**********Java Language Basics - Practice Exercise 2**********

---

1. Write a Java method to Reverse the given input & Check if it is a Palindrome.

---

2. Write a Java method to check if a given number is power of 4

---

3. Create a class named Member with Name, age, Salary as its variable, write an other class named Member Variable that creates an instance of the Member class, initialises its member variables, and then displays the value of each member variable.

Output: Members Name: Harry
Potter Members Age: 30
Members Salary: 2500.3

---

4. Write a boolean method called isEven() in a class called EvenNumTest, which takes an int as input and returns true if the input is even. The signature of the method is as follows: public static boolean isEven(int number)

---

5. Write a program, which reads number of students and n grades as input (of int between 0 and 100, inclusive) and displays the average, minimum and maximum.Your program shall check for valid input. You should keep all the grades in an int[] and use a method for each of the computations.

Output:

Enter the number of students: 4

Enter the grade for student 1: 86

Enter the grade for student 2: 65

Enter the grade for student 3: 98

Enter the grade for student 4: 77
The average is 81.50

The minimum is 65

The maximum is 98

---

6. Write a program to list all the factorials, that can be expressed as an int (i.e., 32-bit signed integer). Your output shall look like:

Int Factorials:

The factorial of 1 is 1

The factorial of 2 is 2

.....

The factorial of 12 is 479001600

he factorial of 13 is out of range

Modify your program and add a method called longFactorial to list all the factorial that can be expressed as a long (64-bit signed integer).The maximum value for long is kept in a constant called Long.MAX_VALUE.

your output shall look like:

Long Factorials:

The factorial of 1 is is 1.

The factorial of 2 is is 2.

.....

The factorial of 20 is is 2432902008176640000.

The factorial of 21 is out of range.

---

7. Write a class named User. The class needs four fields (instance variables) with the names firstName, lastName age, and salary.

Complete the class ensuring encapsulation and selection of proper datatypes.

Ensure age cannot be less than 0 or greater than 100.

Add a isValidAge method to check that employee age is between 18 to 60.

Add a getFullName to return the full name of the person.

NOTE: No methods should be defined static.

NOTE: Do not add any main method to the solution code.