1. Exercise 1 Answer:

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World!");
    }
}
```

2. Exercise 2 Answer:

```
import java.util.Scanner;

public class Calculator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter first number: ");
        double num1 = sc.nextDouble();
        System.out.print("Enter second number: ");
        double num2 = sc.nextDouble();
        System.out.print("Choose operation (+, -, *, /): ");
        char op = sc.next().charAt(0);
        double result = 0;
        switch(op) {
            case '+': result = num1 + num2; break;
            case '-': result = num1 - num2; break;
            case '*': result = num1 * num2; break;
            case '/': result = num2 != 0 ? num1 / num2 : 0; break;
            default: System.out.println("Invalid operator");
        }
        System.out.println("Result: " + result);
    }
}
```

3. Exercise 3 Answer:

```
import java.util.Scanner;

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

4. Exercise 4 Answer:

```
import java.util.Scanner;

public class LeapYear {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a year: ");
        int year = sc.nextInt();
        boolean isLeap = (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
        System.out.println(isLeap ? "Leap year" : "Not a leap year");
    }
}
```

5. Exercise 5 Answer:

```
import java.util.Scanner;

public class MultiplicationTable {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        for (int i = 1; i <= 10; i++) {
            System.out.println(num + " x " + i + " = " + (num * i));
        }
    }
}
```

6. Exercise 6 Answer:

```
public class DataTypesDemo {
    public static void main(String[] args) {
        int a = 10;
        float b = 5.5f;
        double c = 10.55;
        char d = 'A';
        boolean e = true;

        System.out.println("int: " + a);
        System.out.println("float: " + b);
        System.out.println("double: " + c);
        System.out.println("char: " + d);
        System.out.println("boolean: " + e);
    }
}
```

7. Exercise 7 Answer:

```
public class TypeCasting {
    public static void main(String[] args) {
```

```
        double d = 9.78;
        int i = (int) d;
        int j = 5;
        double e = j;

        System.out.println("Double to Int: " + i);
        System.out.println("Int to Double: " + e);
    }
}
```

8. Exercise 8 Answer:

```
public class OperatorPrecedence {
    public static void main(String[] args) {
        int result = 10 + 5 * 2;
        System.out.println("Result: " + result); // 5*2 = 10 + 10 = 20
    }
}
```

9. Exercise 9 Answer:

```
import java.util.Scanner;

public class GradeCalculator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter marks (0-100): ");
        int marks = sc.nextInt();
        if (marks >= 90) System.out.println("Grade A");
        else if (marks >= 80) System.out.println("Grade B");
        else if (marks >= 70) System.out.println("Grade C");
        else if (marks >= 60) System.out.println("Grade D");
        else System.out.println("Grade F");
    }
}
```

10. Exercise 10 Answer:

```
import java.util.Scanner;
import java.util.Random;

public class NumberGuess {
    public static void main(String[] args) {
        Random rand = new Random();
        int number = rand.nextInt(100) + 1;
        Scanner sc = new Scanner(System.in);
        int guess = 0;
        while (guess != number) {
            System.out.print("Guess the number (1-100): ");
            guess = sc.nextInt();
            if (guess < number) System.out.println("Too low!");
```

```
            else if (guess > number) System.out.println("Too high!");
        }
        System.out.println("Correct!");
    }
}
```

11. Exercise 11 Answer:

```
import java.util.Scanner;

public class FactorialCalculator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a non-negative integer: ");
        int num = sc.nextInt();
        long factorial = 1;
        for (int i = 1; i <= num; i++) {
            factorial *= i;
        }
        System.out.println("Factorial: " + factorial);
    }
}
```

12. Exercise 12 Answer:

```
public class Overloading {
    static int add(int a, int b) {
        return a + b;
    }
    static double add(double a, double b) {
        return a + b;
    }
    static int add(int a, int b, int c) {
        return a + b + c;
    }

    public static void main(String[] args) {
        System.out.println(add(2, 3));
        System.out.println(add(2.5, 3.5));
        System.out.println(add(1, 2, 3));
    }
}
```

13. Exercise 13 Answer:

```
import java.util.Scanner;

public class RecursiveFibonacci {
    static int fibonacci(int n) {
        if (n <= 1) return n;
        return fibonacci(n - 1) + fibonacci(n - 2);
```

```
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter n: ");
        int n = sc.nextInt();
        System.out.println("Fibonacci number: " + fibonacci(n));
    }
}
```

14. Exercise 14 Answer:

```
import java.util.Scanner;

public class ArraySumAverage {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter number of elements: ");
        int n = sc.nextInt();
        int[] arr = new int[n];
        int sum = 0;
        for (int i = 0; i < n; i++) {
            System.out.print("Enter element " + (i + 1) + ": ");
            arr[i] = sc.nextInt();
            sum += arr[i];
        }
        double avg = (double) sum / n;
        System.out.println("Sum: " + sum + ", Average: " + avg);
    }
}
```

15. Exercise 15 Answer:

```
import java.util.Scanner;

public class StringReversal {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String input = sc.nextLine();
        String reversed = new StringBuilder(input).reverse().toString();
        System.out.println("Reversed string: " + reversed);
    }
}
```

16. Exercise 16 Answer:

```
import java.util.Scanner;

public class PalindromeChecker {
    public static void main(String[] args) {
```

```
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String input = sc.nextLine().replaceAll("[^a-zA-Z0-9]", "").toLowerCase();
        String reversed = new StringBuilder(input).reverse().toString();
        System.out.println(input.equals(reversed) ? "Palindrome" : "Not a palindrome");
    }
}
```

17. Exercise 17 Answer:

```
class Car {
    String make, model;
    int year;

    void displayDetails() {
        System.out.println("Make: " + make + ", Model: " + model + ", Year: " + year);
    }
}

public class CarTest {
    public static void main(String[] args) {
        Car c = new Car();
        c.make = "Toyota";
        c.model = "Corolla";
        c.year = 2020;
        c.displayDetails();
    }
}
```

18. Exercise 18 Answer:

```
class Animal {
    void makeSound() {
        System.out.println("Animal sound");
    }
}

class Dog extends Animal {
    void makeSound() {
        System.out.println("Bark");
    }
}

public class InheritanceExample {
    public static void main(String[] args) {
        Animal a = new Animal();
        Dog d = new Dog();
        a.makeSound();
        d.makeSound();
    }
}
```

19. Exercise 19 Answer:

```
interface Playable {
    void play();
}

class Guitar implements Playable {
    public void play() {
        System.out.println("Playing Guitar");
    }
}

class Piano implements Playable {
    public void play() {
        System.out.println("Playing Piano");
    }
}

public class InterfaceDemo {
    public static void main(String[] args) {
        Playable g = new Guitar();
        Playable p = new Piano();
        g.play();
        p.play();
    }
}
```

20. Exercise 20 Answer:

```
import java.util.Scanner;

public class TryCatchExample {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        try {
            System.out.print("Enter first number: ");
            int a = sc.nextInt();
            System.out.print("Enter second number: ");
            int b = sc.nextInt();
            System.out.println("Result: " + (a / b));
        } catch (ArithmeticException e) {
            System.out.println("Cannot divide by zero.");
        }
    }
}
```

21. Exercise 21 Answer:

```
class InvalidAgeException extends Exception {
    public InvalidAgeException(String message) {
        super(message);
    }
```

```
}

public class CustomException {
    public static void main(String[] args) {
        int age = 16;
        try {
            if (age < 18) {
                throw new InvalidAgeException("Age must be 18 or above.");
            }
            System.out.println("Valid age");
        } catch (InvalidAgeException e) {
            System.out.println("Exception: " + e.getMessage());
        }
    }
}
```

22. Exercise 22 Answer:

```
import java.io.FileWriter;
import java.util.Scanner;

public class FileWrite {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter text to write to file: ");
        String text = sc.nextLine();
        try {
            FileWriter writer = new FileWriter("output.txt");
            writer.write(text);
            writer.close();
            System.out.println("Data written to output.txt");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

23. Exercise 23 Answer:

```
import java.io.BufferedReader;
import java.io.FileReader;

public class FileRead {
    public static void main(String[] args) {
        try {
            BufferedReader reader = new BufferedReader(new FileReader("output.txt"));
            String line;
            while ((line = reader.readLine()) != null) {
                System.out.println(line);
            }
            reader.close();
        } catch (Exception e) {
```

```
            e.printStackTrace();
        }
    }
}
```

24. Exercise 24 Answer:

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

public class StudentList {
    public static void main(String[] args) {
        ArrayList<String> students = new ArrayList<>();
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter student names (type 'exit' to stop):");
        while (true) {
            String name = sc.nextLine();
            if (name.equalsIgnoreCase("exit")) break;
            students.add(name);
        }
        System.out.println("Student names: " + students);
    }
}
```

25. Exercise 25 Answer:

```
import java.util.HashMap;
import java.util.Scanner;

public class StudentMap {
    public static void main(String[] args) {
        HashMap<Integer, String> map = new HashMap<>();
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter student ID and name (type -1 to stop):");
        while (true) {
            System.out.print("ID: ");
            int id = sc.nextInt();
            if (id == -1) break;
            sc.nextLine();
            System.out.print("Name: ");
            String name = sc.nextLine();
            map.put(id, name);
        }
        System.out.print("Enter ID to search: ");
        int searchId = sc.nextInt();
        System.out.println("Name: " + map.getOrDefault(searchId, "Not found"));
    }
}
```