Java Objects

Assignment

Write a program that checks to see if a credit card number is valid.

Constraints

- The credit card number (CC#) must be 16 digits in length
- The CC# can be in either form: ####-####-####-#### or ################
- The Expiration Date (Exp) must be in the form: MM/YY (Example: 12/15)
- Notify user of correct entry form for CC# and Exp
- Name, CC#, and Exp must be entered as **Strings**
- Use a separate (external to the main class) subclass, **VerifyCard()** to validate the CC# with the following **private** method conditions:
    - *Condition* 1: The first digit must be a 4
    - *Condition* 2: The fourth digit must be one greater than the fifth digit
    - *Condition* 3: The product of the first, fifth, and ninth digits must be 24
    - *Condition* 4: The sum of all digits must be evenly divisible by 4
    - *Condition* 5: The sum of the four digits in the expiration date must be less than the product of the last two digits of the card number
    - *Condition* 6: The sum of the first four digits must be one less than the sum of the last four digits
    - *Condition* 7: If you treat the first two digits as a two-digit number, and the 15th and 16th digits as a two digit number, their sum must be 100 (Example: 4643262531465454 -> 46 + 54 = 100)

Requirements

- Ask the user for the name on the credit card
- Ask the user for the credit card number
- Ask the user for the expiration date
- Use set and get methods in the subclass for the information entered by the user
- Set methods must be private
- All *Condition* listed in **Constraints** should be **private** methods
- Use a **regex** String for *Condition* 1
- Most Condition methods will use an **if** statement
- Condition methods 4-6 will need one to two for loops to step through the int arrays
- This program will utilize code learned from Week 1 through Week 10

Hints

- CC# and Exp can be converted into int Arrays
- The following CC#s can be used as test cases. Each of these numbers will return as valid. Make sure you change them around or enter other CC#s that are not valid:
    - 4192112566331259
    - 4283253533222358
    - 4374398522116157
    - 4292154566732358
    - 4553223534333555
    - 4643262531465454
    - 4732169566119053
    - 4823287533234752

- Each of the CC#s will work with an Exp of 12/15. By increasing the YY, you may find that a few of these cards will not be valid
- Consider using the **.replaceAll()** method to remove slashes and/or dashes from entered Strings
- Consider using the **.split()** method when converting the Strings to int Arrays
- The **.parseInt()** method and the **Integer** class can be your friend in this assignment

Expected Output

The sample output below has executed three runs of the program. User input is in **red**.

```
Enter Card Holder's Name: Tom Jones
Enter Credit Card Number (No dashes): 4553223534333555
Enter Credit Card Expiration Date (MM/YY): 12/15

Card number: 4553223534333555 is a valid credit card.

Run program again? (Y for Yes, N for No): y
Enter Card Holder's Name: Dean Martin
Enter Credit Card Number (No dashes): 4732169566119053
Enter Credit Card Expiration Date (MM/YY): 12/15

Card number: 4732169566119053 is a valid credit card.

Run program again? (Y for Yes, N for No): Y
Enter Card Holder's Name: Frank Sinatra
Enter Credit Card Number (No dashes): 4234253533211358
Enter Credit Card Expiration Date (MM/YY): 12/15

Card number: 4234253533211358 is NOT a valid credit card.

Run program again? (Y for Yes, N for No): n
Thank you. Goodbye!
```

Deliverables

Please zip your program and submit the zip file by the due date listed in the requirements