Q1.

Code:

```
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.";
    }
}
```

```
package Q_01;

public class Cat extends Pet {

    public String speak() {

        return " ";
    }
}
```

```
package Q_01;

public class Dog extends Pet {

    public String speak() {

        return " ";
    }
}
```

```
package Q_01;

public class Main {
    public static void main(String[] args) {
        Dog obj1 = new Dog();
        obj1.setName("Tommy");

        System.out.println(obj1.speak());
        System.out.println(obj1.getName());

        Cat obj2 = new Cat();
        obj2.setName("Kitty");
        System.out.println(obj2.speak());
        System.out.println(obj2.getName());
    }
}
```

Output:

Q2.

Code:

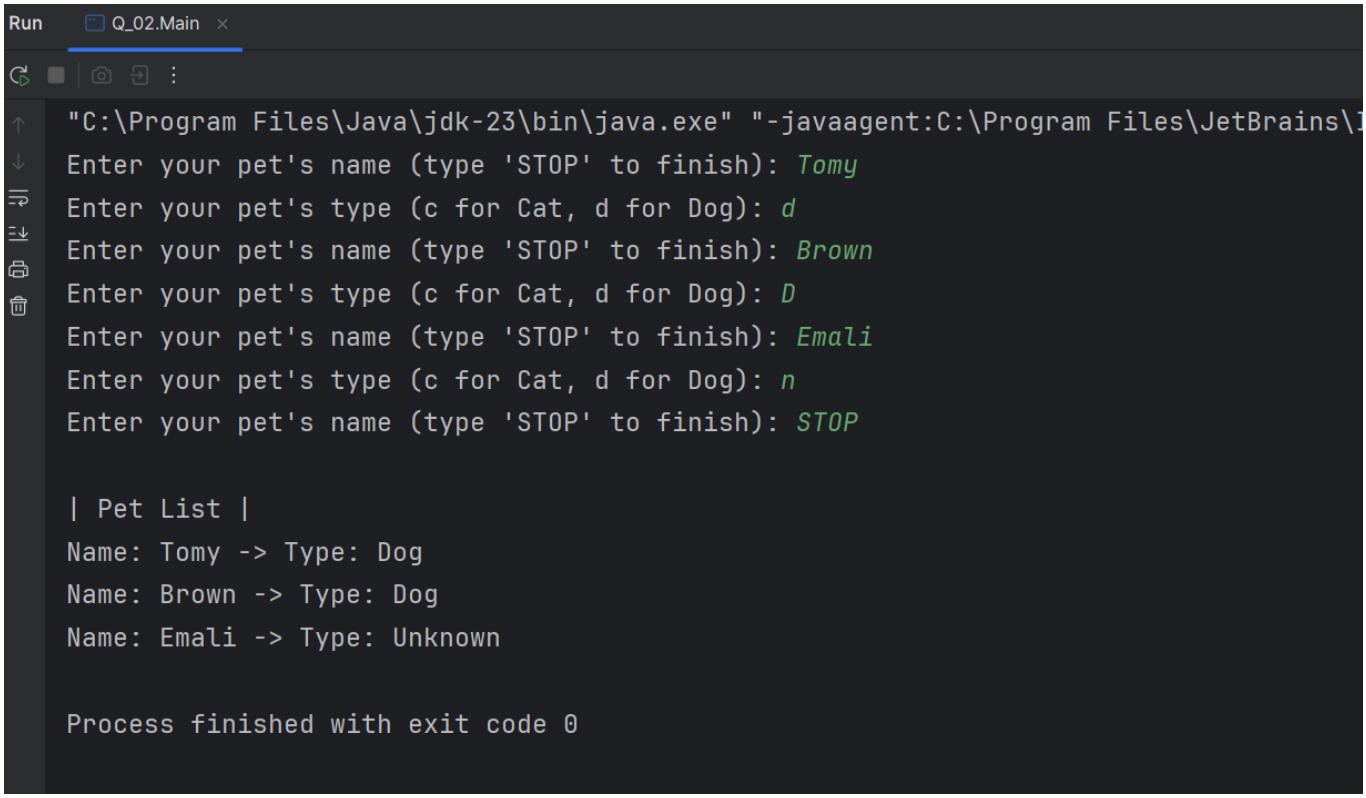
```
package Q_02;

class Pet {
    protected String name;

    public Pet(String name) {
        this.name = name;
    }

    public String getType() {
        return "Unknown";
    }

    public String getName() {
        return name;
    }
}
```

```
package Q_02;

class Cat extends Pet {
    public Cat(String name) {
        super(name);
    }


    public String getType() {
        return "Cat";
    }
}
```

```
package Q_02;

class Dog extends Pet {
    public Dog(String name) {
        super(name);
    }


    public String getType() {
        return "Dog";
    }
}
```

```
package Q_02;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List <Pet> pets = new ArrayList<>();

        while (true) {
            System.out.print("Enter your pet's name (type 'STOP' to finish): ");
            String name = scanner.nextLine();

            if (name.equalsIgnoreCase("STOP")) {
                break;
            }

            System.out.print("Enter your pet's type (c for Cat, d for Dog): ");
            char type = scanner.next().charAt(0);
            scanner.nextLine();

            if (type == 'c' || type == 'C') {
                pets.add(new Cat(name));
            } else if (type == 'd' || type == 'D') {
                pets.add(new Dog(name));
            } else {
                pets.add(new Pet(name)); // Unknown type
            }
        }

        System.out.println("\n| Pet List |");
        for (Pet pet : pets) {
            System.out.println("Name: " + pet.getName() + " -> Type: " +
pet.getType());
        }
    }
}
```

Output:



Q3.

Code:

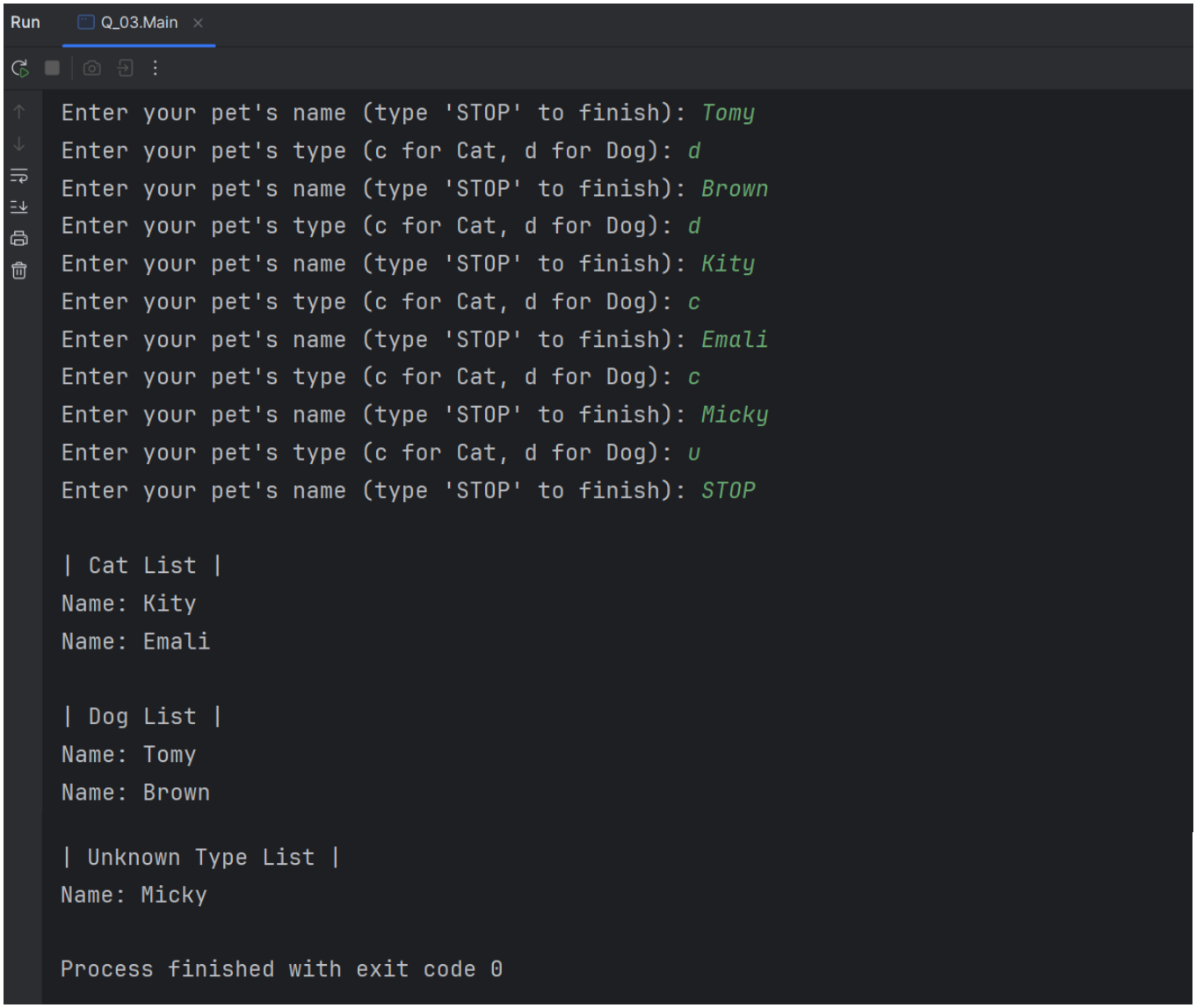
```
package Q_03;

class Pet {
    protected String name;

    public Pet(String name) {
        this.name = name;
    }

    public String getType() {
        return "Unknown";
    }

    public String getName() {
        return name;
    }
}
```

```
package Q_03;

class Cat extends Pet {
    public Cat(String name) {
        super(name);
    }


    public String getType() {
        return "Cat";
    }
}
```

```
package Q_03;

class Dog extends Pet {
    public Dog(String name) {
        super(name);
    }


    public String getType() {
        return "Dog";
    }
}
```

```
package Q_03;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List <Pet> pets = new ArrayList<>();

        while (true) {
            System.out.print("Enter your pet's name (type 'STOP' to finish): ");
            String name = scanner.nextLine();

            if (name.equalsIgnoreCase("STOP")) {
                break;
            }

            System.out.print("Enter your pet's type (c for Cat, d for Dog): ");
            char type = scanner.next().charAt(0);
            scanner.nextLine();

            if (type == 'c' || type == 'C') {
                pets.add(new Cat(name));
            } else if (type == 'd' || type == 'D') {
                pets.add(new Dog(name));
            } else {
                pets.add(new Pet(name));
            }
        }

        System.out.println("\n| Cat List |");
        for (Pet pet : pets) {
            if (pet instanceof Cat) {
                System.out.println("Name: " + pet.getName());
            }
        }

        System.out.println("\n| Dog List |");
        for (Pet pet : pets) {
            if (pet instanceof Dog) {
                System.out.println("Name: " + pet.getName());
            }
        }

        System.out.println("\n| Unknown Type List |");
        for (Pet pet : pets) {
            if (!(pet instanceof Cat) && !(pet instanceof Dog)) {
                System.out.println("Name: " + pet.getName());
            }
        }

    }

}
```

Output:



Q4.

Code:

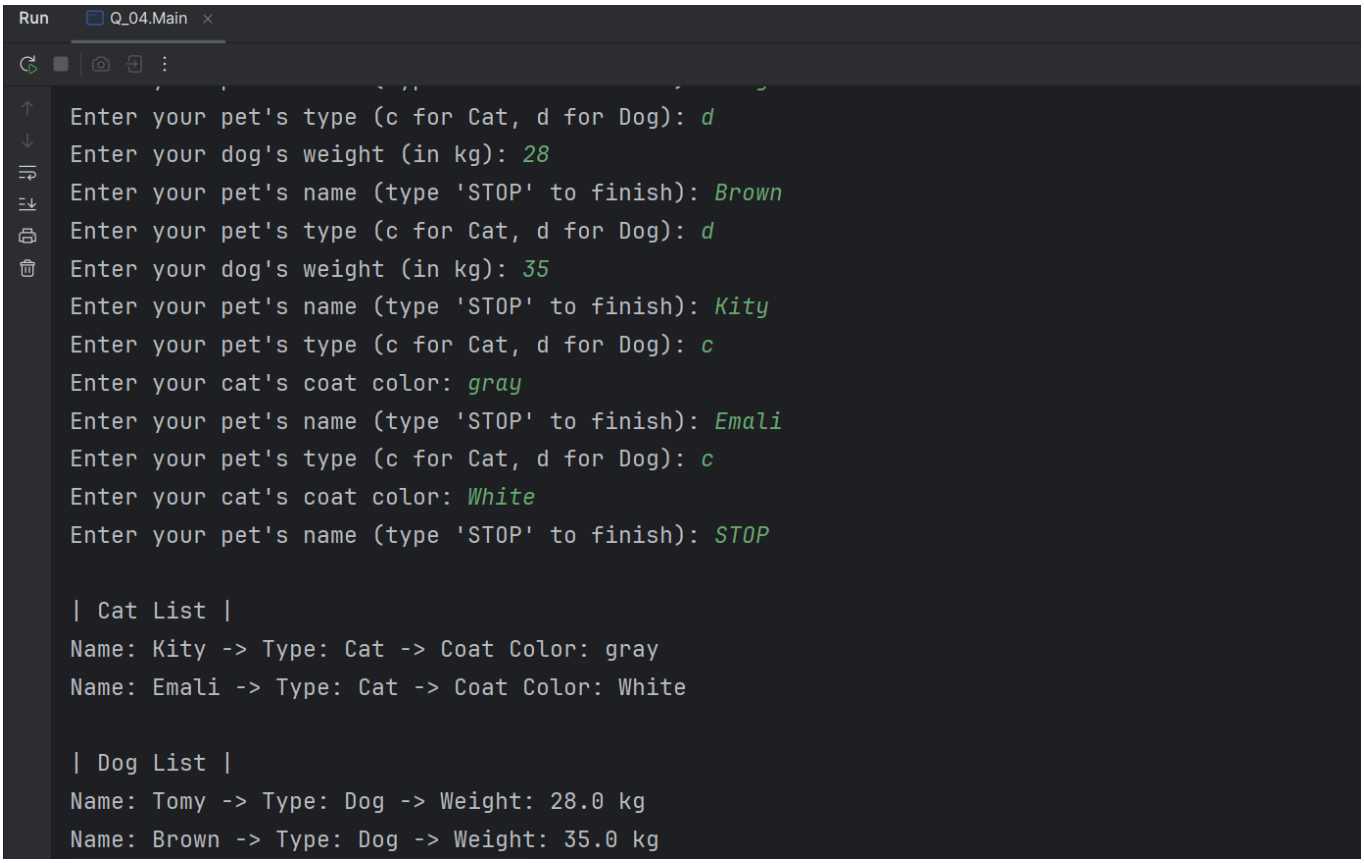
```
package Q_04;

class Pet {
    protected String name;

    public Pet(String name) {
        this.name = name;
    }

    public String getType() {
        return "Unknown";
    }

    public String getName() {
        return name;
    }
}
```

```
package Q_04;

class Dog extends Pet {
    private double weight;

    public Dog(String name, double weight) {
        super(name);
        this.weight = weight;
    }

    public String getType() {
        return "Dog";
    }

    public double getWeight() {
        return weight;
    }

    public void setWeight(double weight) {
        this.weight = weight;
    }
}
```

```
package Q_04;

class Cat extends Pet {
    private String coatColor;

    public Cat(String name, String coatColor) {
        super(name);
        this.coatColor = coatColor;
    }


    public String getType() {
        return "Cat";
    }

    public String getCoatColor() {
        return coatColor;
    }

    public void setCoatColor(String coatColor) {
        this.coatColor = coatColor;
    }
}
```

```
package Q_04;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List <Pet> pets = new ArrayList<>();

        while (true) {
            System.out.print("Enter your pet's name (type 'STOP' to finish): ");
            String name = scanner.nextLine();

            if (name.equalsIgnoreCase("STOP")) {
                break;
            }

            System.out.print("Enter your pet's type (c for Cat, d for Dog): ");
            char type = scanner.next().charAt(0);
            scanner.nextLine();

            if (type == 'c' || type == 'C') {
                System.out.print("Enter your cat's coat color: ");
                String coatColor = scanner.nextLine();
                pets.add(new Cat(name, coatColor));
            } else if (type == 'd' || type == 'D') {
                System.out.print("Enter your dog's weight (in kg): ");
                double weight = scanner.nextDouble();
                scanner.nextLine();
                pets.add(new Dog(name, weight));
            } else {
                pets.add(new Pet(name)); // Unknown
            }

        }

        System.out.println("\n| Cat List |");
        for (Pet pet : pets) {
            if (pet instanceof Cat) {
                Cat cat = (Cat) pet;
                System.out.println("Name: " + cat.getName() + " -> Type: Cat -> Coat
Color: " + cat.getCoatColor());
            }
        }

        System.out.println("\n| Dog List |");
        for (Pet pet : pets) {
            if (pet instanceof Dog) {
                Dog dog = (Dog) pet;
                System.out.println("Name: " + dog.getName() + " -> Type: Dog -> Weight:
" + dog.getWeight() + " kg");
            }
        }


    }

}
```

Output:



Q5.

Code:

```
package Q_05;

class Pet {
    protected String name;

    public Pet(String name) {
        this.name = name;
    }

    public String getType() {
        return "Unknown";
    }

    public String getName() {
        return name;
    }
}
```

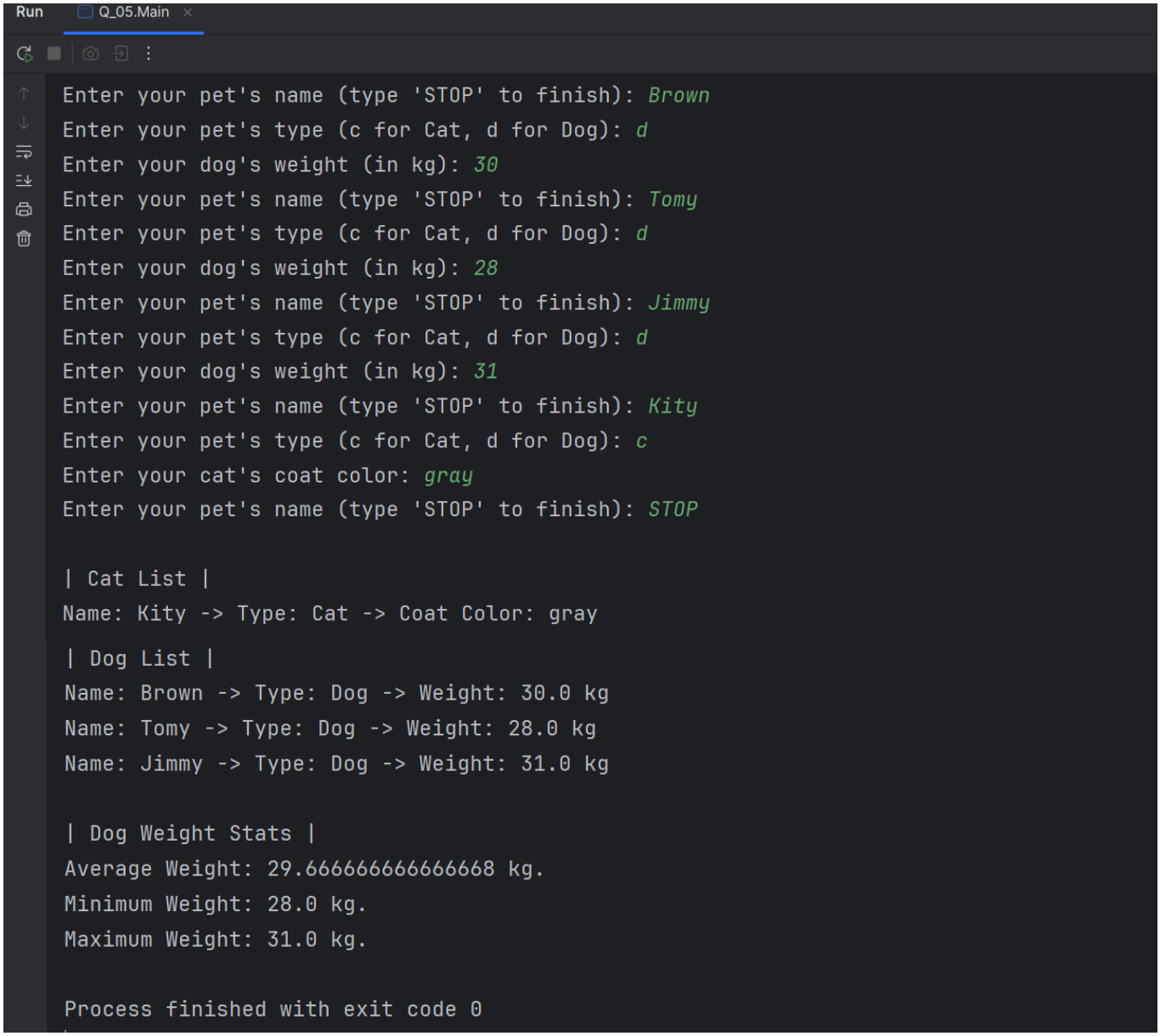
```
package Q_05;

class Dog extends Pet {
    private double weight;

    public Dog(String name, double weight) {
        super(name);
        this.weight = weight;
    }

    public String getType() {
        return "Dog";
    }

    public double getWeight() {
        return weight;
    }

    public void setWeight(double weight) {
        this.weight = weight;
    }
}
```

```
package Q_05;

class Cat extends Pet {
    private String coatColor;

    public Cat(String name, String coatColor) {
        super(name);
        this.coatColor = coatColor;
    }

    public String getType() {
        return "Cat";
    }

    public String getCoatColor() {
        return coatColor;
    }

    public void setCoatColor(String coatColor) {
        this.coatColor = coatColor;
    }
}
```

```
package Q_05;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List <Pet> pets = new ArrayList<>();

        while (true) {
            System.out.print("Enter your pet's name (type 'STOP' to finish): ");
            String name = scanner.nextLine();

            if (name.equalsIgnoreCase("STOP")) {
                break;
            }

            System.out.print("Enter your pet's type (c for Cat, d for Dog): ");
            char type = scanner.next().charAt(0);
            scanner.nextLine();

            if (type == 'c' || type == 'C') {
                System.out.print("Enter your cat's coat color: ");
                String coatColor = scanner.nextLine();
                pets.add(new Cat(name, coatColor));
            } else if (type == 'd' || type == 'D') {
                System.out.print("Enter your dog's weight (in kg): ");
                double weight = scanner.nextDouble();
                scanner.nextLine();
                pets.add(new Dog(name, weight));
            } else {
                pets.add(new Pet(name)); // Unknown
            }

        }
        List<Dog> dogList = new ArrayList<>();
```

```
        for (Pet pet : pets) {
            if (pet instanceof Dog) {
                dogList.add((Dog) pet);
            }
        }

        System.out.println("\n| Cat List |");
        for (Pet pet : pets) {
            if (pet instanceof Cat) {
                Cat cat = (Cat) pet;
                System.out.println("Name: " + cat.getName() + " -> Type: Cat -> Coat
Color: " + cat.getCoatColor());
            }
        }

        System.out.println("\n| Dog List |");
        for (Pet pet : pets) {
            if (pet instanceof Dog) {
                Dog dog = (Dog) pet;
                System.out.println("Name: " + dog.getName() + " -> Type: Dog -> Weight:
" + dog.getWeight() + " kg");
            }
        }
        if (!dogList.isEmpty()) {
            double totalWeight = 0;
            double minWeight = dogList.get(0).getWeight();
            double maxWeight = dogList.get(0).getWeight();

            for (Dog dog : dogList) {
                double weight = dog.getWeight();
                totalWeight += weight;

                if (weight < minWeight) {
                    minWeight = weight;
                }

                if (weight > maxWeight) {
                    maxWeight = weight;
                }
            }

            double averageWeight = totalWeight / dogList.size();

            System.out.println("\n| Dog Weight Stats |");
            System.out.println("Average Weight: " + averageWeight + " kg.");
            System.out.println("Minimum Weight: " + minWeight + " kg.");
            System.out.println("Maximum Weight: " + maxWeight + " kg.");
        } else {
            System.out.println("\nNo dogs were entered to calculate weight
statistics.");
        }



    }

}
```

Output:


Run
Q_05.Main
Enter your pet's name (type 'STOP' to finish): Brown
Enter your pet's type (c for Cat, d for Dog): d
Enter your dog's weight (in kg): 30
Enter your pet's name (type 'STOP' to finish): Tomy
Enter your pet's type (c for Cat, d for Dog): d
Enter your dog's weight (in kg): 28
Enter your pet's name (type 'STOP' to finish): Jimmy
Enter your pet's type (c for Cat, d for Dog): d
Enter your dog's weight (in kg): 31
Enter your pet's name (type 'STOP' to finish): Kity
Enter your pet's type (c for Cat, d for Dog): c
Enter your cat's coat color: gray
Enter your pet's name (type 'STOP' to finish): STOP
| Cat List |
Name: Kity -> Type: Cat -> Coat Color: gray
| Dog List |
Name: Brown -> Type: Dog -> Weight: 30.0 kg
Name: Tomy -> Type: Dog -> Weight: 28.0 kg
Name: Jimmy -> Type: Dog -> Weight: 31.0 kg
| Dog Weight Stats |
Average Weight: 29.666666666666668 kg.
Minimum Weight: 28.0 kg.
Maximum Weight: 31.0 kg.
Process finished with exit code 0

Q6.

```
package Q_06;

class Pet {
    protected String name;

    public Pet(String name) {
        this.name = name;
    }

    public String getType() {
        return "Unknown";
    }

    public String getName() {
        return name;
    }
}
```

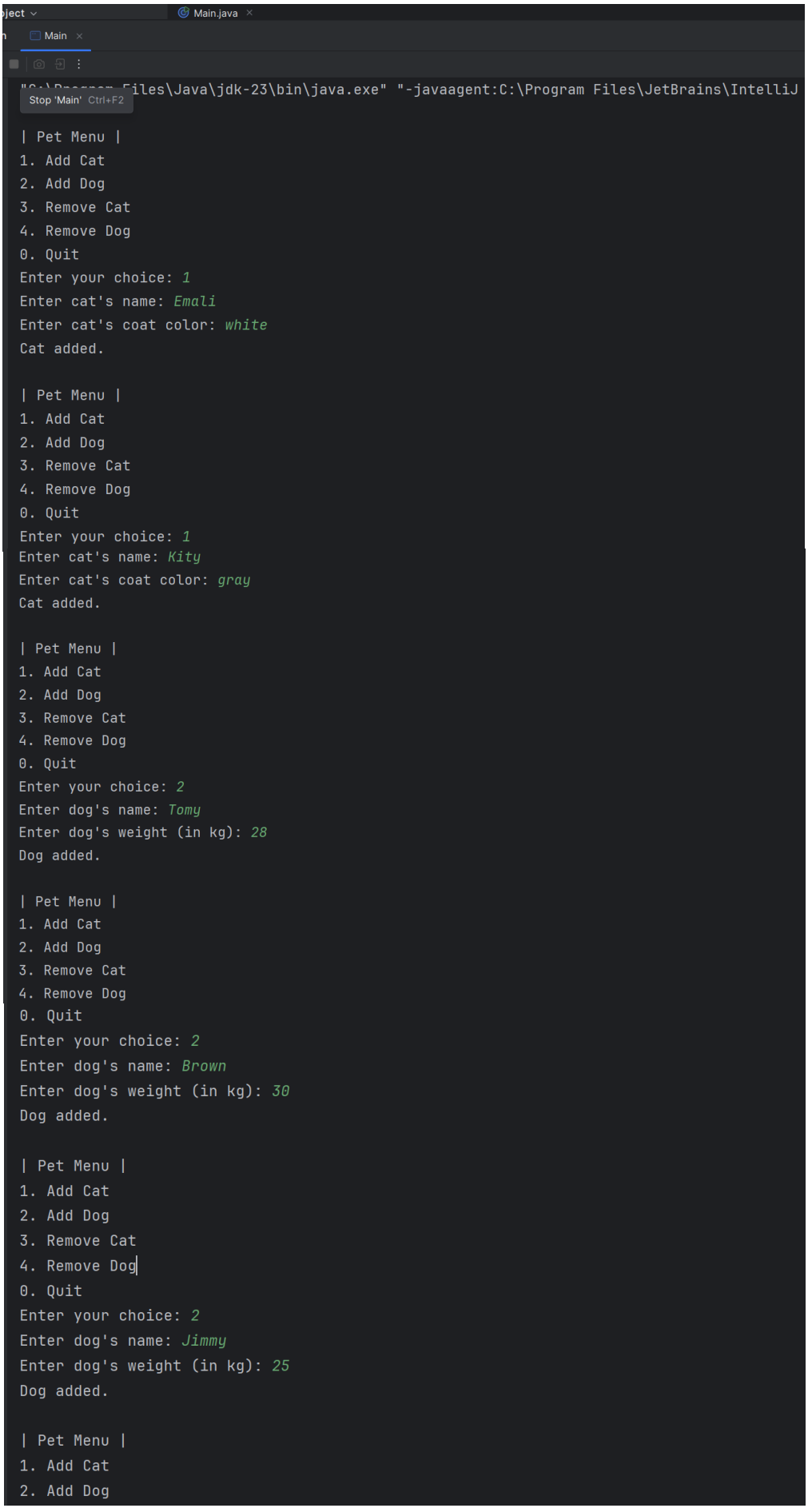
```
package Q_06;

class Dog extends Pet {
    private double weight;

    public Dog(String name, double weight) {
        super(name);
        this.weight = weight;
    }

    public String getType() {
        return "Dog";
    }

    public double getWeight() {
        return weight;
    }

    public void setWeight(double weight) {
        this.weight = weight;
    }
}
```

```
package Q_06;

class Cat extends Pet {
    private String coatColor;

    public Cat(String name, String coatColor) {
        super(name);
        this.coatColor = coatColor;
    }

    public String getType() {
        return "Cat";
    }

    public String getCoatColor() {
        return coatColor;
    }

    public void setCoatColor(String coatColor) {
        this.coatColor = coatColor;
    }
}
```

```
package Q_06;

import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<Pet> pets = new ArrayList<>();
        List<Dog> dogList = new ArrayList<>();
        List<Cat> catList = new ArrayList<>();

        int choice;

        do {
            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: ");
            choice = scanner.nextInt();
            scanner.nextLine();

            switch (choice) {
                case 1: // Add Cat
                    System.out.print("Enter cat's name: ");
                    String catName = scanner.nextLine();
                    System.out.print("Enter cat's coat color: ");
                    String coatColor = scanner.nextLine();

                    Cat cat = new Cat(catName, coatColor);
                    pets.add(cat);
                    catList.add(cat);
                    System.out.println("Cat added.");
                    break;

                case 2: // Add Dog
                    System.out.print("Enter dog's name: ");
                    String dogName = scanner.nextLine();
                    System.out.print("Enter dog's weight (in kg): ");
                    double weight = scanner.nextDouble();
                    scanner.nextLine();

                    Dog dog = new Dog(dogName, weight);
                    pets.add(dog);
                    dogList.add(dog);
                    System.out.println("Dog added.");
                    break;

                case 3: // Remove Cat
                    System.out.print("Enter the name of the cat to remove: ");
                    String removeCatName = scanner.nextLine();

                    catList.removeIf(c -> c.getName().equalsIgnoreCase(removeCatName));
                    pets.removeIf(p -> p instanceof Cat &&
p.getName().equalsIgnoreCase(removeCatName));
                    System.out.println("Cat removed...");
                    break;
```

```
                case 4: // Remove Dog
                    System.out.print("Enter the name of the dog to remove: ");
                    String removeDogName = scanner.nextLine();

                    dogList.removeIf(d -> d.getName().equalsIgnoreCase(removeDogName));
                    pets.removeIf(p -> p instanceof Dog &&
p.getName().equalsIgnoreCase(removeDogName));
                    System.out.println("Dog removed...");
                    break;

                case 0:
                    System.out.println("Exiting program.");
                    break;

                default:
                    System.out.println("Invalid choice. Please enter 0-4.");
            }

        } while (choice != 0);

        // List of currently included dogs and cats
        System.out.println("\n| Final Cat List |");
        if (catList.isEmpty()) {
            System.out.println("No cats in the list!");
        } else {
            for (Cat c : catList) {
                System.out.println("Name: " + c.getName() + " -> Coat Color: " +
c.getCoatColor());
            }
        }

        System.out.println("\n| Final Dog List |");
        if (dogList.isEmpty()) {
            System.out.println("No dogs in the list!");
        } else {
            for (Dog d : dogList) {
                System.out.println("Name: " + d.getName() + " -> Weight: " +
d.getWeight() + " kg");
            }
        }


    }
}
```

Output:


Main.java
Main
Stop 'Main' Ctrl+F2
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ
| Pet Menu |
1. Add Cat
2. Add Dog
3. Remove Cat
4. Remove Dog
0. Quit
Enter your choice: 1
Enter cat's name: Emali
Enter cat's coat color: white
Cat added.
| Pet Menu |
1. Add Cat
2. Add Dog
3. Remove Cat
4. Remove Dog
0. Quit
Enter your choice: 1
Enter cat's name: Kity
Enter cat's coat color: gray
Cat added.
| Pet Menu |
1. Add Cat
2. Add Dog
3. Remove Cat
4. Remove Dog
0. Quit
Enter your choice: 2
Enter dog's name: Tomy
Enter dog's weight (in kg): 28
Dog added.
| Pet Menu |
1. Add Cat
2. Add Dog
3. Remove Cat
4. Remove Dog
0. Quit
Enter your choice: 2
Enter dog's name: Brown
Enter dog's weight (in kg): 30
Dog added.
| Pet Menu |
1. Add Cat
2. Add Dog
3. Remove Cat
4. Remove Dog
0. Quit
Enter your choice: 2
Enter dog's name: Jimmy
Enter dog's weight (in kg): 25
Dog added.
| Pet Menu |
1. Add Cat
2. Add Dog

```
3. Remove Cat
4. Remove Dog
0. Quit
Enter your choice: 3
Enter the name of the cat to remove: Emali
Cat removed...

| Pet Menu |
1. Add Cat
2. Add Dog
3. Remove Cat
4. Remove Dog
0. Quit
Enter your choice: 4
Enter the name of the dog to remove: Tomy
Dog removed...

| Pet Menu |
1. Add Cat
2. Add Dog
3. Remove Cat
4. Remove Dog
0. Quit
Enter your choice: 0
Exiting program.

| Final Cat List |
Name: Kity -> Coat Color: gray

| Final Dog List |
Name: Brown -> Weight: 30.0 kg
Name: Jimmy -> Weight: 25.0 kg

Process finished with exit code 0
```