1. **Develop programs to demonstrate use of switch case statement and conditional if**

Case statement

```
            import java.util.Scanner;

public class SwitchCaseExample {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Choose an option: 1, 2, or 3");
        int choice = scanner.nextInt();

        switch (choice) {
            case 1:
                System.out.println("You chose Option 1");
                break;
            case 2:
                System.out.println("You chose Option 2");
                break;
            case 3:
                System.out.println("You chose Option 3");
                break;
            default:
                System.out.println("Invalid choice");
        }
    }
}
```

Conditional if

```
            import java.util.Scanner;

public class ConditionalTernaryExample {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int number = scanner.nextInt();
        String result = (number % 2 == 0) ? "Even" : "Odd";
        System.out.println(result);
    }
}
```

2. a) Develop program for implementation of constructor

```
public class MyClass {
  private int myNumber;
  private String myString;
  public MyClass(int number, String str) {
    myNumber = number;
    myString = str;
  }
  public void display() {
    System.out.println("Number: " + myNumber);
    System.out.println("String: " + myString);
  }

  public static void main(String[] args) {
    MyClass obj = new MyClass(10, "Hello");
    obj.display();
  }
}
```

b)Develop program for implementation of multiple constructor in the class

```
       public class MyClass {
  private int myNumber;
  private String myString;
  public MyClass(int number, String str) {
    myNumber = number;
    myString = str;
  }

  public MyClass(int number) {
    this(number, "Default");
  }

  public MyClass() {
    this(0);
  }
  public void display() {
    System.out.println("Number: " + myNumber);
    System.out.println("String: " + myString);
  }
  public static void main(String[] args) {
    MyClass obj1 = new MyClass(10, "Hello");
    MyClass obj2 = new MyClass(20);
    MyClass obj3 = new MyClass();

    obj1.display();
    obj2.display();
    obj3.display();
  }
}
```

3. Develop a program for implementation of vector in java

```
import java.util.ArrayList;

public class Vector<T> extends ArrayList<T> {
  // No need to add any methods, ArrayList already provides add(), get(), remove(), and
size() methods
  public static void main(String[] args) {
    Vector<Integer> vector = new Vector<>();
    vector.add(1);
    vector.add(2);
    vector.add(3);

    System.out.println("Vector size: " + vector.size());

    System.out.println("Elements in vector:");
    for (int i = 0; i < vector.size(); i++) System.out.println(vector.get(i));

    vector.remove(1);
    System.out.println("After removing element at index 1:");
    for (int i = 0; i < vector.size(); i++) System.out.println(vector.get(i));
  }
}
```

4. Develop a program for implementation of multiple inheritance

```
interface Animal {
  void eat();
  void sleep();
}
interface Pet {
  void play();
}

class Dog implements Animal, Pet {
  public void eat() {
    System.out.println("Dog is eating.");
  }
  public void sleep() {
    System.out.println("Dog is sleeping.");
  }
  public void play() {
    System.out.println("Dog is playing.");
  }
}

public class Main {
  public static void main(String[] args) {
    Dog dog = new Dog();
    dog.eat();
    dog.sleep();
    dog.play();
  }
}
```

5. **Develop a program to import different classes in package**

File `ClassA.java`:

javaCopy code

```
package myPackage; public class ClassA { public void methodA() { System.out.println("Method A
from Class A"); } }
```

File `ClassB.java`:

javaCopy code

```
package myPackage; public class ClassB { public void methodB() { System.out.println("Method B
from Class B"); } }
```

File `ClassC.java`:

javaCopy code

```
package myPackage; public class ClassC { public void methodC() { System.out.println("Method C
from Class C"); } }
```

File `Main.java`:

javaCopy code

```
package myPackage; public class Main { public static void main(String[] args) { ClassA a = new
ClassA(); ClassB b = new ClassB(); ClassC c = new ClassC(); a.methodA(); b.methodB();
c.methodC(); } }
```

6. **Develop a program for implemantation of try, catch and finally block**

```
public class TryCatchFinallyExample {
    public static void main(String[] args) {
        try {
            int result = divide(10, 0);
            System.out.println("Result of division: " + result);
        } catch (ArithmeticException e) {
            System.out.println("An arithmetic exception occurred: " + e.getMessage());
        } finally {
            System.out.println("Finally block is executed.");
        }
    }
    public static int divide(int numerator, int denominator) {
          return numerator / denominator;
    }
}
```