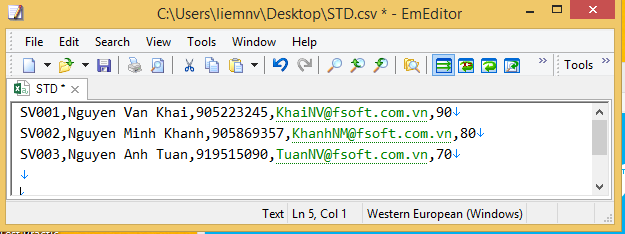
----  **Assignment IO/File** ---

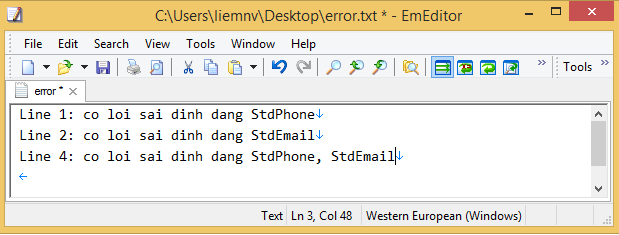
Viết chương trình đọc một file đầu vào có định dạng \*.csv: Nội dung của file csv lưu thông tin của một nhóm sinh viên, bao gồm các thông tin sau: StdNo, StdName, StdPhone, StdEmail, GradePoint.

Ví dụ file đầu vào STD.csv:



* 1. Hãy viết function đọc file csv, sau đó hiển thị toàn bộ thông tin của các sinh viên lên màn hình console.
  2. Viết hàm kiểm tra dữ liệu của file csv, nếu sai định dạng phone hoặc email thì thông báo lỗi ra file error.txt.

Ví dụ file error.txt:



**Assignment OOP and DataBase**

\*Purpose: Students will cover the knowledges:

* Principles of OOP : Encapsulation, Abstraction, Inheritance, Polymorphism
* Access Modifier
* Constructors
* Static, final
* Overload, Override
* Abstract class, Interface
* Is A, Has A relationship
* Super, this keyword
* Exception Handling
* Collection
* ADO.NET Basics
* Debugging With Visual Studio
* Using Log4Net

**Học viên hãy xây dựng ứng dụng như sau:**

1. **Mô tả nghiệp vụ:**

Phần mềm quản lý ứng viên của bộ phận tuyển dụng của FPT software được mô tả nghiệp vụ như sau:

Ứng viên (**Candidate**) khi nộp hồ sơ vào công ty được phần mềm phân thành 3 loại như sau: Ứng viên có kinh nghiệm (**Experience**), Ứng viên mới tốt nghiệp (**Fresher**) , Sinh viên thực tập (**Intern**)

Tất cả các **Candidate** đều có các thuộc tính là: **CandidateID**, **FullName**, **BirthDay**,**Phone**, **Email**, **Candidate\_type**, **Canidate\_count** và phương thức là **ShowInfo** để hiển thị thông tin của ứng viên đó (hiển thị thông tin ứng viên ra màn hình console).

**Trong đó :**

**\* Candidate\_type** có giá trị tương ứng là 0: Experience, 1: Fresher , 2: Intern (tùy vào người dùng nhập vào ứng viên loại nào)

**\* Candidate\_count** dùng để người dùng đếm số lượng candidate trong một đợt người dùng nhập mới ứng viên vào cơ sở dữ liệu. (**mỗi lần người dùng nhập thêm mới ứng viên thì thuộc tính Candidate\_count của class Candidate sẽ tăng lên 1)**

**Ngoài ra :**

Đối với ứng viên **Experience** có thêm thuộc tính: Số năm kinh nghiệm (ExpInYear), Kỹ năng chuyên môn (ProSkill)

Đối vớiứng viên **Fresher** có thêm thuộc tính: Thời gian tốt nghiệp(Graduation\_date), Xếp loại tốt nghiệp (Graduation\_rank) , Trường tốt nghiệp (Education)

Đối với ứng viên **Intern** có thêm thuộc tính: Chuyên ngành đang học (Majors), Học kì đang học (Semester), Tên trường đang học (University\_name)

**Lưu ý:** Tùy mỗi loại ứng viên, phương thức **showMe** sẽ được bổ sung thêm các thuộc tính của riêng loại ứng viên đó.

\*\* Ngoài ra mỗi ứng viên khi tham gia ứng tuyển cần phải nộp bằng cấp nghề nghiệp đi kèm, bộ phận tuyển dụng cần quản lý các bằng cấp này. **Một ứng viên có thể có nhiều bằng cấp.**

Mỗi bằng cấp có các thông tin bao gồm : **CertificatedID**, **CertificateName**, **CertificateRank**, **CertificatedDate**.

**B. Yêu cầu của Học viên:**

**I. Yêu cầu 1:**

1.Học viên hãy thiết kế và viết code của chương trình trên làm sao để tuân thủ theo đúng mô hình **OOP** đã học, áp dụng đầy đủ 4 tính chất : bao đóng (*encapsulation)* , kế thừa (*inheritance)* , đa hình (*polymorphism)* , trừu tượng (*abstraction).*

2. Xác định và viết code constructor cho tất cả các class.

3. Xác định và viết code cho các **abstract** **method**, **abstract** **class**, **override**/**overload** method, **static** field.

4. Xác định Is A, Has A relationship trong phần thiết kế code đã viết ở trên.

**II. Yêu cầu 2 (Day 4, Day 5)**:

1. Học viên hãy viết chương trình cho phép nhập thông tin cho các ứng viên, các thông tin này được lưu vào DataBase (Học viên tự thiết kế DataBase).
2. Sau khi nhập xong các ứng viên vào DataBase, hãy hiển thị số lượng ứng viên vừa được người dùng nhập vào (**chú ý: hãy sử dụng thuộc tính** **Candidate\_count mô tả ở trên,** lưu ý thuộc tính này không cần lưu trữ ở DataBase).
3. Yêu cầu chương trình phải kiểm tra được hợp lệ của các thuộc tính: BirthDay, Email trước khi nhập vào DataBase.

Trong đó Birthday là hợp lệ nếu nằm trong khoảng 1900... năm hiện tại.

**Yêu cầu bắt buộc**: Học viên hãy tạo mới 2 Class exception là : **BirthDayException** và **EmailException** để throw ra màn hình trong trường hợp dữ liệu của một trong hai không hợp lệ (nội dung lỗi do thi sinh tự quyết định). Ngoài ra, nếu có bất cứ một exception nào khác trong quá trình nhập liệu Học viên phải catch được và thông báo nội dung lỗi ra màn hình như sau: "*The system has encountered an unexpected problem, sincerely sorry* !!!”

1. Học viên hãy phân biệt giống và khác nhau giữa String và StringBuffer. Hãy ứng dụng đặc tính của String và StringBuffer để tổng hợp hợp các FullName của tất cả các Candidate (ngăn cách giữa các tên là một dấu “,”) và hiển thị ra màn hình.

Sau khi nhập xong các ứng viên, chương trình cho phép hiển thị danh sách ứng viên với đầy đủ thông tin đã có trong dataBase (Học viên lưu ý hãy sử dụng phương thức **showMe** đã viết code ở trên).