1. Fizz Buzz
   For all numbers from 1 to 100, display "Fizz" if it is a multiple of 3, "Buzz" if it is a multiple of 5, "FizzBuzz"
   if it is a multiple of both 3 and 5 or the number itself if it is neither a multiple of 3 nor 5.

```java
class Main {
    public static void main(String[] args) {
        for (int i = 1; i <= 100; ++i) {
            if (i % 3 == 0 && i % 5 == 0) {
                System.out.println("FizzBuzz");
            } else if (i % 3 == 0) {
                System.out.println("Fizz");
            } else if (i % 5 == 0) {
                System.out.println("Buzz");
            } else {
                System.out.println(i);
            }
        }
    }
}
```

2. Diamond Pattern
   Input a number 'n', display a diamond pattern using "*" where the top triangle of the pattern has 'n' rows and
   the lower triangle has 'n-1' rows.

```java
import java.util.*;
class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Number: ");
        int n = sc.nextInt();
        for (int i = 1; i <= n; ++i) {
            for (int j = i; j < n; ++j) {
                System.out.print("  ");
            }
            for (int j = 1; j <= 2 * i - 1; ++j) {
                System.out.print("* ");
            }
            System.out.println();
        }
        for (int i = n - 1; i >= 1; --i) {
            for (int j = i; j < n; ++j) {
                System.out.print("  ");
            }
            for (int j = 1; j <= 2 * i - 1; ++j) {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}
```

3. Largest and Sum of Array
   Input array size 'n', input and store 'n' integers in an array, find the largest number in the array, also find the
   sum of all the integers in the array.

```java
import java.util.*;
class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
```

```
        System.out.print("Size: ");
        int n = sc.nextInt();
        int[] a = new int[n];
        System.out.println("Numbers:");
        for (int i = 0; i < a.length; ++i) {
            a[i] = sc.nextInt();
        }
        int max = a[0];
        for (int i = 1; i < a.length; ++i) {
            if (a[i] > max) {
                max = a[i];
            }
        }
        int sum = 0;
        for (int i = 0; i < a.length; ++i) {
            sum += a[i];
        }
        System.out.println("Max: " + max);
        System.out.println("Sum: " + sum);
    }
}
```

4. Account Class
   Define a class Account which has balance as instance variable and methods to deposit and withdraw amount
   from an account. Override the toString method to display account balance.
   Define a main to demo the above class.

```
class Account {
    private double balance;
    public void deposit(double amount) {
        balance += amount;
    }
    public void withdraw(double amount) {
        balance -= amount;
    }
    public String toString() {
        return "Account balance: " + balance;
    }
}
class Main {
    public static void main(String[] args) {
        Account ac = new Account();
        ac.deposit(25000);
        ac.withdraw(3000);
        System.out.println(ac);
    }
}
```

5. Student Class
   Define a class Student which has name, DSA, OOPS and DBMS as instance variables. Marks in DSA, OOPS
   and DBMS are out of 60 each. Define a constructor to initialize the instance variables. Override the toString
   method to display the name and percentage of the student. Define a main to demo the above class.

```
class Student {
    private String name;
    private int DSA, OOPS, DBMS;
    public Student(String n, int a, int b, int c) {
        name = n;
        DSA = a;
        OOPS = b;
        DBMS = c;
```

```
        }
        public String toString() {
            double percent = (DSA + OOPS + DBMS) * 100.0 / 180;
            return String.format("%s - %.1f%%", name, percent);
        }
    }
    class Main {
        public static void main(String[] args) {
            Student s = new Student("Alex", 45, 43, 48);
            System.out.println(s);
        }
    }
```

6. Inheritance
   Define a base class Publication which has title and price as instance variables, a constructor and toString
   method. Define a derived class Book which author and pages as instance variables, a constructor and toString
   method. Define a main to demo the Book class.

```
    class Publication {
        private String title;
        private int price;
        public Publication(String t, int p) {
            title = t;
            price = p;
        }
        public String toString() {
            return String.format("title: %s, price: %d", title, price);
        }
    }
    class Book extends Publication {
        private String author;
        private int pages;
        public Book(String t, int p, String a, int pg) {
            super(t, p);
            author = a;
            pages = pg;
        }
        public String toString() {
            return super.toString() +
                String.format(", author: %s, pages: %d", author, pages);
        }
    }
    class Main {
        public static void main(String[] args) {
            Book b = new Book("Halo", 255, "Sam", 125);
            System.out.println(b);
        }
    }
```

7. Interfaces
   Define an interface Shape which has a getArea method. Define a class Circle which implements the Shape
   interface. Circle has radius as instance variable and a constructor to initialize the radius. Override the toString
   method to display radius and area of circle. Define a main to create a Circle object and display it.

```
    interface Shape {
        double getArea();
    }
    class Circle implements Shape {
        private double radius;
        public Circle(double r) {
            radius = r;
```

```
        }
        public double getArea() {
            return Math.PI * Math.pow(radius, 2);
        }
        public String toString() {
            double area = getArea();
            return String.format("radius: %.1f, area: %.1f", radius, area);
        }
    }
    class Main {
        public static void main(String[] args) {
            Circle c = new Circle(5.2);
            System.out.println(c);
        }
    }
```

8. String Manipulation
   Input a string, count the number of vowels, consonants and digits in the string.

```
import java.util.*;
class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("String: ");
        String s = sc.nextLine();
        int vowels = 0, consonants = 0, digits = 0;
        for (char ch: s.toCharArray()) {
            if (Character.isLetter(ch)) {
                if ("aeiouAEIOU".contains(ch + "")) {
                    ++vowels;
                } else {
                    ++consonants;
                }
            } else if (Character.isDigit(ch)) {
                ++digits;
            }
        }
        System.out.println("Vowels: " + vowels);
        System.out.println("Consonants: " + consonants);
        System.out.println("Digits: " + digits);
    }
}
```

9. Exceptional Handling
   Input a string, if the data contained within the string is an integer then display its cube, if it is a double then
   find its square root, if it neither then display it is a string.

```
import java.util.*;
class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("String: ");
        String s = sc.nextLine();
        try {
            int n = Integer.parseInt(s);
            int c = (int)Math.pow(n, 3);
            System.out.printf("%d cube is %d\n", n, c);
        } catch (NumberFormatException e1) {
            try {
                double d = Double.parseDouble(s);
                double r = Math.sqrt(d);
```

```
                System.out.printf("%f square root is %f\n", d, r);
            } catch (NumberFormatException e2) {
                System.out.printf("%s is a string\n", s);
            }
        }
    }
}
```

10. ArrayList

Input names of 5 countries and store them in an ArrayList, sort them in ascending order of names, display the name of the first country in the list after sorting.

```
import java.util.*;
class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        ArrayList<String> list = new ArrayList<>();
        System.out.println("5 countries:");
        for (int i = 1; i <= 5; ++i) {
            String c = sc.nextLine();
            list.add(c);
        }
        Collections.sort(list);
        System.out.println("First country: " + list.get(0));
    }
}
```