Level 2 Practice Programs

1. Create a program to print odd and even numbers between 1 to the number entered by the user.

   Hint =>

   a. Get an integer input from the user, assign to a variable number and check for Natural Number
   b. Using a for loop, iterate from 1 to the number
   c. In each iteration of the loop, print the number is odd or even number

```
import java.util.Scanner;


public class Q1_OddEven {


   public static void printOddEven(int number) {
       if (number <= 0) {
           System.out.println("Invalid input. Please enter a positive
integer.");
           return; // Exit early if input is invalid
       }


       System.out.println("Odd numbers:");
       for (int i = 1; i <= number; i += 2) {
           System.out.print(i + " ");
       }
       System.out.println(); // New line for better formatting


       System.out.println("Even numbers:");
```

```
        for (int i = 2; i <= number; i += 2) {
            System.out.print(i + " ");
        }
        System.out.println(); // New line for better formatting
    }



    public static void printOddEvenSingleLoop(int number) {
         if (number <= 0) {
            System.out.println("Invalid input. Please enter a positive
integer.");
            return;
        }

         System.out.println("Numbers from 1 to " + number + ":");
         for (int i = 1; i <= number; i++) {
             if (i % 2 == 0) {
                 System.out.println(i + " is even");
             } else {
                 System.out.println(i + " is odd");
             }
         }
    }

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int num;

        while (true) {  // Input loop for validation
```

```
            try {
                System.out.print("Enter a positive integer: ");
                num = input.nextInt();
                if (num > 0) {
                    break; // Exit the loop if input is valid
                } else {
                    System.out.println("Please enter a number greater than
zero.");
                }
            } catch (java.util.InputMismatchException e) {
                System.out.println("Invalid input. Please enter an
integer.");
                input.next(); // Clear the invalid input from the scanner
            }
        }

        printOddEven(num);
        System.out.println("--- Single Loop Version ---"); // Separator
        printOddEvenSingleLoop(num);
        input.close(); // Close the scanner to prevent resource leaks
    }
}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q1_OddEven
Enter a positive integer: 10
Odd numbers:
1 3 5 7 9
Even numbers:
2 4 6 8 10
--- Single Loop Version ---
Numbers from 1 to 10:
1 is odd
2 is even
3 is odd
4 is even
5 is odd
6 is even
7 is odd
8 is even
9 is odd
10 is even
```

2. Create a program to find the bonus of employees based on their years of service.

   Hint =>

   a. Zara decided to give a bonus of 5% to employees whose year of service is more than 5 years.
   b. Take salary and year of service in the year as input.
   c. Print the bonus amount.

```java
import java.util.Scanner;


public class Q2_EmployeeBonus {


   public static void main(String[] args) {
        Scanner input = new Scanner(System.in);


        System.out.print("Enter employee salary: ");
        double salary = input.nextDouble();


        System.out.print("Enter years of service: ");
        int yearsOfService = input.nextInt();
```

```
        if (yearsOfService > 5) {
            double bonusPercentage = 0.05; // 5% bonus
            double bonusAmount = salary * bonusPercentage;
            System.out.println("Bonus amount: $" + bonusAmount);
        } else {
            System.out.println("Employee is not eligible for a bonus.");
        }

        input.close(); // Close the scanner to prevent resource leaks
    }
}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q2_EmployeeBonus
Enter employee salary: 15000
Enter years of service: 4
Employee is not eligible for a bonus.
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q2_EmployeeBonus
Enter employee salary: 15000
Enter years of service: 10
Bonus amount: $750.0
```

3. Create a program to find the multiplication table of a number entered by the user from 6 to 9.

   Hint =>

   a. Take integer input and store it in the variable number
   b. Using a for loop, find the multiplication table of number from 6 to 9 and print it in the format number * i = ___

```
import java.util.Scanner;

public class Q3_MultiplicationTable {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

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

```
        int number = input.nextInt();


        System.out.println("Multiplication table of " + number + " from 6
to 9:");


        for (int i = 6; i <= 9; i++) {
            int result = number * i;
            System.out.println(number + " * " + i + " = " + result);
        }


        input.close();
    }
}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q3_MultiplicationTable
Enter a number: 7
Multiplication table of 7 from 6 to 9:
7 * 6 = 42
7 * 7 = 49
7 * 8 = 56
7 * 9 = 63
```

4. Write a program FizzBuzz, take a number as user input, and if it is a positive integer loop from 0 to the number and print the number, but for multiples of 3 print "Fizz" instead of the number, for multiples of 5 print "Buzz", and for multiples of both print "FizzBuzz".

   Hint =>

   a. Write the program and use ***for*** loop

```
import java.util.Scanner;


public class Q4_FizzBuzz {


    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
```

```
        System.out.print("Enter a positive integer: ");
        if (!input.hasNextInt()) { // Check if the next input is an integer
            System.out.println("Invalid input. Please enter an integer.");
            input.close();
            return;
        }

        int number = input.nextInt();

        if (number <= 0) {
            System.out.println("Please enter a positive integer.");
            input.close();
            return;
        }

        for (int i = 0; i <= number; i++) {
            String output = ""; // Initialize an empty string for each
number

            if (i % 3 == 0) {
                output += "Fizz"; // Append "Fizz" if divisible by 3
            }
            if (i % 5 == 0) {
                output += "Buzz"; // Append "Buzz" if divisible by 5
            }

            if (output.isEmpty()) { // If the string is still empty, print
the number
```

```
                System.out.println(i);

            } else {

                System.out.println(output); // Otherwise, print "Fizz",
"Buzz", or "FizzBuzz"

            }

        }


        input.close();

    }

}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q4_FizzBuzz
Enter a positive integer: 13
FizzBuzz
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
```

5. Rewrite the program 5 FizzBuzz using while loop

```
import java.util.Scanner;


public class Q5_FizzBuzz {


   public static void main(String[] args) {

        Scanner input = new Scanner(System.in);


        System.out.print("Enter a positive integer: ");

        if (!input.hasNextInt()) {

            System.out.println("Invalid input. Please enter an integer.");
```

```
            input.close();
            return;
        }

        int number = input.nextInt();

        if (number <= 0) {
            System.out.println("Please enter a positive integer.");
            input.close();
            return;
        }

        int i = 0; // Initialize the loop counter

        while (i <= number) { // Use a while loop
            String output = "";

            if (i % 3 == 0) {
                output += "Fizz";
            }
            if (i % 5 == 0) {
                output += "Buzz";
            }

            if (output.isEmpty()) {
                System.out.println(i);
            } else {
                System.out.println(output);
```

```
            }


            i++; // Increment the loop counter

        }


        input.close();

    }

}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q5_FizzBuzz
Enter a positive integer: 9
FizzBuzz
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
```

6. Create a program to find the youngest friends among 3 Amar, Akbar, and Anthony based on their ages and the tallest among the friends based on their heights

   Hint =>

   a. Take user input for the age and height of the 3 friends and store it in a variable
   b. Find the smallest of the 3 ages to find the youngest friend and display it
   c. Find the largest of the 3 heights to find the tallest friend and display it

```
import java.util.Scanner;


public class Q6_YoungestAmongThree {


    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);


        // Input for Amar

        System.out.print("Enter Amar's age: ");
```

```
        int amarAge = input.nextInt();
        System.out.print("Enter Amar's height (in cm): ");
        double amarHeight = input.nextDouble();

        // Input for Akbar
        System.out.print("Enter Akbar's age: ");
        int akbarAge = input.nextInt();
        System.out.print("Enter Akbar's height (in cm): ");
        double akbarHeight = input.nextDouble();

        // Input for Anthony
        System.out.print("Enter Anthony's age: ");
        int anthonyAge = input.nextInt();
        System.out.print("Enter Anthony's height (in cm): ");
        double anthonyHeight = input.nextDouble();

        // Find the youngest friend
        String youngestFriend;
        int youngestAge = Math.min(amarAge, Math.min(akbarAge,
anthonyAge)); // Efficiently find the minimum

        if (youngestAge == amarAge) {
            youngestFriend = "Amar";
        } else if (youngestAge == akbarAge) {
            youngestFriend = "Akbar";
        } else {
            youngestFriend = "Anthony";
        }
```

```
        System.out.println("The youngest friend is " + youngestFriend + "
with age " + youngestAge + ".");



        // Find the tallest friend
        String tallestFriend;
        double tallestHeight = Math.max(amarHeight, Math.max(akbarHeight,
anthonyHeight));


        if (tallestHeight == amarHeight) {
            tallestFriend = "Amar";
        } else if (tallestHeight == akbarHeight) {
            tallestFriend = "Akbar";
        } else {
            tallestFriend = "Anthony";
        }


        System.out.println("The tallest friend is " + tallestFriend + "
with height " + tallestHeight + " cm.");


        input.close();
    }
}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q6_YoungestAmongThree
Enter Amar's age: 20
Enter Amar's height (in cm): 170
Enter Akbar's age: 18
Enter Akbar's height (in cm): 174
Enter Anthony's age: 19
Enter Anthony's height (in cm): 165
The youngest friend is Akbar with age 18.
The tallest friend is Akbar with height 174.0 cm.
```

7. Create a program to find the factors of a number taken as user input.

Hint =>

a. Get input value for a variable named number.
b. Run a ***for*** loop from i = 1 to i < number. In each iteration of the loop, check if the number is perfectly divisible by i. If true, print the value of i.

```
import java.util.Scanner;


public class Q7_Factors {


   public static void main(String[] args) {
        Scanner input = new Scanner(System.in);


        System.out.print("Enter a positive integer: ");


        if (!input.hasNextInt()) {
            System.out.println("Invalid input. Please enter an integer.");
            input.close();
            return;
        }


        int number = input.nextInt();


        if (number <= 0) {
            System.out.println("Please enter a positive integer.");
            input.close();
            return;
        }


        System.out.print("Factors of " + number + " are: ");
```

```
        for (int i = 1; i < number; i++) {
            if (number % i == 0) {
                System.out.print(i + " ");
            }
        }
        System.out.println(); // New line for better formatting

        input.close();
    }
}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q7_Factors
Enter a positive integer: 15
Factors of 15 are: 1 3 5
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q7_Factors
Enter a positive integer: 13
Factors of 13 are: 1
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$
```

8. Create a program to print the greatest factor of a number beside itself using a loop.

   Hint =>

   a. Get an integer input and assign it to the number variable. As well as define a greatestFactor variable and assign it to 1
   b. Create a ***for*** loop that runs from last but one till 1 as in i = number - 1 to i = 1.
   c. Inside the loop, check if the number is perfectly divisible by i then assign i to greatestFactor variable and break the loop.
   d. Display the greatestFactor variable outside the loop

```
import java.util.Scanner;

public class Q8_GreatestFactor {

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

```
        Scanner input = new Scanner(System.in);

        System.out.print("Enter a positive integer: ");

        if (!input.hasNextInt()) {
            System.out.println("Invalid input. Please enter an integer.");
            input.close();
            return;
        }

        int number = input.nextInt();

        if (number <= 1) { // 1 and numbers less than 1 don't have factors other than 1 and themselves
            System.out.println("Numbers greater than 1 have factors besides 1 and themselves.");
            input.close();
            return;
        }

        int greatestFactor = 1; // Initialize to 1 (1 is always a factor)

        for (int i = number - 1; i >= 1; i--) { // Loop from number-1 down to 1
            if (number % i == 0) {
                greatestFactor = i;
                break; // Exit the loop as soon as the greatest factor is found
            }
```

```
        }


        System.out.println("The greatest factor of " + number + " (besides
itself) is: " + greatestFactor);


        input.close();

    }

}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q8_GreatestFactor
Enter a positive integer: 18
The greatest factor of 18 (besides itself) is: 9
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q8_GreatestFactor
Enter a positive integer: 146
The greatest factor of 146 (besides itself) is: 73
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$
```

9. Create a program to find the power of a number.

   Hint =>

   a. Get integer input for two variables named number and power.
   b. Create a result variable with an initial value of 1.
   c. Run a for loop from i = 1 to i <= power.
   d. In each iteration of the loop, multiply the result with the number and assign the value to the result.
   e. Finally, print the result

```
import java.util.Scanner;


public class Q9_PowerOfNumber {


    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);


        System.out.print("Enter the base number: ");
```

```
        int number = input.nextInt();


        System.out.print("Enter the power: ");
        int power = input.nextInt();


        int result = 1;


        for (int i = 1; i <= power; i++) {
            result *= number;
        }


        System.out.println(number + " raised to the power of " + power + "
is: " + result);


        input.close();
    }
}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q9_PowerOfNumber
Enter the base number: 3
Enter the power: 6
3 raised to the power of 6 is: 729
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$
```

10. Create a program to find all the multiple of a number taken as user input below 100.

    Hint =>

    a. Get input value for a variable named number.
    b. Run a ***for*** loop backwards: from i = 100 to i = 1.
    c. Inside the loop, check if i perfectly divides number.
    d. If true, print the number and ***continue*** the loop.

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

```java
public class Q10_MultiplesBelow100 {

   public static void main(String[] args) {
       Scanner input = new Scanner(System.in);

       System.out.print("Enter a positive integer: ");

       if (!input.hasNextInt()) {
           System.out.println("Invalid input. Please enter an integer.");
           input.close();
           return;
       }

       int number = input.nextInt();

       if (number <= 0) {
           System.out.println("Please enter a positive integer.");
           input.close();
           return;
       }

       System.out.println("Multiples of " + number + " below 100 are:");

       for (int i = 1; i < 100; i++) {
           if (i % number == 0) {
               System.out.print(i + " ");
           }
```

```
        }

        System.out.println(); // New line for better formatting



        input.close();

    }

}
```

```
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$ java Q10_MultiplesBelow100
Enter a positive integer: 8
Multiples of 8 below 100 are:
8 16 24 32 40 48 56 64 72 80 88 96
garv-rahut@garvrahut:~/Bootcamp 2.0/ProgrammingControlFlows/Week 2 lvl 2_10$
```