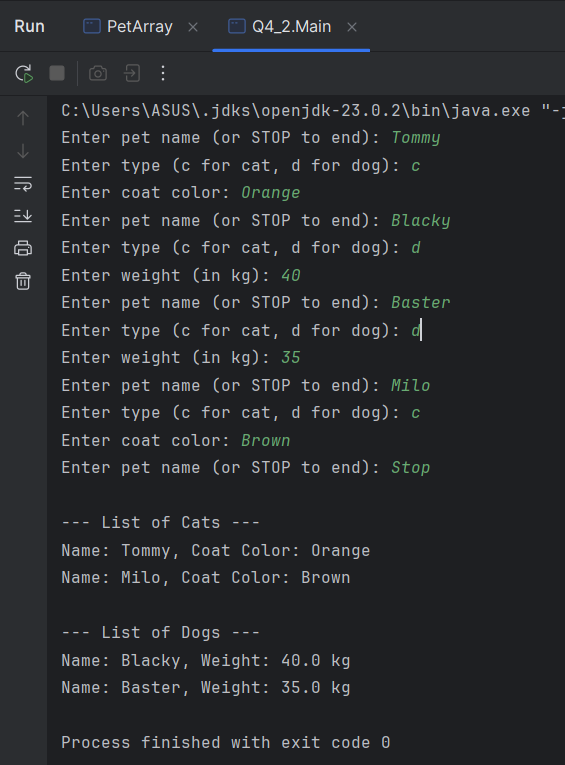


Q4 Method 2;

package Q4\_2;  
  
import Q4.Cat2;  
import Q4.Dog2;  
  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
  
 Scanner input = new Scanner(System.*in*);  
 ArrayList<Cat2> cats = new ArrayList<>();  
 ArrayList<Dog2> dogs = new ArrayList<>();  
  
 // Input Loop  
 while (true) {  
 System.*out*.print("Enter pet name (or STOP to end): ");  
 String name = input.nextLine();  
 if (name.equalsIgnoreCase("STOP")) break;  
  
 System.*out*.print("Enter type (c for cat, d for dog): ");  
 String type = input.nextLine();  
  
 if (type.equalsIgnoreCase("c")) {  
 Cat2 cat = new Cat2();  
 cat.setName(name);  
 System.*out*.print("Enter coat color: ");  
 cat.setCoatColor(input.nextLine());  
 cats.add(cat);  
 }

else if (type.equalsIgnoreCase("d")) {  
 Dog2 dog = new Dog2();  
 dog.setName(name);  
 System.*out*.print("Enter weight (in kg): ");  
 try {  
 dog.setWeight(Double.*parseDouble*(input.nextLine()));  
 } catch (NumberFormatException e) {  
 System.*out*.println("Invalid weight! Setting to 0.");  
 dog.setWeight(0);  
 }  
 dogs.add(dog);  
 } else {  
 System.*out*.println("Invalid type! Please enter 'c' or 'd'.");  
 }  
 }  
  
 // Output Cats  
 System.*out*.println("\n--- List of Cats ---");  
 for (Cat2 c : cats) {  
 System.*out*.println("Name: " + c.getName() + ", Coat Color: " + c.getCoatColor());  
 }  
  
 // Output Dogs  
 System.*out*.println("\n--- List of Dogs ---");  
 for (Dog2 d : dogs) {  
 System.*out*.println("Name: " + d.getName() + ", Weight: " + d.getWeight() + " kg");  
 }  
 }  
}

Output:

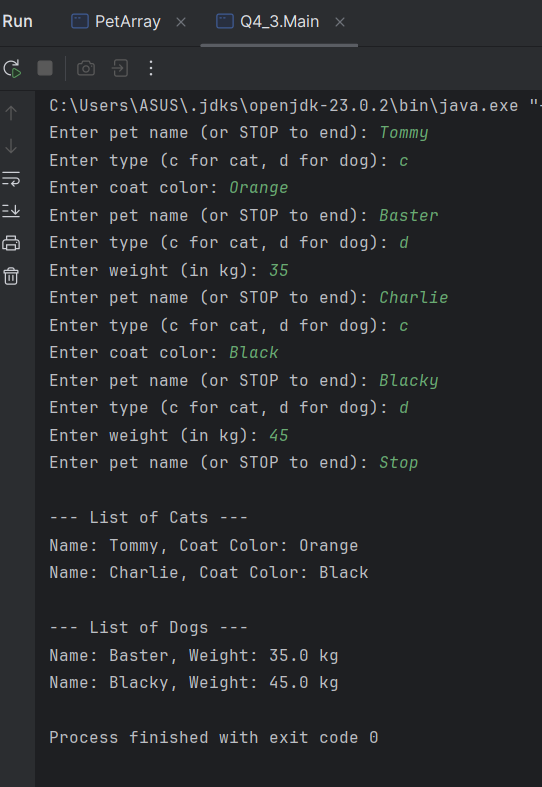


Q4 Code Method 3

package Q4\_3;  
  
import Q1.Pet;  
import Q4.Cat2;  
import Q4.Dog2;  
  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.*in*);  
 ArrayList<Pet> pets = new ArrayList<>();  
  
 // Input Loop  
 while (true) {  
 System.*out*.print("Enter pet name (or STOP to end): ");  
 String name = input.nextLine();  
 if (name.equalsIgnoreCase("STOP")) break;  
  
 System.*out*.print("Enter type (c for cat, d for dog): ");  
 String type = input.nextLine();  
  
 if (type.equalsIgnoreCase("c")) {  
 Cat2 cat = new Cat2();  
 cat.setName(name);  
 System.*out*.print("Enter coat color: ");  
 cat.setCoatColor(input.nextLine());  
 pets.add(cat);  
 }

else if (type.equalsIgnoreCase("d")) {  
 Dog2 dog = new Dog2();  
 dog.setName(name);  
 System.*out*.print("Enter weight (in kg): ");  
 try {  
 dog.setWeight(Double.*parseDouble*(input.nextLine()));  
 } catch (NumberFormatException e) {  
 System.*out*.println("Invalid weight! Setting to 0.");  
 dog.setWeight(0);  
 }  
 pets.add(dog);  
 } else {  
 System.*out*.println("Invalid type! Please enter 'c' or 'd'.");  
 }  
 }  
  
 // Grouped Output  
 System.*out*.println("\n--- List of Cats ---");  
 for (Pet p : pets) {  
 if (p instanceof Cat2 cat) {  
 System.*out*.println("Name: " + cat.getName() + ", Coat Color: " + cat.getCoatColor());  
 }  
 }  
  
 System.*out*.println("\n--- List of Dogs ---");  
 for (Pet p : pets) {  
 if (p instanceof Dog2 dog) {  
 System.*out*.println("Name: " + dog.getName() + ", Weight: " + dog.getWeight() + " kg");  
 }  
 }  
 }  
}

Output



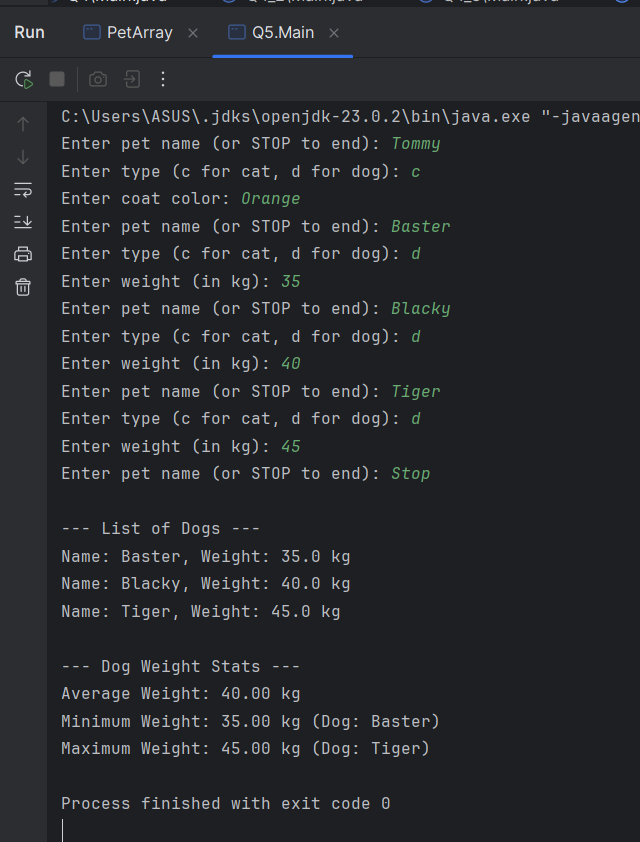
Q5 Code:

package Q5;  
  
import Q1.Pet;  
import Q4.Cat2;  
import Q4.Dog2;  
  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.*in*);  
 ArrayList<Pet> pets = new ArrayList<>();  
  
 // Step 1: Collect input for Dog and Cat objects  
 while (true) {  
 System.*out*.print("Enter pet name (or STOP to end): ");  
 String name = input.nextLine();  
 if (name.equalsIgnoreCase("STOP")) break;  
  
 System.*out*.print("Enter type (c for cat, d for dog): ");  
 String type = input.nextLine();  
  
 if (type.equalsIgnoreCase("c")) {  
 Cat2 cat = new Cat2();  
 cat.setName(name);  
 System.*out*.print("Enter coat color: ");  
 cat.setCoatColor(input.nextLine());  
 pets.add(cat);  
 } else if (type.equalsIgnoreCase("d")) {  
 Dog2 dog = new Dog2();  
 dog.setName(name);  
 System.*out*.print("Enter weight (in kg): ");

try {  
 dog.setWeight(Double.*parseDouble*(input.nextLine()));  
 } catch (NumberFormatException e) {  
 System.*out*.println("Invalid input. Weight set to 0.");  
 dog.setWeight(0);  
 }  
 pets.add(dog);  
 } else {  
 System.*out*.println("Invalid type. Please enter 'c' or 'd'.");  
 }  
}  
  
// Step 2: Create a separate Dog array  
ArrayList<Dog2> dogs = new ArrayList<>();  
for (Pet p : pets) {  
 if (p instanceof Dog2 dog) {  
 dogs.add(dog);  
 }  
}  
  
// Step 3: Print all dog details  
System.*out*.println("\n--- List of Dogs ---");  
if (dogs.isEmpty()) {  
 System.*out*.println("No dogs found.");  
} else {  
 for (Dog2 d : dogs) {  
 System.*out*.println("Name: " + d.getName() + ", Weight: " + d.getWeight() + " kg");  
 }  
  
 // Step 4: Calculate average, min, and max weights  
 double totalWeight = 0;  
 double minWeight = Double.*MAX\_VALUE*;  
 double maxWeight = Double.*MIN\_VALUE*;  
 Dog2 minDog = null;  
 Dog2 maxDog = null;

for (Dog2 d : dogs) {  
 double w = d.getWeight();  
 totalWeight += w;  
  
 if (w < minWeight) {  
 minWeight = w;  
 minDog = d;  
 }  
  
 if (w > maxWeight) {  
 maxWeight = w;  
 maxDog = d;  
 }  
 }  
  
 double average = totalWeight / dogs.size();  
  
 // Step 5: Print weight statistics  
 System.*out*.println("\n--- Dog Weight Stats ---");  
 System.*out*.printf("Average Weight: %.2f kg\n", average);  
 System.*out*.printf("Minimum Weight: %.2f kg (Dog: %s)\n", minWeight, minDog.getName());  
 System.*out*.printf("Maximum Weight: %.2f kg (Dog: %s)\n", maxWeight, maxDog.getName());  
 }  
 }  
}

Output



Q6 Code:

package Q6;  
import Q1.Pet;  
import Q4.Cat2;  
import Q4.Dog2;  
  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.*in*);  
  
 ArrayList<Pet> pets = new ArrayList<>();  
 ArrayList<Dog2> dogs = new ArrayList<>();  
 ArrayList<Cat2> cats = new ArrayList<>();  
  
 int choice;  
  
 do {  
 System.*out*.println("\n--- Pet Management 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: ");  
 choice = Integer.*parseInt*(input.nextLine());  
  
 switch (choice) {  
 case 1 -> {  
 Cat2 cat = new Cat2();  
 System.*out*.print("Enter cat name: ");  
 cat.setName(input.nextLine());  
  
 // Validate coat color (must contain at least one letter)

while (true) {  
 System.*out*.print("Enter coat color: ");  
 String color = input.nextLine();  
 if (color.matches(".\*[a-zA-Z]+.\*")) {  
 cat.setCoatColor(color);  
 break;  
 } else {  
 System.*out*.println("Invalid coat color. It must contain letters.");  
 }  
 }  
  
 cats.add(cat);  
 pets.add(cat);  
 System.*out*.println("Cat added successfully!");  
}  
  
case 2 -> {  
 Dog2 dog = new Dog2();  
 System.*out*.print("Enter dog name: ");  
 dog.setName(input.nextLine());  
  
 // Validate weight (must be >= 0)  
 while (true) {  
 System.*out*.print("Enter weight (in kg): ");  
 try {  
 double weight = Double.*parseDouble*(input.nextLine());  
 if (weight > 0) {  
 dog.setWeight(weight);  
 break;  
 } else {  
 System.*out*.println("Weight cannot be negative or Zero. Try again.");  
 }  
 } catch (NumberFormatException e) {  
 System.*out*.println("Invalid input! Please enter a valid number.");  
 }  
 }

dogs.add(dog);  
 pets.add(dog);  
 System.*out*.println("Dog added successfully!");  
}  
  
case 3 -> {  
 System.*out*.print("Enter cat name to remove: ");  
 String nameToRemove = input.nextLine();  
  
 Cat2 toRemove = null;  
 for (Cat2 c : cats) {  
 if (c.getName().equalsIgnoreCase(nameToRemove)) {  
 toRemove = c;  
 break;  
 }  
 }  
  
 if (toRemove != null) {  
 cats.remove(toRemove);  
 pets.remove(toRemove);  
 System.*out*.println("Cat removed successfully.");  
 } else {  
 System.*out*.println("Cat not found.");  
 }  
}  
  
case 4 -> {  
 System.*out*.print("Enter dog name to remove: ");  
 String nameToRemove = input.nextLine();  
  
 Dog2 toRemove = null;  
 for (Dog2 d : dogs) {  
 if (d.getName().equalsIgnoreCase(nameToRemove)) {  
 toRemove = d;  
 break;  
 }  
 }

if (toRemove != null) {  
 dogs.remove(toRemove);  
 pets.remove(toRemove);  
 System.*out*.println("Dog removed successfully.");  
 } else {  
 System.*out*.println("Dog not found.");  
 }  
 }  
  
 case 0 -> System.*out*.println("Exiting...");  
  
 default -> System.*out*.println("Invalid choice. Please try again.");  
 }  
  
 // Show updated lists  
 System.*out*.println("\n--- Current Cats ---");  
 for (Cat2 c : cats) {  
 System.*out*.println("->Name: " + c.getName() + ", Coat Color: " + c.getCoatColor());  
 }  
  
 System.*out*.println("\n--- Current Dogs ---");  
 for (Dog2 d : dogs) {  
 System.*out*.println("->Name: " + d.getName() + ", Weight: " + d.getWeight() + " kg");  
 }  
  
 } while (choice != 0);  
 }  
}

Output:

