

Thank you for taking the time to work through the official *Mission Capital Software Development Homework*! There are 3 questions. Feel free to use the internet to help you out to best simulate your working environment here, and don't hesitate to reach out with any questions or clarifications!

Please submit the csv output, the code and any supporting documents for all questions.

Thanks,
Mission Capital Software Team

Question 1: The Duchess' Properties

The Duchess of Dunderly is worried that her properties may not be the best investments anymore. Her secretary has given you a database of loans she has provided to borrowers (*loan_data.csv*). Along with this database is a detailed description and format of each field that you will find in the data (*sample_loan_format.pdf*).

The Duchess bids you to use your computer talents to **grade these loans on a scale of 0-100**. The programming language you use to accomplish this is your decision, but the format of the delivery file should have 3 columns:

Loan ID: The loan ID of this specific loan

Loan Grade: The calculated grade after finding exceptions (0 - 100)

Exception IDs: A sorted, pipe-separated list of the exceptions this loan triggers

Example:

Loan ID	Loan Grade	Exception IDs
12345.67	82	1 3
98765.43	39	2 3 4

Exceptions:

Every loan has the potential to break 1 or more exceptions. The rules for each exception are found below:

Exception ID	Exception triggers if:
1	"Maturity Date" < "Completion Date"
2	"Current Index" == FIX && "Current Margin" != "Current Interest Rate"
3	"Current LTV" > "Original LTV"
4	"CMS" > (("Income Borrower 1" + "Income Borrower 2") / 12)

If a loan breaks a rule, it gets points subtracted based on the following rules:

Exceptions 1 through 3:

Based on the Fitch Product Category of that loan, subtract this amount of points per exception type:

Fitch Product Category	Exception ID 1	Exception ID 2	Exception ID 3
U	4	1	0
P	4	1	2
N	4	1	4
L	4	2	6
M	4	2	8
B	4	3	10

Exception 4 is a bit more complicated:

Calculate the following:

```
total_monthly_income = ("Income Borrower 1" + "Income Borrower 2") / 12  
difference = "CMS" - total_monthly_income  
percent = difference / "CMS" * 100
```

Then subtract the amount of points according to this matrix:

Fitch Product Category	percent = 0-10.0%	percent = 10.01-20%	percent = 20.01-30%	percent = 30.01-50%	percent = 50.01-100%
U	0	0	1	3	10
P	2	3	5	12	20
N	4	6	9	20	40
L	6	10	14	30	50
M	8	13	21	40	60
B	10	20	30	50	80

Additional Notes:

You can assume that all fields will be of the correct type (i.e. Numeric will always be able to convert to a float), so don't worry about robustness of data. Try to focus on the logic implementation.

Question 2: Changing the laws

How would your implementation change if exceptions were constantly changing? For example, laws change every year and new exceptions arise that need to be calculated based on the loan date.

How would your implementation change if exceptions got more complicated? For example, if each cell on the Exception 4 table also had a matrix that depended on Product Type.

Note: you do not have to implement anything, just describe the types of challenges and changes that would arise.

Question 3: The Duchess' Curers

Now that you have graded each loan, the Duchess of Dunderly wants to be able to fix these loans. She has hired you as an architect for her team of loan 'Curers'. She wants a Curer to go through exceptions and try to clear them up by calling banks, paying down balances, etc.

Your task is to write up notes or diagrams on how you would design a system that can:

- Store loan information that can be retrieved at will
- Allow Curers to view a list of loans, their grades, and their exceptions
- Allow a Curer to 'cure' a loan by fixing the data of a loan and recalculating the grades and exceptions

You can get as detailed as you wish, but think about things like - what tech stack to use, how will curers access the product, database design, and anything else you find relevant.

Note: You do not have to actually implement this, only write up paragraphs describing your solution and/or drawings. We just ask that the drawings be legible!