Individual Project

✓ Published ✎ Edit ⋮

There are 3 parts to be submitted for Individual project.

Credit card problem:

Part 1: (20 points)

You have a **CSV** file that contains credit card records. Each record is on a line. It contains a field for the card number, the expiration date, and the name of the card holder. The fields are comma-separated. In your system you have the following class structure for the credit cards:

a class CreditCard,

classes VisaCC, MasterCC, AmExCC that are all subclasses of CreditCard,

you can assume more subclasses for other credit card types will be added later on.

You now have to design the method(s) (and maybe additional classes) that reads a record from the file, verifies that the credit card number is a possible account number, and creates an instance of the appropriate credit card class. What design patterns could you use for that?

Important details: Credit card numbers cannot exceed 19 digits, including a single check digit in the rightmost position. The exact algorithm for calculating the check digit is not important for this assignment. You can also determine the card issuer based on the credit card number:

MasterCard	First digit is a 5, second digit is in range 1 through 5 inclusive. Only valid length of number is 16 digits.
Visa	First digit is a 4. Length is either 13 or 16 digits.
AmericanExpress	First digit is a 3 and second digit a 4 or 7. Length is 15 digits.
Discover	First four digits are 6011. Length is 16 digits.

Deliverables:

Upload a PDF document containing the text and diagrams in your Git repo. You can use Astah tool for the diagrams.

- Describe what is the primary problem you try to solve.

- Describe what are the secondary problems you try to solve (if there are any).

- Describe what design pattern(s) you use how (use plain text and diagrams).

- Describe the consequences of using this/these pattern(s).

Hint: you face here (at least) two problems, one has to do with how you figure out what kind of card a specific record is about, the other one with how you create the appropriate objects. Look at behavioural patterns and at creational patterns.

Part2: 15 points (Design only)

Continue with the design from Part 1 and extend it to parse different input file formats (json, xml, csv) and detect the type of credit card and then output to a file (in the **same** format as the input - json or xml or csv) - with each line showing the card number, type of card (if a valid card number) and an error (if the card number is not valid). The design should accommodate newer file formats for the future.

Part 3: 65 points

Implement an application (Java code and JUnit tests) for Part 1 and Part2 - that accepts input file name and output file name and writes an output file in the same format as the input (CSV or JSON or XML). Output should contain the details specified in Part 2.

Submit design and code in the github repo (classroom invite will be sent by grader)

Grader will provide sample JUnit formats and input file formats

input_file-1.csv (https://sjsu.instructure.com/courses/1559474/files/72658397?wrap=1) (https://sjsu.instructure.com/courses/1559474/files/72658397/download?download_frd=1)

(https://sjsu.instructure.com/courses/1559474/files/72658397?wrap=1)

output_file.csv (https://sjsu.instructure.com/courses/1559474/files/72658400?wrap=1) (https://sjsu.instructure.com/courses/1559474/files/72658400/download?download_frd=1)

input_file.json (https://sjsu.instructure.com/courses/1559474/files/72658410?wrap=1) (https://sjsu.instructure.com/courses/1559474/files/72658410/download?download_frd=1)

output_file.json (https://sjsu.instructure.com/courses/1559474/files/72658413?wrap=1) (https://sjsu.instructure.com/courses/1559474/files/72658413/download?download_frd=1)

input_file.xml (https://sjsu.instructure.com/courses/1559474/files/72658416?wrap=1) (https://sjsu.instructure.com/courses/1559474/files/72658416/download?download_frd=1)

output_file.xml (https://sjsu.instructure.com/courses/1559474/files/72658417?wrap=1) (https://sjsu.instructure.com/courses/1559474/files/72658417/download?download_frd=1)

Points	100
Submitting	a file upload

Due	For	Available from	Until
May 13	SP23: CMPE-202 Section 01	-	-

+ Rubric