**Klase u JAVA, nasljeđivanje i polimorfizam**

// Base class

class Animal {

public void sound() {

System.out.println("This is an animal sound");

}

}

// Derived class

class Dog extends Animal {

@Override

public void sound() {

System.out.println("The dog barks");

}

}

// Derived class

class Cat extends Animal {

@Override

public void sound() {

System.out.println("The cat meows");

}

}

public class Main {

public static void main(String[] args) {

Animal myAnimal = new Animal(); // Animal object

Animal myDog = new Dog(); // Dog object (polymorphism)

Animal myCat = new Cat(); // Cat object (polymorphism)

myAnimal.sound();

myDog.sound();

myCat.sound();

}

}

**Varijable i programske strukture u JAVA**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Input: Ask for a number

System.out.print("Enter a number: ");

int number = scanner.nextInt();

// If-else control structure

if (number % 2 == 0) {

System.out.println(number + " is an even number.");

} else {

System.out.println(number + " is an odd number.");

}

// Loop to print numbers from 1 to the entered number

System.out.println("Printing numbers from 1 to " + number + ":");

for (int i = 1; i <= number; i++) {

System.out.print(i + " ");

}

}

}

**Rad sa kolekcijama u JAVA**

import java.util.ArrayList;

public class Main {

public static void main(String[] args) {

// Create an ArrayList of strings

ArrayList<String> fruits = new ArrayList<>();

// Add elements to the list

fruits.add("Apple");

fruits.add("Banana");

fruits.add("Orange");

// Print the ArrayList

System.out.println("Fruits list: " + fruits);

// Remove an element from the list

fruits.remove("Banana");

// Loop through the ArrayList

System.out.println("Updated fruits list:");

for (String fruit : fruits) {

System.out.println(fruit);

}

}

}

```

\*\*Concepts Covered\*\*:

- \*\*Collections\*\*: Introduction to `ArrayList`.

- \*\*Basic Operations\*\*: Adding, removing, and iterating over elements.

---

These programs provide a solid foundation in OOP, control structures, and collections—essential Java concepts for moving into Spring MVC in later weeks.