Assignment 03 - Github and group collaboration

CS3354

Submission

This is a group assignment.
Only one team member is required to submit the corresponding link of repository for this assignment.

Github

1. Each team member must *activate* their github-txstate account i.e. to log in at least once.
2. One of the team members must create a **private** repository for this assignment. Suggested name, assignment03-groupNumber
    1. Where groupNumber is 1, 2, 3, …., 1.1, 1.2, …, 3.1, 3.2, …, etc.
3. Add the rest of the group members as collaborators.
4. Also, add the instructor and GTA as collaborators.

Project

Create the usual classes and testing packages (folders).

Classes

In the classes packages include the classes below.

CSDate class

Create the CSDate class (CSDate.java)

CSDate
-day: int -month: int -year: int
+setDay(int d): void +getDay(): int +setMonth(int m): void +getMonth(): int +getMonthName(): string +setYear(int y): void +getYear(): int +CSCdate() +CSCdate(int d, int m, int y)

The constructor receives the day, month, and year as input.

getDay, getMonth, and getYear are accessors that returns the value of the respective variable.
getMonthName returns the corresponding name of the month value.
setDay, setMonth, and setYear are mutators that change the value of the respective variable.

Business rules

Day can only be between 1 and 31. If setDay receives an invalid value, it does not change the day value.
Month can only be between 1 and 12. If setMonth receives an invalid value, it does not change the month value.
Year can only be between 1800 and 2025. If setYear receives an invalid value, it does not change the year value.
If the 3-argument constructor receives an invalid value in day, month, or year; change the corresponding variable to the default values: 1, 1, 1800, respectively.
The no-argument constructor initializes day, month, and year with the default values.
Note: When changing the day and the month, remember to verify the full date e.g. February 30, April 31, etc. are invalid dates.
This class does not consider leap years.

CSDateTime class

Create the CSDateTime class (CSDateTime.java)
CSDateTime is a subclass of CSDate

CSDateTime
-hours: int -minutes: int
-getHours(): int +getMinutes(): int +setHours(int h): void +setMinutes(int m): void +toString(): String +CSCdate() +CSCdate(int d, int m, int y, int h, int min)

The constructor receives the day, month, year, hours, and minutes as input.
getHours and getMinutes are accessors that returns the value of the respective variable.
setHours and setMinutes are mutators that change the value of the respective variable.
Override toString to display the date and time in the following format: mm/dd/yyyy hh:mm

Business rules

Hours can only be between 0 and 23. If setHours receives and invalid value, it does not change it.
Minutes can only be between 0 and 59. If setMinutes receives and invalid value, it does not change it.
If the argument constructor receives an invalid date time in day, month, year, hours, or minutes; change the corresponding variable to the default values: 1, 1, 1800, 0, 0, respectively.
The no-argument constructor initializes day, month, year, hours, and minutes with the default values.

CSEvent class

Create the CSEvent class (CSEvent.java)

CSEvent
-name: String -description: String -eventDate: CSDateTime
+getName): String +getDescription(): String +getEventDateTime(): CSDateTime +setName(String n): void +setDescription(String d): void +toString(): String +CSEvent(String n, String d, int day, int m, int y, int h, int min)

The constructor receives the name, description, day, month, year, hours, and minutes as input.
getName, getDescription, and getEventDateTime are accessors that returns the value of the respective variable. Please notice that getEventDateTime returns the reference to the CSDateTime object inside CSEvent.
setName and setDescription are mutators that change the value of the respective variable.
To change the value of the eventDate, the user of this class should retrieve the eventDate attribute through getEventDateTime, and then call the corresponding mutator for day, month,, year, hours, and minutes, e.g. if e is a CSEvent object, then, to change the day of the event,

```
e.getEventDate().setDay(15);
```

Override toString to display the content of this object in 3 lines

- Name of the event within square brackets
- Description of the event (if any)
- Date and time of the event

Business rules

CSDateTime rules for eventDateTime still apply.
Default values for name and description are “Unknown” and “”, respectively.
In the constructor,

- Initialize name with n value. If n value is “”, then change it to the default value.
- Initialize description with d value.
- Create a CSDateTime object to be referenced by eventDateTime.
- Update the date and time according to the input parameters (day, m, y, h, min)

Testing

MyMain.java

Create an object of CSEvent class.
Display its content using toString method.
Update its name, description, day, month, year, hour, and minutes.
Display its content using toString method.