Q1.

Code:

Q_01.java

```java
package Q_01;

public class Q_01 {
    public static void main(String[] args) {
        //creating cat object
        Cat myCat = new Cat();
        //creating dog object
        Dog myDog = new Dog();
        //setting name for cat
        myCat.setName("Puff Puff");
        System.out.println(myCat.getName() + " says " + myCat.speak());
        //setting name for dog
        myDog.setName("Fifi");
        System.out.println(myDog.getName() + " says " + myDog.speak());
    }
}
```

Pet.java

```java
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.";
    }
}
```

Dog.java

```java
package Q_01;

public class Dog extends Pet{

    @Override
    public String speak() {
        return " ";
    }
}
```

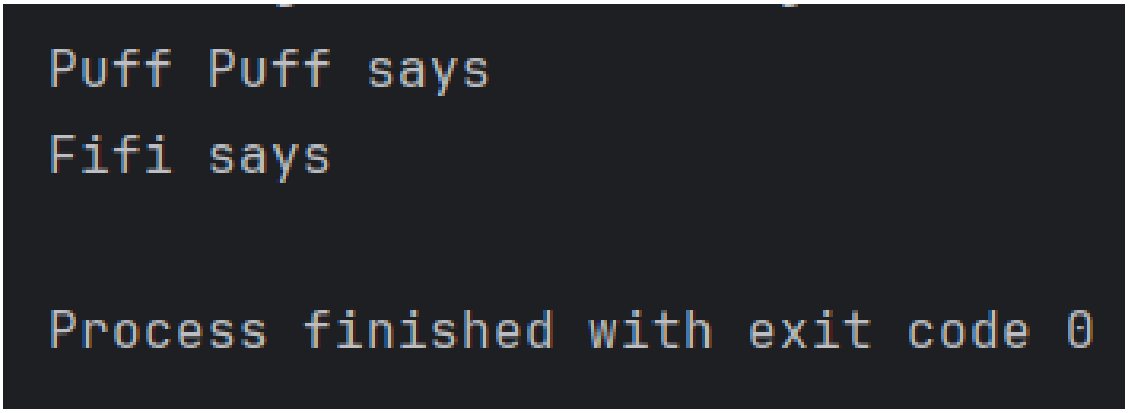
Cat.java

```
package Q_01;

public class Cat extends Pet{

    @Override
    public String speak() {
        return " ";
    }
}
```

Output:



Q2.

Code:

```
package Q_02;

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

//import the necessary classes
import Q_01.Pet;
import Q_01.Dog;
import Q_01.Cat;

public class Q_02 {
    public static void main(String[] args) {
        //create a Scanner object to read input
        Scanner scanner = new Scanner(System.in);
        //create an ArrayList to store the pets
        ArrayList<Pet> petList = new ArrayList<>();
        //create a Pet object
        Pet pet;

        while(true)
        {
            System.out.println("Enter pet name ('STOP' to stop)");
            String name = scanner.nextLine();
            //check if the user wants to stop
```

```
            {
                break;
            }
            System.out.println("Enter pet type ('d' for Dog and 'c' for Cat)");
            String type = scanner.nextLine();

            //create a new Pet object based on the type
            if(type.equals("d"))
            {
                pet = new Dog();
            }
            else if(type.equals("c"))
            {
                pet = new Cat();
            }
            else
            {
                System.out.println("Invalid type.");
                continue;
            }
            //set the name of the pet
            pet.setName(name);
            //add the pet to the list
            petList.add(pet);
        }

        for(Pet p : petList)
        {
            System.out.print("Name: "+p.getName());
            //check the type of the pet
            if (p instanceof Dog)
                System.out.println("\tType: Dog");
            else
                System.out.println("\tType: Cat");
        }
    }
}
```

Output:

```
Enter pet name ('STOP' to stop)
Scooby
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet name ('STOP' to stop)
Kitty
Enter pet type ('d' for Dog and 'c' for Cat)
c
Enter pet name ('STOP' to stop)
Tommy
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet name ('STOP' to stop)
Twinkle
Enter pet type ('d' for Dog and 'c' for Cat)
c
Enter pet name ('STOP' to stop)
Tiny
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet name ('STOP' to stop)
STOP
Name: Scooby    Type: Dog
Name: Kitty Type: Cat
Name: Tommy Type: Dog
Name: Twinkle   Type: Cat
Name: Tiny  Type: Dog

Process finished with exit code 0
```

Q3.

Code:

```java
package Q_03;

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

//import the necessary classes
import Q_01.Pet;
import Q_01.Cat;
import Q_01.Dog;

public class Q_03 {
    public static void main(String[] args) {
        //create scanner object for input
        Scanner input = new Scanner(System.in);
        //create an ArrayList to store pets
        ArrayList<Pet> petList = new ArrayList<>();
        //create a pet object
        Pet pet;

        while(true)
        {
            System.out.println("Enter pet name ('STOP' to stop)");
            String name = input.nextLine();
            //check if the user wants to stop
            if(name.equals("STOP"))
            {
                break;
            }
            System.out.println("Enter pet type ('d' for Dog and 'c' for Cat)");
            String type = input.nextLine();
            //create a new Pet object based on the type
            if(type.equals("d"))
            {
                pet = new Dog();
            }
            else if(type.equals("c"))
            {
                pet = new Cat();
            }
            else
            {
                System.out.println("Invalid type.");
                continue;
            }
            //set the name of the pet
            pet.setName(name);
            //add the pet to the list
            petList.add(pet);
        }
        System.out.println("Names of cats");

        //print the names of cats
        for(Pet p : petList)
        {
            if(p instanceof Cat)
                System.out.println(p.getName());
        }
```

```
        System.out.println("\nNames of dogs");
        //print the names of dogs
        for(Pet p : petList)
        {
            if(p instanceof Dog)
                System.out.println(p.getName());
        }
    }
}
```

Output:

```
Enter pet name ('STOP' to stop)
Scooby
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet name ('STOP' to stop)
Kitty
Enter pet type ('d' for Dog and 'c' for Cat)
c
Enter pet name ('STOP' to stop)
Tommy
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet name ('STOP' to stop)
Twinkle
Enter pet type ('d' for Dog and 'c' for Cat)
c
Enter pet name ('STOP' to stop)
STOP
Names of cats
Kitty
Twinkle

Names of dogs
Scooby
Tommy

Process finished with exit code 0
```

Q4.

Code:

Q_04.java

```java
package Q_04;

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

//import Pet calss
import Q_01.Pet;

public class Q_04 {
    public static void main(String[] args) {
        //create  scanner object for input
        Scanner input = new Scanner(System.in);
        //create an ArrayList to store pets
        ArrayList<Pet> petList = new ArrayList<>();
        //create a pet object
        Pet pet;

        while(true)
        {
            System.out.println("Enter pet name ('STOP' to stop)");
            String name = input.nextLine();
            //check if the user wants to stop
            if(name.equals("STOP"))
            {
                break;
            }
            System.out.println("Enter pet type ('d' for Dog and 'c' for Cat)");
            String type = input.nextLine();

            //create a new Cat object based on the type
            if(type.equals("c"))
            {
                System.out.println("Enter pet coat color");
                String coatColor = input.nextLine();
                pet = new Cat();
                //set the name of the Cat
                pet.setName(name);
                //typecasting pet object to add coatcolor
                ((Cat)pet).setCoatColor(coatColor);
            }
            //create a new Dog object based on the type
            else if(type.equals("d"))
            {
                System.out.println("Enter pet weight (kg)");
                double weight = input.nextDouble();
                input.nextLine(); // Consume the newline character

                pet = new Dog();
                //set the name of the Dog
                pet.setName(name);
                //typecasting pet object to add weight
                ((Dog)pet).setWeight(weight);
            }
            else
```

```
            //add the pet to the list
            petList.add(pet);
        }
        System.out.println("List of cats");
        //print cats
        for(Pet p : petList)
        {
            if(p instanceof Cat)
                System.out.println("Name: "+p.getName()+"\tType: Cat "+"\tCoat Color:
"+((Cat)p).getCoatColor());
        }

        System.out.println("\nList of dogs");
        //print dogs
        for(Pet p : petList)
        {
            if(p instanceof Dog)
                System.out.println("Name: "+p.getName()+"\tType: Dog "+"\tWeight:
"+((Dog)p).getWeight());
        }
    }
}
```

Dog.java

```
package Q_04;
//import Pet class
import Q_01.Pet;

public class Dog extends Pet {
    private double weight;

    public double getWeight() {
        return weight;
    }

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

    @Override
    public String speak() {
        return " ";
    }
}
```

Cat.java

```java
package Q_04;
//import Pet class
import Q_01.Pet;

public class Cat extends Pet {
    private String coatColor;

    public String getCoatColor() {
        return coatColor;
    }

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

    @Override
    public String speak() {
        return " ";
    }
}
```

Output:

```
Scooby
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet weight (kg)
34
Enter pet name ('STOP' to stop)
Kitty
Enter pet type ('d' for Dog and 'c' for Cat)
c
Enter pet coat color
White
Enter pet name ('STOP' to stop)
Tommy
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet weight (kg)
26
Enter pet name ('STOP' to stop)
Twinkle
Enter pet type ('d' for Dog and 'c' for Cat)
c
Enter pet coat color
Orange
Enter pet name ('STOP' to stop)
STOP
```

```
List of cats
Name: Kitty Type: Cat   Coat Color: White
Name: Twinkle   Type: Cat   Coat Color: Orange

List of dogs
Name: Scooby    Type: Dog   Weight: 34.0
Name: Tommy Type: Dog   Weight: 26.0

Process finished with exit code 0
```

Q5.

Code:

```
package Q_05;

import java.text.DecimalFormat;
import java.util.ArrayList;
import java.util.Scanner;

//import necessary classes
import Q_01.Pet;
import Q_04.Dog;
import Q_04.Cat;

public class Q_05 {
    public static void main(String[] args) {
        //create  scanner object for input
        Scanner input = new Scanner(System.in);
        //create decimal format object
        DecimalFormat df = new DecimalFormat("#.##");
        //create an ArrayList to store pets
        ArrayList<Pet> petList = new ArrayList<>();
        //create an ArrayList to store dog weight
        ArrayList<Dog> dogList = new ArrayList<>();
        //create a pet object
        Pet pet;

        while(true)
        {
            System.out.println("Enter pet name ('STOP' to stop)");
            String name = input.nextLine();
            //check if the user wants to stop
            if(name.equals("STOP"))
            {
                break;
            }
            System.out.println("Enter pet type ('d' for Dog and 'c' for Cat)");
            String type = input.nextLine();
            //create a new Cat object based on the type
            if(type.equals("c"))
            {
                System.out.println("Enter pet coat color");
                String coatColor = input.nextLine();
                pet = new Cat();
                //set the name of the Cat
                pet.setName(name);
                //typecasting pet object to add coatcolor
                ((Cat)pet).setCoatColor(coatColor);
            }
            //create a new Dog object based on the type
            else if(type.equals("d"))
            {
                System.out.println("Enter pet weight (kg)");
                double weight = input.nextDouble();
                input.nextLine(); // Consume the newline character
                pet = new Dog();
                Dog dog = new Dog();
                //set the name of the Dog
                pet.setName(name);
                //typecasting pet object to add weight
                ((Dog)pet).setWeight(weight);
```

```
                //set the name of the dog for the dogList Array
                dog.setName(name);
                //set the weight of the dog for the dogList Array
                dog.setWeight(weight);
                dogList.add(dog);
            }
            else
            {
                System.out.println("Invalid type.");
                continue;
            }
            //add the pet to the list
            petList.add(pet);
        }
        System.out.println("List of cats");
        //print cats
        for(Pet p : petList)
        {
            if(p instanceof Cat)
                System.out.println("Name: "+p.getName()+"\tType: Cat "+"\tCoat Color:
"+((Cat)p).getCoatColor());
        }
        System.out.println("\nList of dogs");
        //print dogs
        for(Pet p : petList)
        {
            if(p instanceof Dog)
                System.out.println("Name: "+p.getName()+"\tType: Dog "+"\tWeight:
"+((Dog)p).getWeight()+" kg");
        }
        double min = dogList.getFirst().getWeight();
        double max = dogList.getFirst().getWeight();
        double total = 0;
        //calculate the min, max and total weight
        for(Dog d : dogList)
        {
            double weight = d.getWeight();
            total += weight;
            if(weight < min)
                min = weight;
            if(weight > max)
                max = weight;
        }
        double avg = total / dogList.size(); //calculate average weight
        //print min, max and average weight
        System.out.println("\nDog max weight: "+max+" kg\nMin weight: "+min+"
kg\nAverage weight: "+df.format(avg)+" kg");
    }

}
```

Output:

```
Enter pet name ('STOP' to stop)
Tommy
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet weight (kg)
34
Enter pet name ('STOP' to stop)
Kitty
Enter pet type ('d' for Dog and 'c' for Cat)
c
Enter pet coat color
White
Enter pet name ('STOP' to stop)
Scooby
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet weight (kg)
42
Enter pet name ('STOP' to stop)
Tiny
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet weight (kg)
21
Enter pet name ('STOP' to stop)
STOP
List of cats
```

```
Name: Kitty Type: Cat   Coat Color: White

List of dogs
Name: Tommy Type: Dog   Weight: 34.0 kg
Name: Scooby    Type: Dog   Weight: 42.0 kg
Name: Tiny  Type: Dog   Weight: 21.0 kg

Dog max weight: 42.0 kg
Min weight: 21.0 kg
Average weight: 32.33 kg

Process finished with exit code 0
```

Q6.

Code:

```java
package Q_06;

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

//import necessary classes
import Q_01.Pet;
import Q_04.Dog;
import Q_04.Cat;

import static java.lang.System.exit;

public class Q_06 {
    public static void main(String[] args) {
        //create  scanner object for input
        Scanner input = new Scanner(System.in);
        //create an ArrayList to store pets
        ArrayList<Pet> petList = new ArrayList<>();
        //create an ArrayList to store Dogs
        ArrayList<Dog> dogList = new ArrayList<>();
        //create an ArrayList to store Cats
        ArrayList<Cat> catList = new ArrayList<>();
        //create a pet object
        Pet pet;

        while(true)
        {
            System.out.println("Enter pet name ('STOP' to stop)");
            String name = input.nextLine();
            //check if the user wants to stop
            if(name.equals("STOP"))
```

```
            {
                break;
            }
            System.out.println("Enter pet type ('d' for Dog and 'c' for Cat)");
            String type = input.nextLine();
            //create a new Cat object based on the type
            if(type.equals("c"))
            {
                System.out.println("Enter pet coat color");
                String coatColor = input.nextLine();
                Cat cat = new Cat();
                pet = new Cat();
                //set the name of the Cat
                pet.setName(name);
                cat.setName(name);
                //set the coat color of the Cat
                cat.setCoatColor(coatColor);
                //typecasting pet object to add coatcolor
                ((Cat)pet).setCoatColor(coatColor);
                //add Cat to the cat list
                catList.add(cat);
            }
            //create a new Dog object based on the type
            else if(type.equals("d"))
            {
                System.out.println("Enter pet weight (kg)");
                double weight = input.nextDouble();
                input.nextLine(); // Consume the newline character
                pet = new Dog();
                Dog dog = new Dog();
                //set the name of the Dog
                pet.setName(name);
                dog.setName(name);
                //set the weight of the Dog
                dog.setWeight(weight);
                //typecasting pet object to add weight
                ((Dog)pet).setWeight(weight);
                //add Dog to the dog list
                dogList.add(dog);
            }
            else
            {
                System.out.println("Invalid type.");
                continue;
            }
            //add the pet to the list
            petList.add(pet);
        }
        System.out.println("List of cats");
        //print cats
        for(Pet p : petList)
        {
            if(p instanceof Cat)
                System.out.println("Name: "+p.getName()+"\tType: Cat "+"\tCoat Color:
"+((Cat)p).getCoatColor());
        }

        System.out.println("\nList of dogs");
        //print dogs
        for(Pet p : petList)
        {
            if(p instanceof Dog)
                System.out.println("Name: "+p.getName()+"\tType: Dog "+"\tWeight:
"+((Dog)p).getWeight()+" kg");
        }
        //Display the menu choices
        System.out.println("\n\nEnter corresponding number to perform task");
        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");
        int selection = input.nextInt();

        switch(selection){
            case 1:
                //Add a new cat to the Cat array
                System.out.println("Enter the name of the cat: ");
                input.nextLine();
                String catName = input.nextLine();
                System.out.println("Enter the coat color of the cat: ");
                String coatColor = input.nextLine();
                Cat cat = new Cat();
                cat.setName(catName);
                cat.setCoatColor(coatColor);
                catList.add(cat);
                break;
            case 2:
                //add a new dog to the Dog array
                System.out.println("Enter the name of the dog: ");
                input.nextLine();
                String dogName = input.nextLine();
                System.out.println("Enter the weight of the dog: ");
                double dogWeight = input.nextDouble();
                Dog dog = new Dog();
                dog.setName(dogName);
                dog.setWeight(dogWeight);
                dogList.add(dog);
                break;
            case 3:
                //Remove a cat from the Cat array by entering the name of the cat
                System.out.println("Enter the name of the cat to remove: ");
                input.nextLine();
                String catNameToRemove = input.nextLine();
                for (int i = 0; i < catList.size(); i++) {
                    if (catList.get(i).getName().equals(catNameToRemove)) {
                        catList.remove(i);
                        break;
                    }
                }
                break;
            case 4:
                //removing a dog from the Dog array by entering the name of the dog
                System.out.println("Enter the name of the dog to remove: ");
                input.nextLine();
                String dogNameToRemove = input.nextLine();
                for (int i = 0; i < dogList.size(); i++) {
                    if (dogList.get(i).getName().equals(dogNameToRemove)) {
                        dogList.remove(i);
                        break;
                    }
                }
                break;
            case 0:
                System.out.println("Exiting...");
                break;
            default:
                //exit if the selection is invalid
                System.out.println("Invalid selection.");
                exit(0);
                break;
        }
        System.out.println("Updated list\n");
        //Print updated Dogs
        for (Dog d : dogList)
        {
            System.out.println("Dog Name: "+d.getName()+"\tWeight: "+d.getWeight());
        }
```

```
        //Print updated Cats
        for (Cat c : catList)
        {
            System.out.println("Cat Name: "+c.getName()+"\tCoat Color:
"+c.getCoatColor());
        }

    }
}
```

Output:

```
Enter pet name ('STOP' to stop)
Tommy
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet weight (kg)
43
Enter pet name ('STOP' to stop)
Kitty
Enter pet type ('d' for Dog and 'c' for Cat)
c
Enter pet coat color
White
Enter pet name ('STOP' to stop)
Scooby
Enter pet type ('d' for Dog and 'c' for Cat)
d
Enter pet weight (kg)
32
Enter pet name ('STOP' to stop)
Twinkle
Enter pet type ('d' for Dog and 'c' for Cat)
c
Enter pet coat color
Orange
Enter pet name ('STOP' to stop)
STOP
List of cats
```

```
Name: Kitty Type: Cat   Coat Color: White
Name: Twinkle   Type: Cat   Coat Color: Orange

List of dogs
Name: Tommy Type: Dog   Weight: 43.0 kg
Name: Scooby    Type: Dog   Weight: 32.0 kg


Enter corresponding number to perform task
1. Add Cat
2. Add Dog
3. Remove Cat
4. Remove Dog
0. Quit
3
Enter the name of the cat to remove:
Kitty
Updated list

Dog Name: Tommy Weight: 43.0
Dog Name: Scooby    Weight: 32.0
Cat Name: Twinkle   Coat Color: Orange

Process finished with exit code 0
```