CSE1322L Assignment 6 - Fall 2024

Introduction:

In the United States, when two individuals or companies have a civil dispute, they may bring that dispute before a court of law. Due to the history of the United States, federal courts are not necessarily above state courts when it comes to hearing lawsuits, as their names would imply. Instead, federal courts are used to hearing lawsuits which involve citizens from two different states, or when the matter being heard involves federal law.

As such, federal courts are very limited in what cases they can actually hear; the states reserve the right to hear any case which happen in their territories, or which involve their citizens, and only through select exceptions may citizens request that their case be heard at a federal court. Cases which do not fall under those exceptions must either be dismissed or remanded (read: sent back to the state court where they were originally filed).

The full set of requirements for a case to be heard by a federal court are fairly complex and, therefore, beyond the scope of this assignment. Nevertheless, we will simulate a program which determines if a list of cases being brought before a federal court can be heard at said court. The cases will be read in bulk from an external file, and all cases in this assignment were originally filed at a state court (and, as such, can either be accepted or remanded, but not dismissed). Specifically, we will be implementing simplified versions of 28 U.S.C. 1331, 28 U.S.C. 1332(a), 28 U.S.C. and 1441(b)(2).

This assignment makes use of file reading and writing. Your submission does not need to worry about valid paths (either relative or absolute), but the names of the files being read from and written to must be correct.

Requirements

The features described below must be in your program.

- A total of 3 classes: Driver, Complaint, StateComplaintException.
- StateComplaintException is a subclass of Exception.
    - It contains a single overloaded constructor which takes in a string. This string is passed to the super class constructor.
- Complaint contains 7 private fields:
    - causeOfAction, plaintiffCitizenship, defendandCitizenship, and originalStateOfFilling, which are all strings.
    - amountInControversy, which is a double.
    - id, which is an integer.
    - nextID, which is a static integer initialized at 1.
- Complaint contains getters for all the fields except nextID.

- Complaint contains an overloaded constructor which accepts 4 strings and 1 double, assigning them as appropriate to the object's fields. It then assigns nextID to id and increments nextID by 1.
- The Driver contains two static methods.
    - processComplaint(Complaint c): a static method which returns nothing and accepts a Complaint. If the complaint is a valid federal complaint, the method does nothing. If the complaint is not a valid federal complaint, it throws a StateComplaintException with the appropriate message. The rules defining valid federal complaints are as follows and should be checked in the order they appear:
        - All complaints whose cause of action are as below are **always** valid federal complaints (i.e. all other bullet points can be ignored):
            - "Equal Protection Challenge"
            - "Title IX Workplace Discrimination"
            - "Prisoner Civil Rights Claim"
            - "Fair Labor Standard Act Claim"
        - If the plaintiff's state of citizenship and the defendant's state of citizenship are the same, throw a StateComplaintException with the message "Lack of Diversity".
        - If the complaint's Amount in Controversy is less than or equal to $75000, throw a StateComplaintException with the message "Amount in controversy less than or equal to $75000".
        - If the defendant's state of citizenship matches the complaint's original state of filing, throw a StateComplaintException with the message "No prejudice through diversity".
        - All other complaints are valid federal complaints.
    - The main method. It must prompt the user for a file name.
        - If the file name doesn't exist, print a message that the file could not be found and terminate the program.
        - If the file is found, process all complaints in the input file. The input file will be in the comma-separated standard, meaning each line holds a single complaint, and all the different pieces of information of the complaint will be separated by commas (without spaces between the pieces of information and the commas):

[causeOfAction],[amountInControversy],[plaintiffCitizenship],[defendantCitizenship],[originalStateOfFilling]

        - Valid federal complaints should be written to "accepted.txt", with each complaint in the following format:

Case ID: [ID]
Cause of action: [causeOfAction]
Amount in Controversy: $[amountInControversy]
Plaintiff's Citizenship: [plaintiffCitizenship]
Defendant's Citizenship: [defendantCitizenship]
Originally filled in: [originalStateOfFilling]
==========

- All invalid federal complaints (i.e.: complaints which threw a StateComplainException) should be written to “remanded.txt”, with each complaint in the following format:

Case ID: [ID]
Cause of action: [causeOfAction]
Amount in Controversy: $[amountInControversy]
Plaintiff’s Citizenship: [plaintiffCitizenship]
Defendant’s Citizenship: [defendantCitizenship]
Originally filled in: [originalStateOfFilling]

Reason for remand: [Exception’s message]
==========

- The main method should then print a message stating that the processing has been completed, where the data has been written to, and the total number of accepted and remanded cases.

Considerations

- Remember that you will get partial credit for partial work. Try to deliver as much of the assignment as you can.
- You may add any helper methods you believe are necessary, but you will not get points for them.
- You can assume that the input file will only contain valid entries (e.g.: a complaint will not be lacking the amount in controversy).
- While there will only be rubric items checking if you caught file exceptions or StateComplaintExceptions, it is good practice to anticipate and catch all exceptions when dealing with user input.
- You can find a sample input file on the FYE Website (“complaints.txt”) and its associated outputs (“accepted.txt” and “remanded.txt”).
    - Your program needs to produce the correct output for **any** valid input file.

Example 1: [User input in red]

[Federal Court Complaint Processor]
Enter file name to process: complaint.txt
No file with name "complaint.txt"
Shutting down...

Example 2: [User input in red]

[Federal Court Complaint Processor]
Enter file name to process: complaints.txt

Processing complete. Accepted cases written to accepted.txt and remanded cases written to remanded.txt
Number of remanded cases: 83
Number of accepted cases: 117
Shutting down...

Submitting your answer:

Please follow the posted submission guidelines here:
https://ccse.kennesaw.edu/fye/submissionguidelines.php

Ensure you submit before the deadline listed on the lab schedule for CSE1322L here:
https://ccse.kennesaw.edu/fye/courseschedules.php