Lab 5 Compute, display credit card minimum payment

Goal: **Create a test version of a system to compute the minimum payment due on a credit card account at the new Tiny National Bank of Walterville.**

When complete, your program will ask for credit card account balance (how much you owe). It computes the minimum payment (if any) due:

The minimum payment due is the **larger** *of* **$12** *or* **2.7 %** *of the customer's balance.*

Except:

The minimum payment due will never be greater than the account balance.

The Tiny Bank insists on displaying all values rounded to 2 decimal places:

Right: Your minimum payment due this month is **$13.75**

Wrong*: Your minimum payment due this month is* **$13.74666**

If the customer has a zero or negative balance, your program should politely note that no payment is needed.

Your program should handle any number of customers.

The Tiny Bank wants to use your Python program to verify the logic of their solution; when your solution is working right, they will modify it to get the account balance from the bank's account database [but we won't do database accesses in this course].

Here's an example of how the program might look when running:

```
Tiny National Bank of Walterville
Credit Card Payments

Credit card balance? 100.00
Minimum payment due: $12.0

Another customer (y or n)? y

Credit card balance? 501.00
Minimum payment due: $13.53

Another customer (y or n)? y

Credit card balance? 8.97
Minimum payment due: $8.97

Another customer (y or n)? y

Credit card balance? 0.00
Your balance is current, no payment is needed.

Another customer (y or n)? n
Program Lab5-3.py has ended at your request.
```

*It's possible to get Python to show 12.***00***, but our main focus is on getting the right answers and not showing too many decimals*

Note that the program does **not** *display $13.***527**

Your program will need a main loop to ask for balance, then call a function to compute the minimum payment due given the account balance and returning minimum payment due rounded to 2 decimals.

The function that computes minimum payment due can use either min and max functions or if else style logic to figure out the minimum due. Make sure that whatever method you choose to use actually gets the correct an-

swers.

Grading 40 points + XC

Please start 1st line of each program with
`# by Jane Student` *but using your own name...*

Lab 5-1.py 10 points

1 point - introductory print — something like this

```
Tiny National Bank of Walterville
Credit Card Payments
```

2 points - gets account balance, returns it as a float rounded to 2 decimal places.
2 points - calls a function to set minimum payment due (at this point, need not work right)
5 **loop** that repeats until user says n when asked whether to process another customer.

Lab 5-2.py 20 more points

Function to compute minimum payment due now works correctly when given account balance.
5 points: **docstring** describing function's purpose
5 points: calculates amount due based on percent of account balance
5 points: figures whether to use $12 or value calculated via percent
3 points: decides whether to use amount above or account balance as minimum due
2 points: returns value rounded to 2 decimal places

Run the program; test with the values shown in the example on the first page.

Lab 5-3.py 10 points

Display minimum payment (if one is due), or display a polite message that none is needed.

Extra Credit + 3 points

Modify the minimum payment due function:
min_due = compute_min_due(balance, percent, min_amt)

 # for a $10 min, 3.2%:
min_due = compute_min_due(balance, 0.032, 10.00)