**Thực Hành Lập Trình .NET (Tuần 4)**

**1/ Thiết kế giao diện**

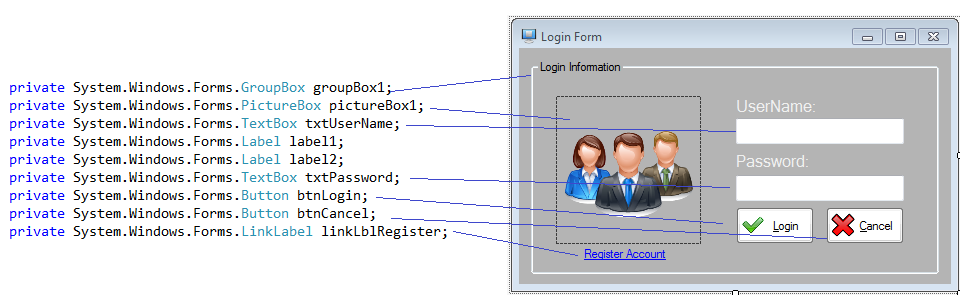
Yêu cầu:

Thiết kế giao diện của ứng dụng hỗ trợ sinh viên đăng ký môn học. Với các chức năng sau

* Sinh viên tạo tài khoản nếu chưa có tài khoản đăng nhập
* Nếu sinh viên có tài khoản thì cho phép sinh viên đăng nhập
* Sau khi đăng nhập vào thì sinh viên có thể chọn các môn học cần đăng ký

Hướng dẫn:

* Tạo một Project dạng **Window Form** tên **ClassRegistrationApp**
* Tại **Solution Explorer** > nhấn phải chuột lên Form1.cs đổi tên là **Login.cs**
* Tại **Solution Explorer** > nhấn phải chuột lên **Login.cs** chọn chế độ **View Design** và thiết kế như sau (kéo thả & đặt tên):



Một vài thuộc tính cần thay đổi như sau

*Form*





 (Chọn Local Resource > Import ảnh Login.ico)



*Picture Box*

 (Vì không sử dụng trong code nên không đổi tên)

 (Chọn hình login-ico.png)

 (Nếu chọn Sketch thì hình sẽ phủ toàn control)

*Label (UserName) và Label (Password)*

 (Chọn Font với Size = 12)

*TextBox (UserName) và TextBox (Password)*



 (Chọn Font với Size = 12)



 (Chọn Font với Size = 12)



*Button (Login) và Button (Cancel)*



 (Chọn Local Resource > Import ảnh Login.png)



 (Lưu ý thêm ít khoảng trắng để ko đè lên hình)





(Chọn Local Resource > Import ảnh Cancel.png)

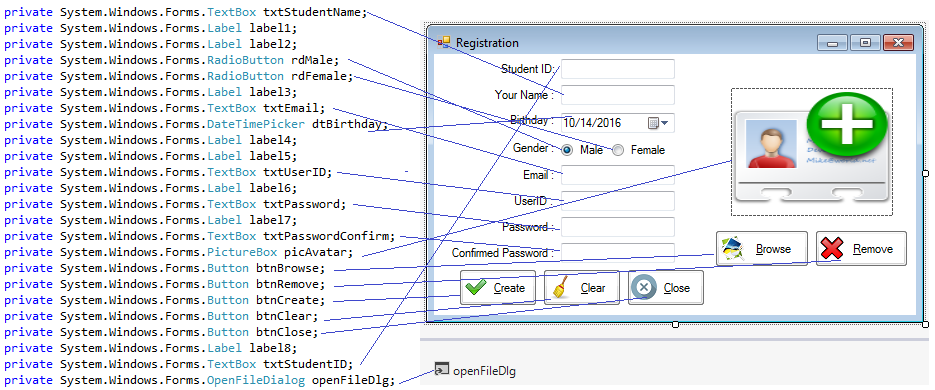


*LinkLabel (Register Account)*





* Tại **Solution Explorer** > nhấn phải chuột chọn Add > Window Form rồi đặt tên là RegistrationForm giúp người dùng tạo một account đăng nhập mới
* Tại **Solution Explorer** > nhấn phải chuột lên **RegistrationForm.cs** chọn chế độ **View Design** và thiết kế như sau (kéo thả & đặt tên):



**Gợi ý:**

* + Các đối tượng giống nhau có thể copy bằng Ctrl + C và Ctrl + V)
  + Muốn canh các control thẳng hàng ta chọn các control (giữ Shift) sẽ thấy thanh canh hàng hiện lên



**Textbox (StudentID, Name, Email, UserID )** - Giống TextBox UserName trong Login Form

**Textbox (Password, Confirmed Password)** – Giống TextBox Password trong Login Form

**Button (Create, Clear, Close, Browse, Remove)** – Giống Button Login trong Login Form

*DatetimePicker (Birthday)*



 (Sẽ giúp hiển thị ngày ngắn gọn)

*Radio Button (Male, Female)*







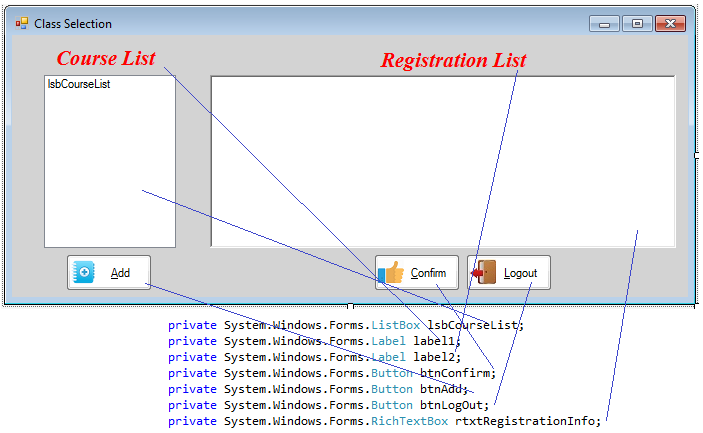


**OpenFileDialog**





* Tạo tiếp tục một Form mới tên là **ClassSelectionForm** giúp cho sinh viên lựa chọn môn học để đăng ký (Giống cách tạo Form **RegistrationForm )**
* Thiết kế Form **ClassSelectionForm** như sau (bằng cách kéo thả và đặt tên)



**Button (Add, Confirm, Logout)** - Giống Button Login trong Login Form

**ListBox (Course List)**





**RichTextBox** (**Registration Form**)



**Label (Course List , Registration List)**

 (Chọn Font Bold Italic + Size 16)

**2/ Khai báo cấu trúc dữ liệu cho ứng dụng**

Yêu cầu:

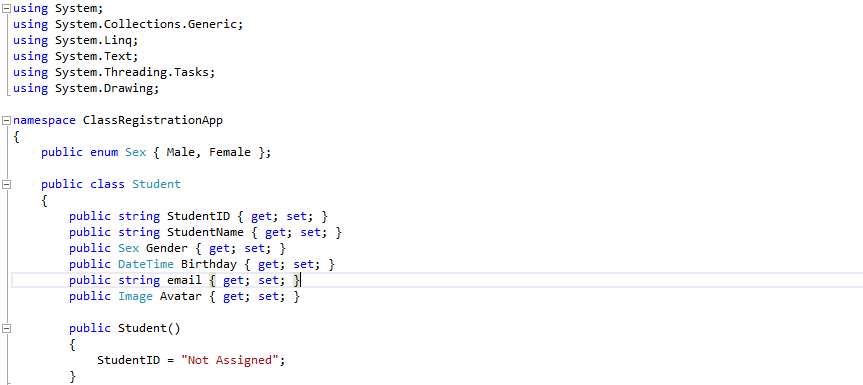
Xây dựng cấu trúc dữ liệu cho ứng dụng đảm bảo

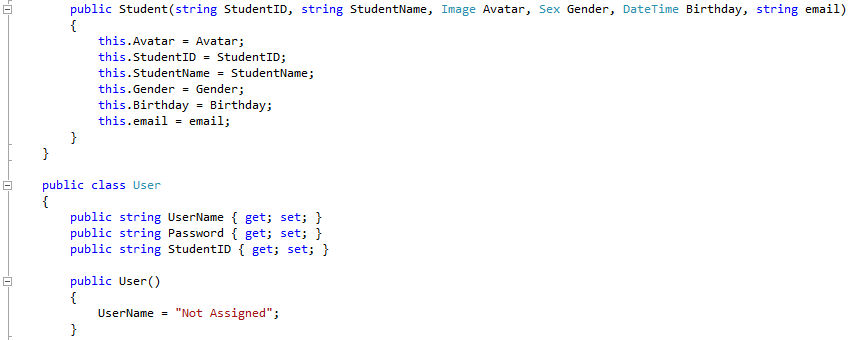
* Một sinh viên cần có đầy đủ thông tin (StudentID, StudentName, v.v…)
* Một sinh viên có 1 account login
* Có một danh sách các môn học để sinh viên đăng ký
* Một sinh viên chỉ được đăng ký duy nhất một danh sách môn học

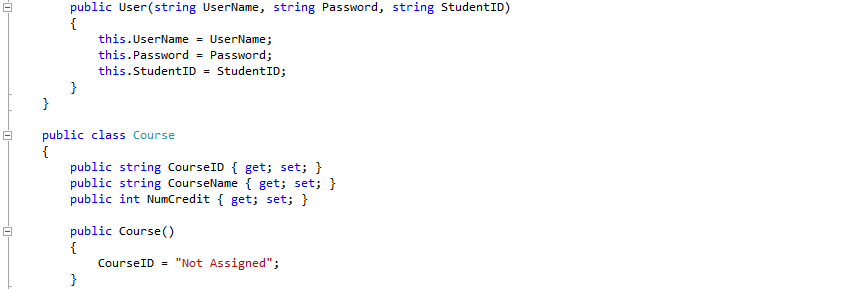
Hướng dẫn:

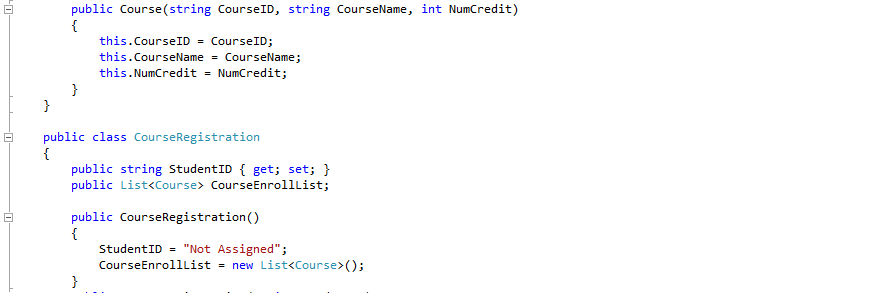
* Tại **Solution Explorer** > nhấn phải chuột vào ClassRegistrationApp project chọn **Add > New Item** rồi nhập tên lớp là **Student**.
* Tạo một lớp **Student** gồm các thuộc tính **StudentID, StudentName, Gender, Birthday, email, Avatar**
* Tiếp đó tạo thêm lớp **User** gồm các thuộc tính **UserName, Password, và StudentID** (nghĩa là mỗi StudentID chỉ có 1 UserName)
* Tạo tiếp lớp **Course** (Môn Học) gồm **CourseID, CourseName và NumCredit**
* Cuối cùng tạo lớp **CourseRegistration**, lớp này lưu thông tin StudentID và danh sách các môn học mà sinh viên đó đăng ký

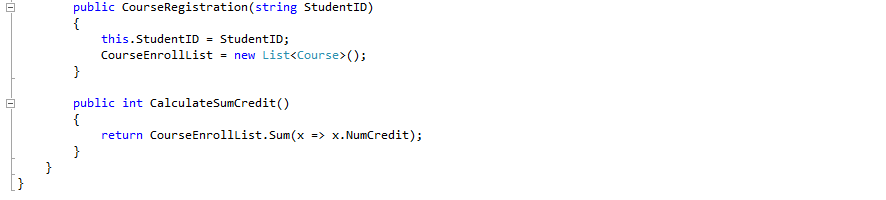
(Lưu ý: Ở đây Avatar là đối tượng kiểu Image nên phải thêm using Syste,.Drawing vào)











**3/ Xử lý sự kiện Form Login**

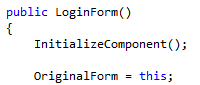
Yêu cầu:

* Form Login thông báo sai thông tin account khi đăng nhập mà chưa có tài khoản
* Khi đó người dùng sẽ nhập vào link “Register Account” để vào màn hình đăng ký
* Khi có account rồi thì nhấn nút Login sẽ vào màn hình đăng ký môn học
* Nhấn nút Cancel sẽ thoát khỏi chương trình

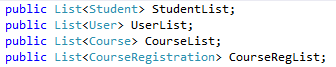
Hướng dẫn:

* Trong suốt quá trình thực thi chương trình, Form Login này sẽ là Form chính tồn tại do vậy ta khai đối tượng OriginalForm để lưu thông tin form này





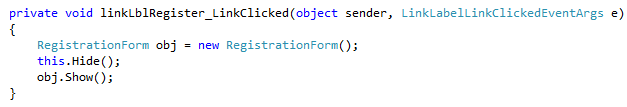
* Trong Form này ta lần lượt khai báo danh sách sinh viên, danh sách người dùng, danh sách môn học, danh sách đăng ký môn học



* Ngoài ra còn thêm hai biến : một để nhớ đường dẫn project (giúp tìm ảnh avatar), một là biến kiểm tra ResetLogIn để biết là Account đó có logout ra hay chưa.

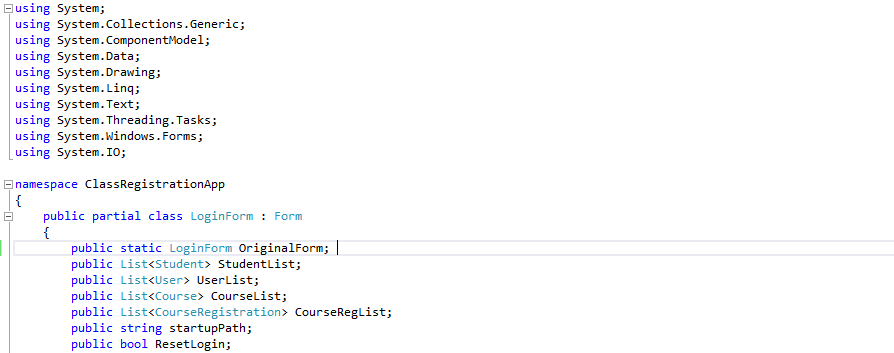


* Thêm các sự kiện sau : Nhấn bút Login, nhấn nút Cancel, nhấn vào Link “Register Account”, và sự kiện VisibleChanged (của Form) (sự kiện này xảy ra khi chuyển từ Form khác về lại Form Login)
* Để gọi một Form khác thì ta tiến hành ẩn Form hiện tại và khởi tạo một Form mới. Sau đó gọi hàm Show để gọi Form mới đó

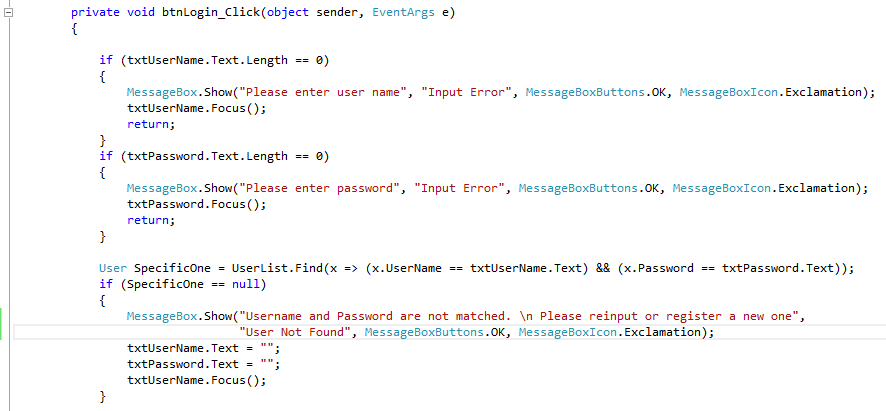


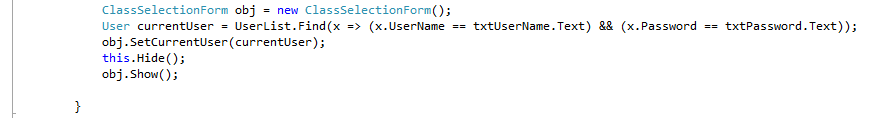
(Ghi chú:

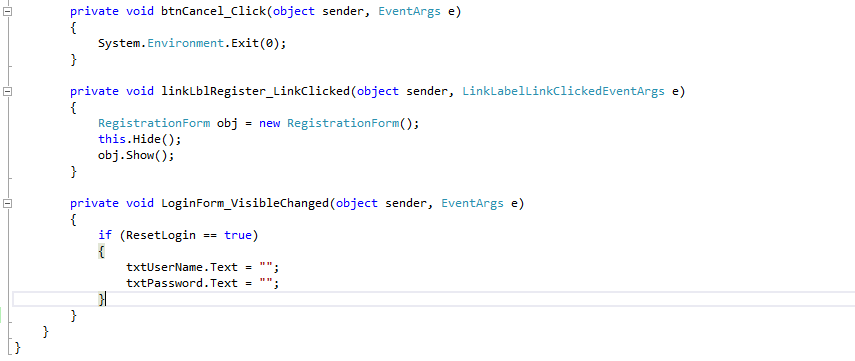
* + Việc tìm ảnh trong thư mục dự án thì ta sẽ cần thêm using System.IO
  + Có sẵn một Student tên là Admin , mã số ST001
  + Có sẵn một account cho Student đó với Username là Admin và Pass là 1234 )











**4/ Xử lý sự kiện Form RegistrationForm**

Yêu cầu:

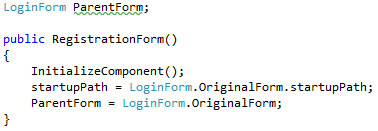
* Form RegistrationForm hiện lên khi nhấn link “Register Account” từ Form Login
* Xử lý sự kiện nhấn nút Browse để kiếm 1 ảnh avatar cho sinh viên
* Xử lý sự kiện nhấn nút Remove khi đó xóa ảnh avatar đi và lấy 1 ảnh trong resourse làm Default
* Xứ lý sự kiện khi nhấn nút Create để tạo một tài khoản đăng nhập mới cho sinh viên

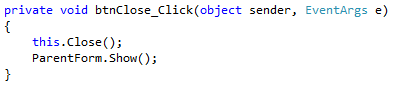
Hướng dẫn:

* Tạo các sự kiện nhấn các nút Browse, Remove, Clear, Close, và Create
* Ảnh lấy mặc định trong resource sẽ gọi bằng lệnh sau



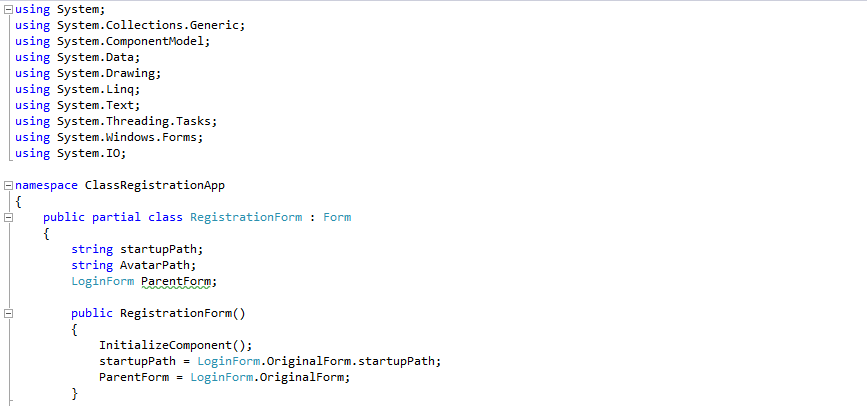
* Khai báo đối tượng ParentForm kiểu LoginForm và gán giá trị cho nó là OriginalForm (ta đã lưu từ khi tạo LoginForm rồi). Khi đó ta muốn quay về Form cha (Login Form) sẽ dễ dàng bằng cách gọi hàm Show

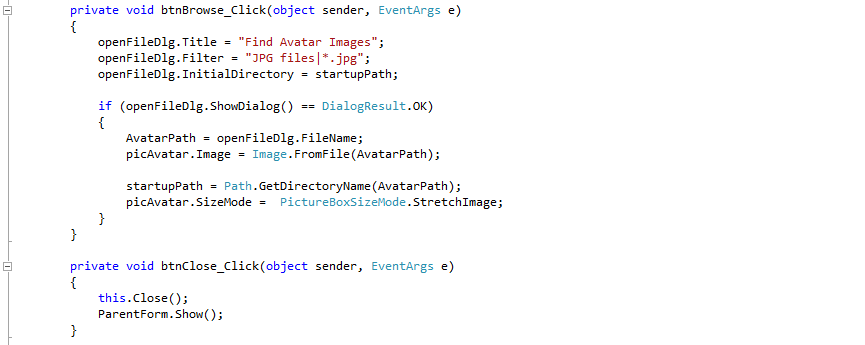


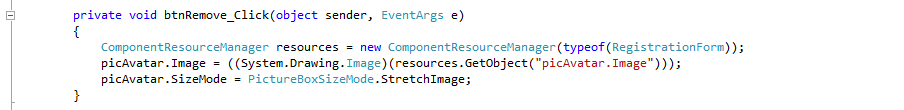


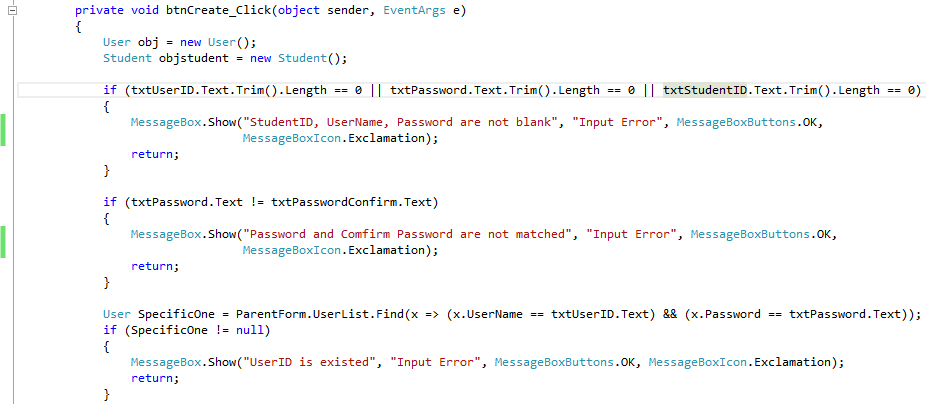
* Khi nhấn nút Create thì ta sẽ thêm một Student và User vào danh sách Student và danh sách User đã được khởi tạo từ Form Cha (LogIn Form)

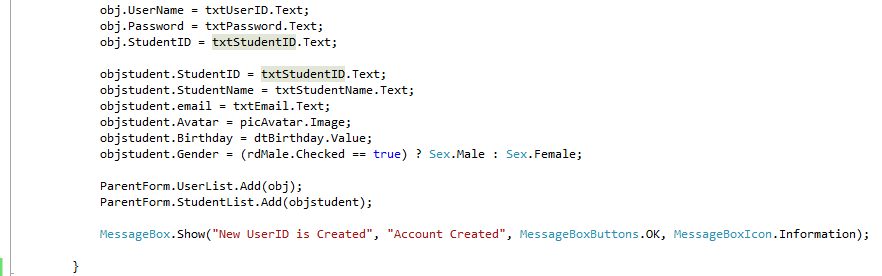


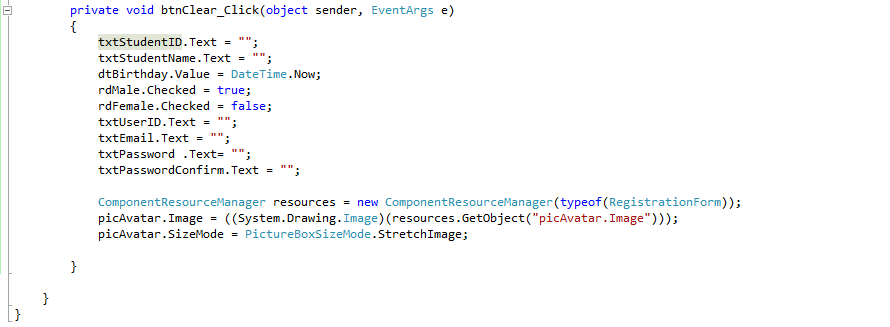












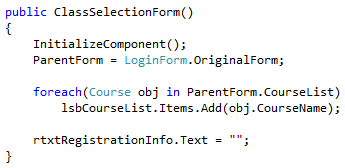
**5/ Xử lý sự kiện Form ClassSelectionForm**

Yêu cầu:

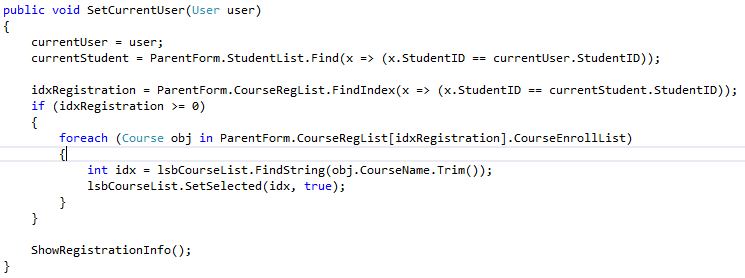
* Form cho người dùng chọn nhiều môn học
* Sau đó nhấn nút Add để thêm môn vào
* Nhấn nút Confirm để xác định môn học
* Nhấn nút Logout để quay lại Form Login
* Khi Sinh Viên đó đã đăng ký môn học rồi thì vào Form này chỉ để xem chứ không thêm được nữa

Hướng dẫn

* Tạo đối tượng ParentForm để lấy thông tin Form cha (Form Login) (Giống như TH Form Registration)
* Lấy danh sách môn học đã được tạo sẵn trong Login Form và đưa vào danh sách môn học cho người dùng chọn



* Tạo biến currentStudent và currentUser cho biết người dùng nào đang đăng nhập thông qua hàm SetCurrentUser (Lưu ý : hàm này được khai báo ở đây nhưng lại được gọi trong sự kiện Login của Login Form)



* Tạo biến idxRegistration cho biết sinh viên đó có đăng ký chưa. Nếu chưa đăng ký biến này sẽ là -1, còn đăng ký rồi thì biến này sẽ là chỉ số trong danh sách đăng ký
* Hàm ShowRegistrationInfo dùng để hiển thị tất cả thông tin đăng ký trên RichTextBox
* Khi thực hiện đăng ký môn học sẽ kiểm tra biến idxRegistration. Nếu biến này = -1 thì mới cho sinh viên đăng ký , còn không thì không cho sinh viên đăng ký thêm.

