A08

Due Date: Wednesday, March 29
File(s) to be submitted: MonthlyPrecipitation.java
Sample output

Monthly Precipitation (Arrays)

Summary

In this assignment, we keep track of total precipitation and rainfall (in mm) for all days of different months.

Create a datatype class (**MonthlyPrecipitation**). Include all appropriate instance variables and constants. Your class needs to work with the driver program provided.

Note: you will not be able to run the driver program until you have created a **MonthlyPrecipitation** class with every method defined. (Remember to start with *stubs* of all methods, because it'd be a real pain to have to write all that code before you could test any of it!)

Details

You have been provided with a driver program (**PrecipitationDriver**.java) that tests a **MonthlyPrecipitation** class. Your task is to write that class. DO NOT CHANGE/SUBMIT PrecipitationDriver.java.

The **MonthlyPrecipitation** class has two instance variables: `monthName` (a String) and `pctVals` (a double array to store the precipitation amount for all days of a given month). You also need to declare a constant called `DEF_NUM_DAYS` representing the array's length which is set to 31.

The class has two constructors:

- *Primary* constructor takes initial values for month name and the double array of requested values for precipitation. Note that when the passed array parameter has less than `DEF_NUM_DAYS` elements, the rest of the elements of `pctVals` will be set to `Double.NaN` (i.e., it means Not a Number). For example, for February, the passed array parameter will probably have only 28 elements which will be used to set the first 28 elements of `pctVals` and the last 3 elements will be set to `Double.NaN`.
- A *secondary* constructor takes only an initial value for `monthName`, and `pctVals` will be initialized to `Double.NaN` for all its elements.

Note: Remember that the secondary constructor is supposed to call the primary one! You can pass an empty array to the primary constructor by using `new double[0]` as the second argument.

The class has the following methods:

- **getMonthName()** -- a getter for the `monthName` property.
- **setMonthName(String name)** -- a setter for the `monthName` property.
- **getPctVal(**int dayNum**)** -- a getter for one element of `pctVals` array.
- **getPctVals()** -- a getter for the `pctVals` array.
- **setPctVal(**int dayNum, double value**)** -- a setter for one element of `pctVals` array.
- **getMonthlyAverage()** -- returns the average of `pctVals` elements that have been set to a number (not `Double.NaN`).
- **getWettestDayNumber()** -- returns the *day number* with the highest amount of precipitation among those days that have been set to a number.
- **toString()** -- a method to create a String suitable for printing. The String contains information about:
  - The month
  - The average monthly precipitation
  - The day number with the highest amount of precipitation
  - The precipitation for different days (if it's not NaN)

```
Month: February
Average precipitation: 2.5714285714285716
The wettest day was day#20
Precipitation amount per day:
     1: 0.0
     2: 0.0
     3: 2.8
     4: 0.0
     5: 0.9
     6: 0.0
     7: 0.0
     8: 0.0
     9: 0.0
     10: 10.7
     11: 0.0
     12: 0.0
     13: 5.6
     14: 0.9
     15: 3.2
     16: 0.3
     17: 8.7
```

```
     18: 1.1
     19: 0.0
     20: 33.3
     21: 0.0
     22: 0.0
     23: 0.0
     24: 4.5
     25: 0.0
     26: 0.0
     27: 0.0
     28: 0.0
```

For more details, see the sample output.

Note: In getPctVal(int dayNum) method, a) if the requested `dayNum` is not between 1 and `DEF_NUM_DAYS`, e.g., `dayNum = 32`, print the following error message:

```
ERROR: Invalid day number 32
```

and b) if no number has been set for the corresponding element of `pctVals`, e.g., `pctVals[28] = Double.NaN`, we will print something like:

```
No defined precipitation amount for day#29
```

In both cases a) and b), return `Double.NaN` after printing the message.

Similarly, in setPctVal(int dayNum, double value) method, if the requested `dayNum` is not between 1 and `DEF_NUM_DAYS`, e.g., `dayNum = 32`, print the following error message:

```
ERROR: Invalid day number 32
```

Hint: Utilize the `Double.isNaN` method to check whether a given element of `pctVals` is set to a number. The method has a `double` parameter and returns `True` if the argument is `NaN` (not a number) and `False` otherwise.

Grading Outline

- 60% -- Methods perform as required
- 20% -- Methods show good design
- 20% -- Submitted material meets the standard requirements.

Sample output

```
MonthlyPrecipitation Names:
December, Janary, February, march

...press enter...
```

```
MonthlyPrecipitation Names:
December, January, February, March

...press enter...

Precipitation for some random days:
Dec 2nd: 0.0
Dec 31st: 6.4
Jan 20th: 8.8
No defined precipitation amount for day#26
Jan 26th: NaN
Feb 17th: 8.7
No defined precipitation amount for day#30
Feb 30th: NaN
No defined precipitation amount for day#10
March 10th: NaN
ERROR: Invalid day number -10
March -10th: NaN
ERROR: Invalid day number 32
March 32th: NaN

...press enter...
ERROR: Invalid day number -2
ERROR: Invalid day number 32

...press enter...

The monthly average precipitation:
December: 4.612903225806452 mm
January: 7.466666666666667 mm
February: 2.5714285714285716 mm
March: 2.66 mm

...press enter...

Finding the wettest day in each month:
December 8
January 23
Februray 20
March 15

...press enter...

Displaying info:
================================================
Month: December
```

```
Average precipitation: 4.612903225806452
The wettest day was day#8
Precipitation amount per day:
       1: 20.9
       2: 0.0
       3: 11.9
       4: 5.1
       5: 0.0
       6: 0.0
       7: 2.7
       8: 30.5
       9: 11.5
       10: 0.0
       11: 0.0
       12: 0.0
       13: 0.0
       14: 0.0
       15: 0.0
       16: 0.2
       17: 28.9
       18: 1.4
       19: 1.7
       20: 0.4
       21: 0.0
       22: 0.0
       23: 21.2
       24: 0.2
       25: 0.0
       26: 0.0
       27: 0.0
       28: 0.0
       29: 0.0
       30: 0.0
       31: 6.4

==================================================
Month: January
Average precipitation: 7.466666666666667
The wettest day was day#23
Precipitation amount per day:
       1: 14.2
       2: 0.0
       3: 0.4
       4: 3.6
       5: 3.4
       6: 1.1
       7: 3.5
```

```
        8: 0.0
        9: 2.7
        10: 2.5
        11: 0.2
        12: 0.0
        13: 13.6
        14: 21.8
        15: 20.9
        16: 24.7
        17: 1.1
        18: 0.2
        19: 0.2
        20: 8.8
        21: 2.9
        22: 0.6
        23: 50.3
        24: 0.3
        26: 32.6
        27: 0.2
        28: 0.0
        29: 4.0
        30: 3.3
        31: 6.9

================================================
Month: February
Average precipitation: 2.5714285714285716
The wettest day was day#20
Precipitation amount per day:
        1: 0.0
        2: 0.0
        3: 2.8
        4: 0.0
        5: 0.9
        6: 0.0
        7: 0.0
        8: 0.0
        9: 0.0
        10: 10.7
        11: 0.0
        12: 0.0
        13: 5.6
        14: 0.9
        15: 3.2
        16: 0.3
        17: 8.7
        18: 1.1
```

```
        19: 0.0
        20: 33.3
        21: 0.0
        22: 0.0
        23: 0.0
        24: 4.5
        25: 0.0
        26: 0.0
        27: 0.0
        28: 0.0

==================================================
Month: March
Average precipitation: 2.66
The wettest day was day#15
Precipitation amount per day:
        1: 1.3
        2: 9.3
        3: 2.8
        4: 0.0
        5: 0.0
        6: 0.7
        8: 0.0
        9: 0.5
        10: 0.0
        11: 0.0
        12: 0.0
        13: 0.0
        14: 15.4
        15: 18.8
        16: 2.4
        17: 0.0
        18: 1.6
        19: 0.2
        20: 0.0
        21: 0.2
```