Assignment 03 (Mandatory)

Class Design

Deadline : April 20, 2021

Marks : 12% of entire course (Equivalent)

This is a Mandatory Assignment. This is counted as class evaluation marks.

1. The weight of this assignment is 13% of the course grade.
2. 50% of viva will be from this assignment.
3. This assignment will test your knowledge of Chapter 1-10.
4. This assignment includes code, output and handwritten components.
5. Marks for this assignment will be finalized based on explaining code choices on viva.
6. If viva is not attended - this assignment will be given a 0 grading.
7. In the PDF, submit UML diagram of the classes with appropriate symbols.
8. In google classroom, submit the .java files and Code output snapshots.
9. Write your Name, student ID and page number at the bottom of each page in pdf.
10. Write the semester and “Assignment 03” at the top of each page.
11. Submit a single .pdf file containing all the handwritten answer.
12. Submission deadline will not be extended.

Problem Description

Design two classes, one for holding passport information and another one for randomly generating passport information.

- The class for holding passport information is called **PassportInfo**. Atmost 15 unique passport numbers will be created. This has the following fields:
    - **i. Data Fields - All data fields should be private. Except for Passport number, all data fields will have a public getter and a public setter method.**
        1. **Passport Number:** a String field with exactly 9 characters - First two are Capital letters, and last seven characters are digits from 0 to 9. When each **PassportInfo** object is created, a passport Number is randomly generated using **StringBuilder** class. Passport number will be created and assigned when any constructor of PassportInfo is created. This will have only a public getter field. Setter field should be private.
        2. **First Name**: A String field containing the given name (First name) of a person.
        3. **Last Name**: A String field containing the surname (Last name) of a person.
        4. **NID number**: A String field containing 10 digit NID number. First digit must not be 0, and the field can not contain any other characters except for 0 to 9.
        5. **Date of Birth**: A field containing month, date and year of the date of birth. Use an appropriate data type, or create your own. You can create your own class for this.
        6. **Picture**: A two dimensional field (40 * 50). Each field contains three elements - Red, Green and Blue color value. Each color value can be from 0 to 255 . Use an appropriate data type, or create your own. You can create your own class for this.
    - **ii. Static Fields - All static fields should be public. Static fields are not object specific, and accessible to all objects.**
        1. **All Passport Numbers :** A static array of 15 passport numbers. When each potential passport number is created - before assigning it to a newly created

---

passport number - it is checked against this array. If this array already contains this passport number, then a new random passport number is generated. (i.e : **i.1**). If a unique passport number is generated - it is assigned to Passport Number, and put inside this array. Assume that no more than 15 passport numbers will ever be created. A value inside this array means a Passport with this number has already been created. You can show appropriate console messages in this case.

2. **All NID numbers**: A static array of 15 NID numbers. Each time a NID is assigned to a passport, if this array does not contain the NID - a new NID is added to this number.

iii. Methods:

1. Create appropriate getter and setter methods and constructors.
2. All Constructors should be private. A static method called **Builder()** should create and return a new **PassportInfo** object with a unique passport number.
3. Create appropriate private or private static methods.
4. Add javadoc comments for each method created explaining the operation of the method, and what the parameters represent.
5. In the comments section, explain why this method is private or public, and why this method is static or non-static.
6. Add a toString method that returns a string containing passport information, Full Name, NID and Date of birth in the same line in an understandable format. Use the StringBuilder or StringBuffer method.

- The class for randomly generating passport info is called **GenerateRandomPassportInfo.** This class will contain the main method, and other methods to help generate Passport information for 15 users.
  - For each randomly generated PassportInfo, randomly assign the following information:
    i. First Name: Should randomly select one of 50 predefined first names.
    ii. Last name: Should randomly select one of 50 predefined last names.
    iii. Date of birth: Random, Should be between 1940 and 2000.
    iv. NID: Random. Two passports can have same NID.
    v. Picture: randomly assign pixels at time of creation.
  - **Main method:**
    i. Create 15 random passport holders and hold their information in an array.
    ii. Given a NID, use the above array to find and display one or multiple passport information containing the same NID. If no NID, then display - “Passport not found”.