Collections Assignment

Java based on the java day three contents covered: Implementation of Reactive programming, encapsulation Collections and Stream, filter, method reference, count, also use Enum and wrapper classes.

Problem Statement: Below are the addition and modification need to be done for the previous day assignment.

1. Add the below conversion functionalities to the existing calculator in reactive programming paradigms. Take (input) a value based on the event -> conversion click (at this stage another input) based on the event convert as below.
2. Decimal to binary
3. Decimal to octal
4. Decimal to hexadecimal
5. Binary to decimal
6. Octal to decimal
7. Hexadecimal to decimal

Handle Invalid\_Number\_System user defined exception if the input value is not in the given number system

1. Modify the history storage created earlier and the store the history data along with date and time, expression, result, type of expression create a bean class [POJO] of history.
2. Using Collection and stream do the following
   1. Display the history date wise.
   2. Display the history arithmetic expression wise, Date Expression wise, and conversion operation wise.
   3. Prepare a summary report which shows operator wise summary.

Solution:

Step 1: Create new java project 3 and copy the src folder of the previous project.

Step 2: Create user defined exception as done earlier.

Step 3: Create History class inside the package bean [POJO] as shown in demonstration with private data member, setter, getter, toString, and constructor.

Step 4: Create an enum ConversionOp to store event constant dtobin (decimal to binary) dtooct …..

Step 5: Create class Conversion\_Evaluatro which extends Evaluator abstract class and override evaluate method as shown to do conversion. Use wrapper class for conversion.

Step 6: modify the evaluator abstract class as shown to do sort, summary, display the historic data using functional programming.

Step 7: Modify the Factory method to return object of the Conversion\_Evaluatro

Step 8: Create main and check the conversion