ASSIGNMENT EIGHT
DUE DATE: TUESDAY, MAR. 11TH @11:58 PM

The purpose of this assignment is to assess your understanding of

- User-defined classes
- Class methods
- Class variables
- Instance variables
- Exceptions

Notes:

- Assignment extensions will **NOT** be given and late assignments will **NOT** be accepted. To earn partial credit for your assignment you must submit the work you completed by the deadline. The submitted program should execute without syntax or run time errors.
- You are encouraged to use computational thinking to solve the problems. These are straightforward solutions, but developing a good habit of analyzing the problem and describing the steps will serve you well as the problems get more complex.

Problem Requirements and Descriptions (55 pts.)

Many cities have implemented an electronic fare payment system for the city's public transportation system. A rider uses a Transit Card to board and ride the city's buses and trains.

You are to develop a Transit Card **class** that manages a transit cardholder's account. A transit account is associated with a Transit Card. Transit value is money stored in the transit account to pay for bus and train fares. Fares are deducted from the transit account balance when you ride. If your transit card has a negative balance, you will not be allowed to board a bus or train.

A new transit card is purchased for a transit value of $5. You can purchase a new card with up to $100 of transit value.

A transit cardholder may add transit value (money) to their account at any time. The amount of money is added to the account balance and is limited to $100 per transaction. The balance on a transit cardholder's account may not exceed $350.

When the transit cardholder rides a bus or train, the fare is deducted from the balance. If the balance is less than the fare but more than $0.00, the fare will be deducted from the balance resulting in a negative balance. The next time the rider uses the card and the balance is less than $0.00, the rider will not be allowed on the bus or train and the account balance will not be updated.

Create the **class TransitCard** that supports the following methods:

- __init__(amount): Initializes a transit card with a balance amount up to $100 or default amount of $5. If amount is greater than 100.00, **raise ValueError** ('Amount is > 100.00') (5 points)

- **ride(fare)**: Takes a fare amount as input. (10 points)
    - If the card balance is:
        - less than or equal to 0, the fare is not deducted from the card balance and **raise ValueError** ('Card balance is 0 or negative; ride is denied').
        - greater then 0 but less then the fare amount, the fare is deducted from the card balance.
        - greater then 0 and the fare is less than the card balance, the fare is deducted from the card balance.
- **addValue(amount)**: Takes an amount as input. (10 points)
    - If the amount is greater than $100 **raise ValueError** ('Amount is > 100.00;).
    - If the amount is less then or equal to $100 and the amount input increases the current balance to over $350, the balance is not updated; **raise Value Error** ('Card balance will be greater than 350.00; request is denied').
    - If the amount is less then or equal to $100 and the amount input does not increase the current balance to over $350, the balance is updated.
- **balance()**: Takes no input and returns the balance amount. (5 points)
- **__repr__()**: Takes no input and returns a canonical representation of the object. Include the message ' The card balance is ' with the balance amount. (5 points)

Additional Requirements

- (3 points) Include a line of documentation in each method.
- (2 points) There is no status variable in the class. A negative balance does not deactivate the card or the account.
- (5 points) Use a class variable for the maximum transaction amount, use instance variable for the maximum card balance amount. The name of the class is TransitCard; variations of this name are not allowed.
- (5 points) The program executes correctly with no syntax or run time errors
- (5 points) Programs may not use Python statements, functions, data types, etc. that where not discussed in the reading assignment or the lecture notes/videos for this week or previous weeks. If you have a better/different way of writing the code, then upload two versions of your solutions: one that codes according to the specifications and the other the demonstrates advance programming techniques. Global variables are **not** allowed.

Notes

- You will need to test your class. If you wish, you may include your test function in the class file. It is not part of the assignment. When I evaluate your assignment, I will run my test function to determine if all the functionality required is coded in the class.

Submission

ASSIGNMENT EIGHT
DUE DATE: TUESDAY, MAR. 11TH @11:58 PM

- **Include your full name as a comment in the first line of your Python program**
- Code the class in one Python file (.py) labeled as YourName_HW8.py
- Upload one file to Submissions folder in D2L.

IF YOU HAVE ANY QUESTIONS REGARDING THIS ASSIGNMENT
PLEASE POST THEM TO THE ASSIGNMENT EIGHT DISCUSSION FORUM.