Assignment No 1

Important Note:

- For all the questions, you must use Visual Studio and upload all files of the solution.
- Create separate solutions files for each question (i.e., one project per solution)
- Every Question should be named as "k<Year><StudentID>_Q<QuestionNumber>
  for instance: k180123_Q1
- Your Assignment should have proper Objected Oriented Programming and Data Structures
- Last Date for Submission: **October 23, 2021 (11:59 PM)**
- All URLs, File Paths will be included in app.config or web.config. No Hardcoding
- Assignment will be marked as 0 if:
  - Submitted late.
  - The submitted assignment does not open or file is corrupted.
  - Assignments submitted via any medium except for Google Classroom.
  - Copied amongst students or from another source

Scenario:

Your university, in coordination with the Alumni Association is planning to hold its first on-campus elections. Election is being held for 3 positions i.e., President, Vice President and General Secretary at both campuses. Due to security concerns, Alumni Association has decided not to conduct online and due to lack of resources, they have decided not to use any database and conduct the election on paper and then uploaded on the STR system.

University has provided you with two files (as shown in the Appendix section):

- Voter List
- Candidate List

Question No 1: Password Encryptor

The department that is dealing with the Credential List is not familiar with computers. You will have to create a Console Application where the user will provide 3 command line arguments i.e., username, password and Station ID in plain text. Every time they want to add another user, they repeat the same line with newer username, password, and station id. Your task is to take these inputs and store it in a text file named "cList.dat" which will have a comma separated credentials and station id per line for a user, but the passwords will be encoded using base64. You may fix the station id as "Main Campus" = "210001" and "City Campus" = "210002".

Sample Input: command line input

```
C:\> K180123_Q1.exe arif@nu.edu.pk Test1234
210001
C:\> K180123_Q1.exe farooq@nu.edu.pk 1234Test
210002
```

Sample Output: cList.data

```
arifl@nu.edu.pk,VGVzdDEyMzQ=,210001
farooq@nu.edu.pk, MTIzNFRlc3Q=,210002
```

Question No 2: Record a Vote

Create a Windows Form application which will be used by any of the polling station officers. The application will have a login screen and the credentials will be verified against the encoded credential list generated as output of Question No 1 for the given polling station. Once the application has logged on, the polling station officer will be able to record a vote on behalf of a voter by entering his details. Invalid voters cannot vote. Each voter will have three different votes, but the user cannot vote again for any of the positions once recorded. Following details will be entered on screen.

NIC Number of the Voter (Format: 99999-9999999-9)
Position (i.e., President as PRES, Vice President as VPRE or General Secretary as GSEC)
Candidate Name

Against each vote, a record will be saved in an XML file in the following format:

```
<Vote>
  <NIC>99999-9999999-9</NIC>
  <Position>XXXX</Position>
  <CandidateID>X999999</CandidateID>
</Vote>
```

The xml will be stored on a shared location so that it can be uploaded on a centralized location. To reduce the size of the uploads, it has been decided that your application will generate xml file on hourly basis in the following format:
file1: AA_Elec_<StationID>_16Sept21_1000.xml (file generated at 10 am)
file2: AA_Elec_<StationID>_16Sept21_1100.xml (file generated at 11 am)
fileN: AA_Elec_<StationID>_16Sept21_1600.xml (file generated at 04 pm)

Question No 3: Data Aggregation

Create a Console application which will be used to merge all xml files generated from Question 2 into one xml file per polling station. Currently you have xml files generated on hourly basis for each of the polling station but to cumulate results, the Association members have decided to have one file per polling station for future analytics purpose. One of the members has already written code which will trigger your code on hourly basis, they just need logic to merge XMLs (including existing).

Code executed at 10 am:

Sample Input:
AA_Elec_K1_16Sept21_1000.xml
AA_Elec_K2_16Sept21_1000.xml

Sample Output:
AA_Elec_K1_16Sept21.xml
AA_Elec_K2_16Sept21.xml

Code executed at 11 am:

Sample Input:
AA_Elec_K1_16Sept21_1100.xml
AA_Elec_K2_16Sept21_1100.xml
AA_Elec_K1_16Sept21.xml
AA_Elec_K2_16Sept21.xml

Sample Output:
AA_Elec_K1_16Sept21.xml
AA_Elec_K2_16Sept21.xml

Question No 4: Displaying on a Web App

Create a Web Application using HTML/JavaScript/ASP.Net which will be used to display the election results. The application will read the XML files (all campuses) and display data in the following format:

President	
Candidate Name	**Votes**
<Candidate Name 1>	<Number of Votes>
<Candidate Name 2>	<Number of Votes>

Vice President	
Candidate Name	**Votes**
<Candidate Name 1>	<Number of Votes>
<Candidate Name 2>	<Number of Votes>

General Secretary	
Candidate Name	**Votes**
<Candidate Name 1>	<Number of Votes>
<Candidate Name 2>	<Number of Votes>

The web page will refresh every 5 minutes. (Hint: you can use JavaScript's method setTimeout)

Appendix

- **Voter List (VoterList.txt)**
  Format: <StudentID>,<StudentName>,<NIC>

  k060069, Muhammad Ali, 42301-1234567-8
  k040122, Abdul Basit, 42201-2345678-9
  k051679, Irtiza, 42201-3456789-0

- **Candidate List (CandidateList.txt)**
  Format: <CandidateID>,<Candidate Name>,<Position>

  k030001, Ahmed Khan, President
  k040003, Aslam Mehmood, Vice President
  k030002, Kamran Akmal, General Secretary
  k050004, Sarah, President
  k030001, Ahmed Khan, General Secretary
  k060123, Mutahir, General Secretary
  k071241, Shameer, Vice President
  k073000, Khurram, President
  k091234, Rida, Vice President