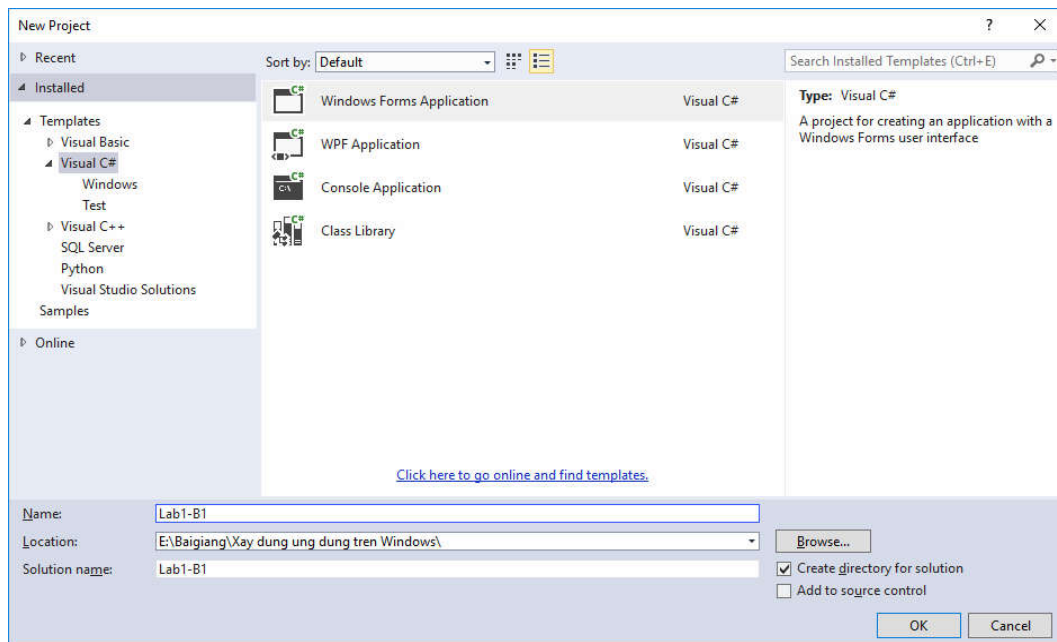


Lab 2

ỨNG DỤNG WINDOWS FORMS VỚI C#

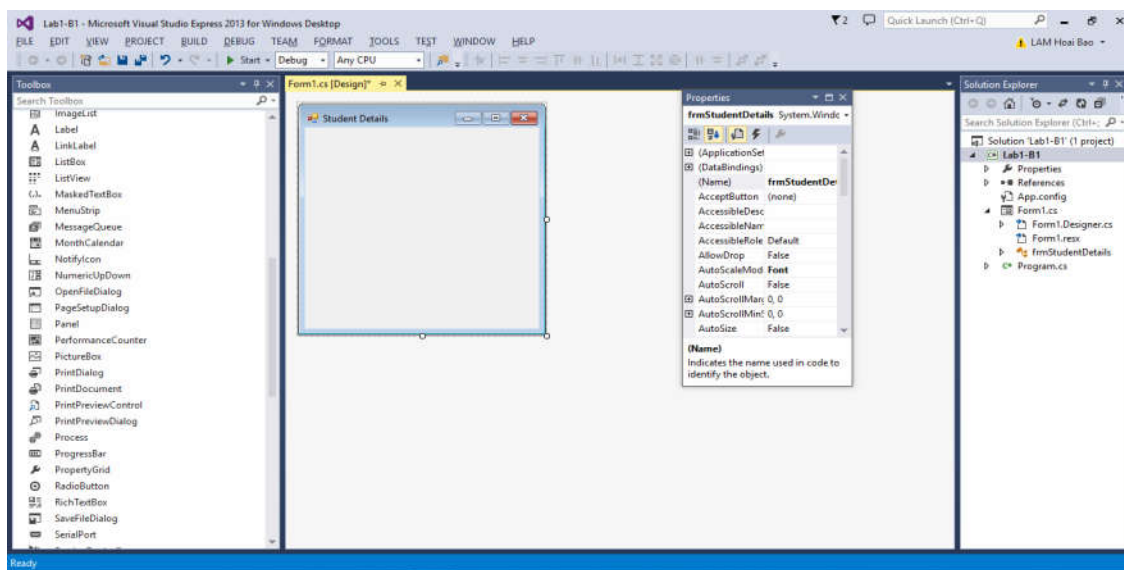
Bài 1:

1. Khởi động Microsoft Visual Studio.
2. Chọn File→New→Project để tạo dự án có tên là Lab2-B1
3. Chọn Project types là Visual C#/Windows; Templates là Windows Application



Hình 2.1: Tạo dự án Windows Application

4. Một form mặc định xuất hiện với tên là Form1. Sửa lại tên lớp của form này là StudentDetails.



Hình 2.2: Form Student Details

5. Thêm các controls lên form để tạo giao diện như hình 2.3

The screenshot shows a Windows application window titled "Student Details". Inside the window, the title "Student's Class Details" is centered at the top. Below the title, there are five input controls arranged vertically: a text box for "Student Name", a text box for "Year of birth", a list box for "Standard/Class" showing items I, II, and III, a dropdown menu for "Division", and a checkbox group for "Hobbies" with options Singing, Dancing, and Surfing. At the bottom of the form, there are two buttons: "Confirm Student Details" and "Clear".

Hình 2.3: Giao diện form Student Details

6. Thiết lập các thuộc tính cho các controls:

Control	Thuộc tính	Giá trị
TextBox	Name Text	txtName
TextBox	Name Text	txtYear
ListBox	Name Items	lstClass I II III IV V VI VII VIII IX X
ComboBox	Name Items	cboDiv A B C D

CheckedListBox	Name Items	lstHob Singing Dancing Swimming Reading Travelling
Button	Name Text	btnConfirm Confirm Student Details
Button	Name Text	btnClear Clear

7. Xác nhận thông tin người dùng nhập vào các controls trên form. Xử lý sự kiện Click của button btnConfirm:

```
private void btnConfirm_Click(object sender, EventArgs e)
{
    int cnt = lstHob.CheckedItems.Count;
    string hobby="";
    for(int index=0;index<=cnt-1;index++)
    {
        hobby = hobby + lstHob.CheckedItems[index];
        if (index<cnt-1)
            hobby = hobby + ",";
    }

    MessageBox.Show(txtName.Text + " born in " + txtYear.Text + " from Class " +
        lstClass.Text + ", Division " + cboDiv.Text +
        " has following hobbies " + hobby);
}
```

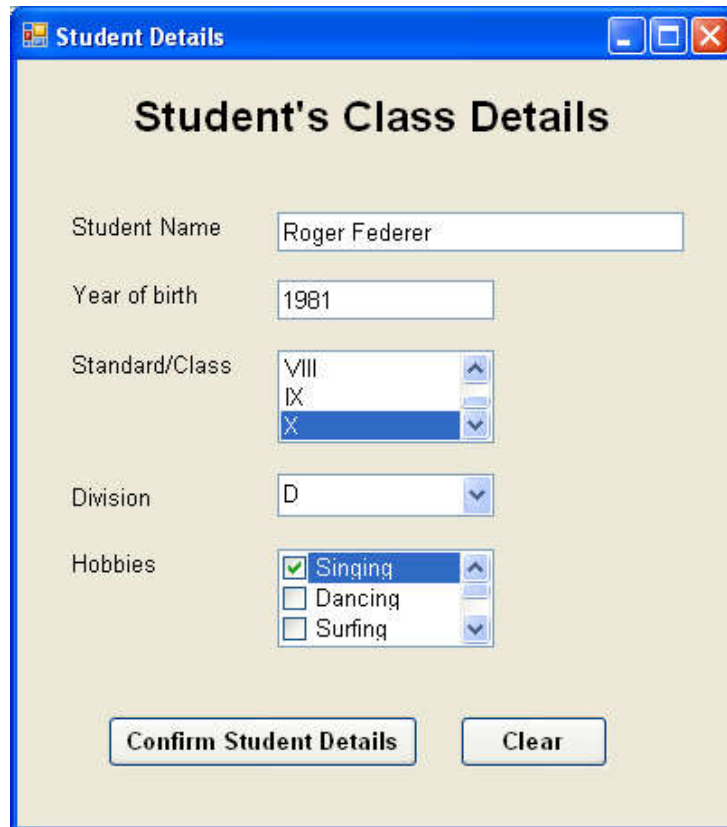
8. Button Clear sẽ xóa hết các giá trị người dùng đã nhập. Xử lý sự kiện Click của button btnClear:

```
private void btnClear_Click(object sender, EventArgs e)
{
    lstClass.ClearSelected();
    cboDiv.Text="";
    lstHob.ClearSelected();
    for(int index=0;index<=lstHob.Items.Count-1;index++)
    {
        if(lstHob.GetItemChecked(index))
            lstHob.SetItemChecked(index,false);
    }
}
```

9. Lưu dự án rồi chọn Build→Build Solution

10. Thực thi chương trình bằng cách chọn Debug/Start

11. Nhập các thông tin vào các controls (hình 2.4)



Student Details

Student's Class Details

Student Name: Roger Federer

Year of birth: 1981

Standard/Class: VIII, IX, X (selected)

Division: D

Hobbies: ☒ Singing, ☐ Dancing, ☐ Surfing

Confirm Student Details Clear

Hình 2.4: Nhập thông tin

12. Sau khi nhập thông tin, click button Confirm Student Details, kết quả sẽ là:



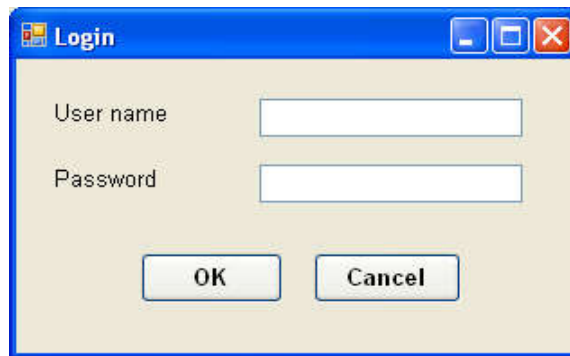
Roger Federer born in 1981 from Class X, Division D has following hobbies Singing,Swimming, Travelling

OK

Hình 2.5: Kết quả của Student Details

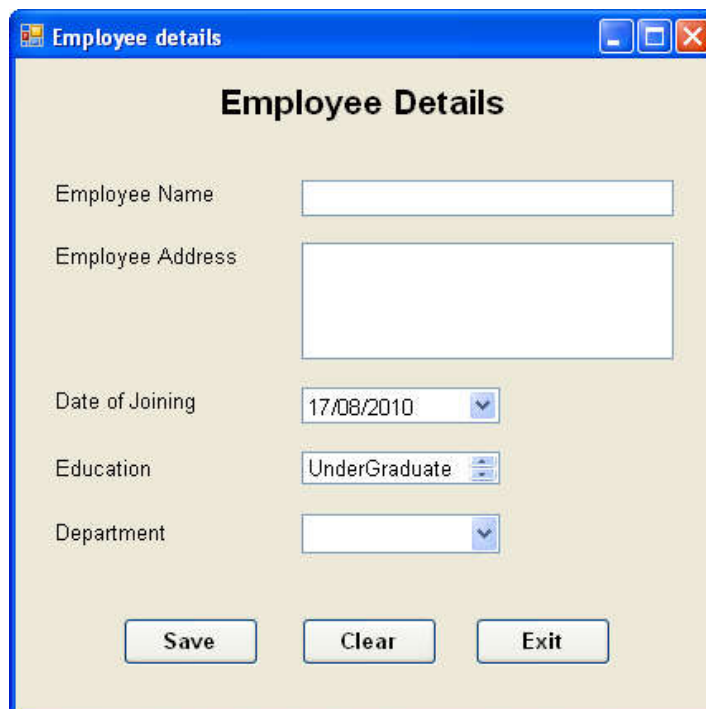
13. Xóa các giá trị đã nhập bằng cách click button Clear.

Bài 2: Tạo 1 ứng dụng có 2 forms. Form thứ nhất là form đăng nhập. Người dùng phải đăng nhập vào hệ thống với tài khoản là **Admin** và mật khẩu là **Admin**. Nếu tài khoản và mật khẩu người dùng nhập vào không chính xác, các thông báo lỗi sẽ được hiển thị. Giao diện form này như hình 2.6:



Hình 2.6: Form đăng nhập

Nếu đăng nhập thành công (tài khoản là **Admin** và mật khẩu là **Admin**), form như hình 2.7 bật ra:



Hình 2.7: Form Employee Details

Form này cho phép người dùng nhập vào họ tên, địa chỉ nhân viên, ngày vào làm. Bên cạnh đó, người dùng có thể lựa chọn mục Education bằng cách lựa chọn từ danh sách (UnderGraduate, PostGraduate). Danh sách Department cho phép người dùng lựa chọn phòng ban (Human Resource, Marketing, R&D) mà nhân viên đó làm việc. Một số yêu cầu khác của form này:

1. Địa chỉ nhân viên cho phép nhập nhiều hơn 1 dòng văn bản.
2. Button Save cho phép xác nhận thông tin người dùng nhập vào.
3. Button Clear cho phép xóa các thông tin người dùng đã nhập trước đó.
4. Button Exit cho phép kết thúc chương trình

Bài 3: Tạo ứng dụng cho phép người dùng xác nhận thông tin đã nhập để giữ chỗ một chuyến bay. Một số thông tin về chương trình:

1. Chương trình cho phép người dùng nhập mã số vé, tên của hành khách, số hộ chiếu của hành khách và ngày bay.

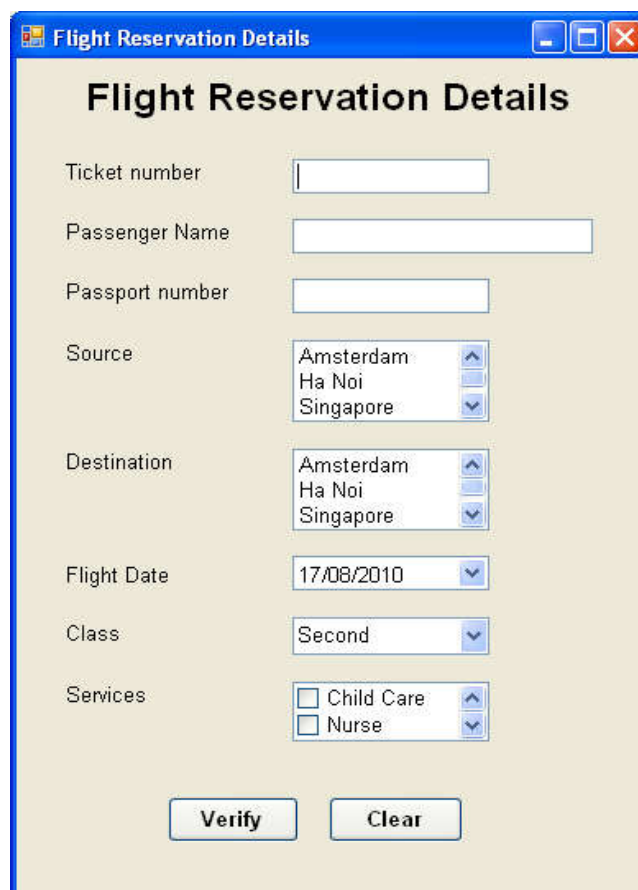
2. Người dùng có thể lựa chọn thành phố xuất phát, thành phố đích và hạng vé (phổ thông, hạng nhất, thương gia).

3. Người dùng có thể lựa chọn thêm một số dịch vụ trên máy bay (chăm sóc trẻ nhỏ, y tá, ...).

4. Button Verify sẽ hiển thị thông báo lỗi nếu thành phố xuất phát và thành phố đích giống nhau.

5. Button Clear cho phép xóa các thông tin người dùng đã nhập.

Form có thể có giao diện như sau:



The screenshot shows a Windows application window titled "Flight Reservation Details". The window has a blue title bar with standard Windows controls (minimize, maximize, close). The main content area has a light beige background. The form is titled "Flight Reservation Details" in bold black text. It contains the following fields and controls:

- Ticket number: A single-line text box.
- Passenger Name: A single-line text box.
- Passport number: A single-line text box.
- Source: A dropdown menu with three options: "Amsterdam", "Ha Noi", and "Singapore".
- Destination: A dropdown menu with three options: "Amsterdam", "Ha Noi", and "Singapore".
- Flight Date: A dropdown menu with one visible option: "17/08/2010".
- Class: A dropdown menu with one visible option: "Second".
- Services: Two checkboxes, "Child Care" and "Nurse", each followed by a small dropdown arrow.
- At the bottom of the form are two buttons: "Verify" and "Clear".

Hình 2.8: Form Reservation Details