

LỜI CẢM ƠN

Lời đầu tiên, nhóm chúng em xin gửi lời cảm ơn sâu sắc đến thầy Hoàng Văn Dũng, phụ trách bộ môn “Cấu trúc dữ liệu và giải thuật”, người đã giúp đỡ nhóm chúng em rất nhiều về kiến thức môn học, định hướng đề tài, hướng dẫn cho nhóm chúng em quá trình thực hiện đề tài này.

Mặc dù đây là bộ môn rất mới lạ, tuy kiến thức của chúng em còn rất nhiều hạn chế, thậm chí còn mơ hồ chưa hình dung rõ về môn học này, nhưng thầy đã cố gắng giảng dạy, tận tình trao đổi để nhóm chúng em có thể tiếp thu đủ kiến thức từ đó hoàn thành đề tài theo đúng yêu cầu. Nhóm chúng em đã rất cố gắng tìm hiểu, nghiên cứu nhiều nguồn tài liệu để hoàn thành bài báo cáo cuối kỳ. Tuy nhiên cũng không thể tránh khỏi những sai sót đáng tiếc xảy ra, hay những hạn chế trong quá trình thực hiện, mong thầy đóng góp ý kiến, bổ sung kiến thức quý báu và sửa chữa để bài báo cáo cuối kỳ của nhóm được hoàn thiện và thành công hơn. Để hoàn thành cuốn báo cáo này theo như đúng thời gian quy định của nhà trường cũng như của khoa không chỉ là sự nỗ lực của nhóm chúng em mà còn có sự trợ giúp, chỉ bảo rất lớn của thầy hướng dẫn.

Chúng em xin chân thành cảm ơn thầy Hoàng Văn Dũng đã giảng dạy chúng em trong suốt học kỳ vừa qua.

Nhóm sinh viên thực hiện:

Nhóm 24

Phan Văn Thạch Quang - 20133083

Nguyễn Đức Linh - 20133007

Võ Hữu Nghĩa – 20133074

MỤC LỤC

LỜI CẢM ƠN.....	1
I. MỞ ĐẦU.....	3
1. Đặt vấn đề.....	3
2. Lý do chọn đề tài.....	3
3. Mục tiêu nghiên cứu.....	4
4. Phương pháp nghiên cứu.....	4
II. NỘI DUNG.....	5
1. Giới thiệu đề tài, công nghệ và phương pháp thực hiện	
1.1 Quản lý ký túc xá.....	5
1.2 Cách quản lý dữ liệu chương trình.....	6
1.3 Công nghệ và thuật toán sử dụng.....	6
2. Phân tích và xây dựng chương trình.....	7
2.1 Giới thiệu về SQL Server.....	7
2.2 Giới thiệu về DAO Pattern.....	8
2.3 Giới thiệu về chương trình “Quản lý ký túc xá sinh viên” trên giao diện Console.....	9
2.4 Sơ đồ khối và code chương trình.....	17
3. Phạm vi giới hạn và hướng phát triển đề tài.....	23
3.1 Phạm vi giới hạn.....	23
3.2 Hướng phát triển đề tài.....	23
III. KẾT LUẬN.....	24

I. MỞ ĐẦU

1. Đặt vấn đề

- Trong thời đại công nghệ số càng ngày càng phát triển như ngày nay, hay người ta còn gọi là “thời đại 4.0”. Việc áp dụng công nghệ số trong công việc là một xu hướng tất yếu, giúp giảm bớt chi phí hoạt động, tăng hiệu quả làm việc. Qua đó ta thấy được tầm quan trọng của việc áp dụng chuyển đổi số trong các công việc quản lý một khối lượng công việc, giấy tờ, dữ liệu lớn ở các môi trường làm việc cụ thể như công ty, bệnh viện, trường học, ... Việc quản lý công việc thông qua các ứng dụng, mạng lưới nội bộ bằng các phần mềm, chương trình chính là một trong những áp dụng phổ biến nhất ta thường thấy, và “Quản lý ký túc xá sinh viên bằng chương trình máy tính” là một trong số đó.

2. Lý do chọn đề tài

- Hiện nay, với sự phát triển mạnh mẽ về cả phần mềm và phần cứng máy tính đã mở ra một thời đại mới trong cách làm việc, vận hành và quản lý trong công việc, đời sống hàng ngày của chúng ta. Với sự bùng nổ về số lượng các thiết bị điện tử thông minh như điện thoại smartphone, máy tính,... cũng như các sự đa dạng về phần mềm, chương trình chạy trên các hệ điều hành có trong những thiết bị ấy đã giúp ta có thể học hỏi, xây dựng và phát triển nhiều cách làm việc hiệu quả hơn. Đặc biệt là trong lĩnh vực quản lý, từ đó nhóm nảy ra ý tưởng về đề tài nghiên cứu: “Xây dựng chương trình máy tính giúp hỗ trợ quản lý ký túc xá sinh viên” được hiệu quả, an toàn và nhanh chóng hơn cách làm việc giấy tờ truyền thống.

3. Mục tiêu nghiên cứu

- Thu thập và xây dựng một cấu trúc dữ liệu hoàn chỉnh liên quan đến việc quản lý ký túc xá trên máy tính.
- Thực hiện phát triển chương trình, phần mềm để quản lý cấu trúc dữ liệu đó một cách hiệu quả với ngôn ngữ C# và hệ quản trị cơ sở dữ liệu SQL Server.
- Tìm hiểu các cấu trúc dữ liệu trong C# và SQL Server.
- Tìm hiểu được cách hoạt động của một số thuật toán sắp xếp dữ liệu.
- Nâng cao kỹ năng thiết kế một chương trình máy tính hoàn chỉnh.
- Rèn luyện và nâng cao kỹ năng làm việc nhóm, khả năng nghiên cứu, đọc và tìm kiếm tài liệu nghiên cứu trên nhiều nguồn khác nhau.

4. Phương pháp nghiên cứu

- Phương pháp nghiên cứu cấu trúc dữ liệu.
- Phương pháp nghiên cứu tài liệu.
- Phương pháp nghiên cứu và xây dựng phần mềm máy tính.
- Phương pháp nghiên cứu và làm việc nhóm.

II. NỘI DUNG

1. Giới thiệu đề tài, công nghệ và phương pháp thực hiện

1.1 Quản lý ký túc xá

- Quản lý ký túc xá là một quy trình quản lý liên quan đến cơ sở vật chất, con người và dịch vụ cơ bản về nơi ở cho sinh viên, người ở ký túc xá. Việc quản lý trước kia được thực hiện đa phần trên các loại giấy tờ, ghi chép thủ công với số lượng người làm việc lớn, phức tạp và gặp nhiều vấn đề tồn đọng. So với hiện nay khi áp dụng các công nghệ số trong thu thập dữ liệu, chỉnh sửa và tính toán các chi phí, việc quản lý đã được giảm bớt gánh nặng về công việc, cũng như thời gian. Giúp cho ta đạt được hiệu quả làm việc tốt hơn, đặc biệt tiết kiệm chi phí hơn.

- Chương trình quản lý ký túc xá là một loại chương trình cho phép ta thu thập, xử lý và tính toán dữ liệu về con người, cơ sở vật chất thông qua các thiết bị điện tử thông minh như máy tính, với khả năng tính toán chính xác và nhanh chóng, khiến cho công việc dễ dàng và tiện lợi hơn so với việc quản lý trên giấy tờ thủ công rất nhiều.

Ưu điểm:

- Chương trình quản lý mang lại một hiệu quả cao trong công việc quản lý ký túc xá, giúp tiết kiệm chi phí nhân sự, thời gian cũng như gánh nặng công việc.

Nhược điểm:

- Tuy vậy vì là một chương trình máy tính nên ta cần phải luôn phát triển, cập nhật, nâng cấp các tính năng cho phù hợp với sự thay đổi trong quản lý, sự phát triển của các lĩnh vực phần cứng hơn.

1.2 Cách quản lý dữ liệu chương trình

- Cách quản lý dữ liệu chương trình được thông qua hệ quản trị cơ sở dữ liệu SQL Server. SQL Server có khả năng cung cấp đầy đủ các công cụ cho việc quản lý từ giao diện GUI đến sử dụng ngôn ngữ cho việc truy vấn SQL. Điểm mạnh của SQL điểm mạnh của nó là có nhiều nền tảng được kết hợp cùng như: ASP.NET, C# để xây dựng chương trình lưu trữ quản lý dữ liệu cũng chính nó có khả năng hoạt động độc lập.

- SQL Server thông thường được sử dụng cho mục đích lưu trữ dữ liệu. Ngoài ra, nó còn mang lại những tính năng làm việc giúp người dùng làm việc hiệu quả hơn như sau:

- Giúp người sử dụng có thể duy trì việc lưu trữ bền vững.
- Cho phép bạn tạo ra nhiều cơ sở dữ liệu hơn.
- Nó có khả năng bảo mật cao

1.3 Công nghệ và thuật toán sử dụng

- Sử dụng công nghệ xây dựng chương trình máy tính bằng ngôn ngữ C# và Visual Studio 2019: để xây dựng và phát triển giao diện cũng như cách chương trình tương tác với cấu trúc dữ liệu có trong SQL Server.

- Ngoài ra chương trình còn sử dụng các cấu trúc dữ liệu mảng, các thuật toán sắp xếp phòng, sinh viên theo một trường dữ liệu cho trước.

2. Phân tích và thiết kế

2.1 Giới thiệu về SQL Server

- Microsoft SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ do Microsoft phát triển. Là một máy chủ cơ sở dữ liệu, nó là một sản phẩm phần mềm có chức năng chính là lưu trữ và truy xuất dữ liệu theo yêu cầu của các ứng dụng phần mềm khác mà có thể chạy trên cùng một máy tính hoặc trên một máy tính khác trong một mạng (bao gồm cả Internet). Microsoft đưa ra thị trường ít nhất hàng chục phiên bản Microsoft SQL Server khác nhau, nhằm vào các đối tượng khác nhau và cho khối lượng công việc khác nhau, từ các ứng dụng một máy nhỏ đến các ứng dụng lớn chạy trên Internet với nhiều người dùng đồng thời.

- SQL Server hỗ trợ các kiểu dữ liệu khác nhau, bao gồm các kiểu nguyên thủy như Integer , Float , Decimal , Char (bao gồm các chuỗi ký tự), Varchar (chuỗi ký tự có độ dài thay đổi), binary (cho các đốm dữ liệu không có cấu trúc), Text (cho dữ liệu dạng văn bản) trong số những người khác . Và cũng cho phép định nghĩa và sử dụng các kiểu kết hợp do người dùng định nghĩa (UDT). Nó cũng cung cấp thống kê máy chủ dưới dạng bảng và chế độ xem ảo (được gọi là Chế độ xem quản lý động hoặc DMV). Ngoài các bảng, một cơ sở dữ liệu cũng có thể chứa các đối tượng khác bao gồm các khung nhìn, các thủ tục

được lưu trữ, các chỉ mục và các ràng buộc , cùng với một bản ghi giao dịch.

2.2 Giới thiệu về DAO pattern

- Data Access Object Pattern or DAO pattern được sử dụng để tách API hoặc hoạt động truy cập dữ liệu cấp thấp khỏi các dịch vụ kinh doanh cấp cao.

- Giao diện đối tượng truy cập dữ liệu - Giao diện này xác định các hoạt động tiêu chuẩn được thực hiện trên (các) đối tượng mô hình.

- Lớp cụ thể Data Access Object - Lớp này thực hiện giao diện trên. Lớp này chịu trách nhiệm lấy dữ liệu từ một nguồn dữ liệu có thể là cơ sở dữ liệu / xml hoặc bất kỳ cơ chế lưu trữ nào khác.

- Đối tượng Mô hình hoặc Đối tượng Giá trị - Đối tượng này là đơn giản chứa các phương thức get / set để lưu trữ dữ liệu được truy xuất bằng cách sử dụng lớp DAO.

Ưu điểm:

- Ưu điểm chính của việc sử dụng các đối tượng truy cập dữ liệu là sự tách biệt tương đối đơn giản và chặt chẽ giữa hai phần quan trọng của một ứng dụng có thể nhưng không nên biết bất kỳ điều gì về nhau và phần này có thể phát triển thường xuyên và độc lập. Thay đổi logic nghiệp vụ có thể dựa trên cùng một giao diện DAO, trong khi các thay đổi đối với logic bên

vững không ảnh hưởng đến các máy khách DAO miễn là giao diện vẫn được triển khai chính xác.

- Tất cả các chi tiết về bộ nhớ đều được ẩn khỏi phần còn lại của ứng dụng (xem phần ẩn thông tin). Do đó, những thay đổi có thể xảy ra đối với cơ chế duy trì có thể được thực hiện bằng cách chỉ sửa đổi một triển khai DAO trong khi phần còn lại của ứng dụng không bị ảnh hưởng. DAO đóng vai trò trung gian giữa ứng dụng và cơ sở dữ liệu. Chúng di chuyển dữ liệu qua lại giữa các đối tượng và các bản ghi cơ sở dữ liệu. Kiểm tra đơn vị mã được tạo thuận lợi bằng cách thay thế DAO bằng một thử nghiệm kép trong thử nghiệm, do đó làm cho các thử nghiệm độc lập với lớp bên vững.

Nhược điểm:

- Những nhược điểm tiềm ẩn của việc sử dụng DAO bao gồm tính trừu tượng bị rò rỉ, trùng lặp mã và đảo ngược trừu tượng. Đặc biệt, việc trừu tượng hóa DAO thông thường có thể che giấu tần suất cao của mỗi lần truy cập cơ sở dữ liệu và cũng có thể buộc các nhà phát triển phải kích hoạt nhiều truy vấn cơ sở dữ liệu để truy xuất thông tin có thể được trả lại trong một thao tác bằng cách sử dụng các hoạt động tập hợp SQL.

2.3 Giới thiệu về chương trình “Quản lý ký túc xá sinh viên” trên giao diện Console

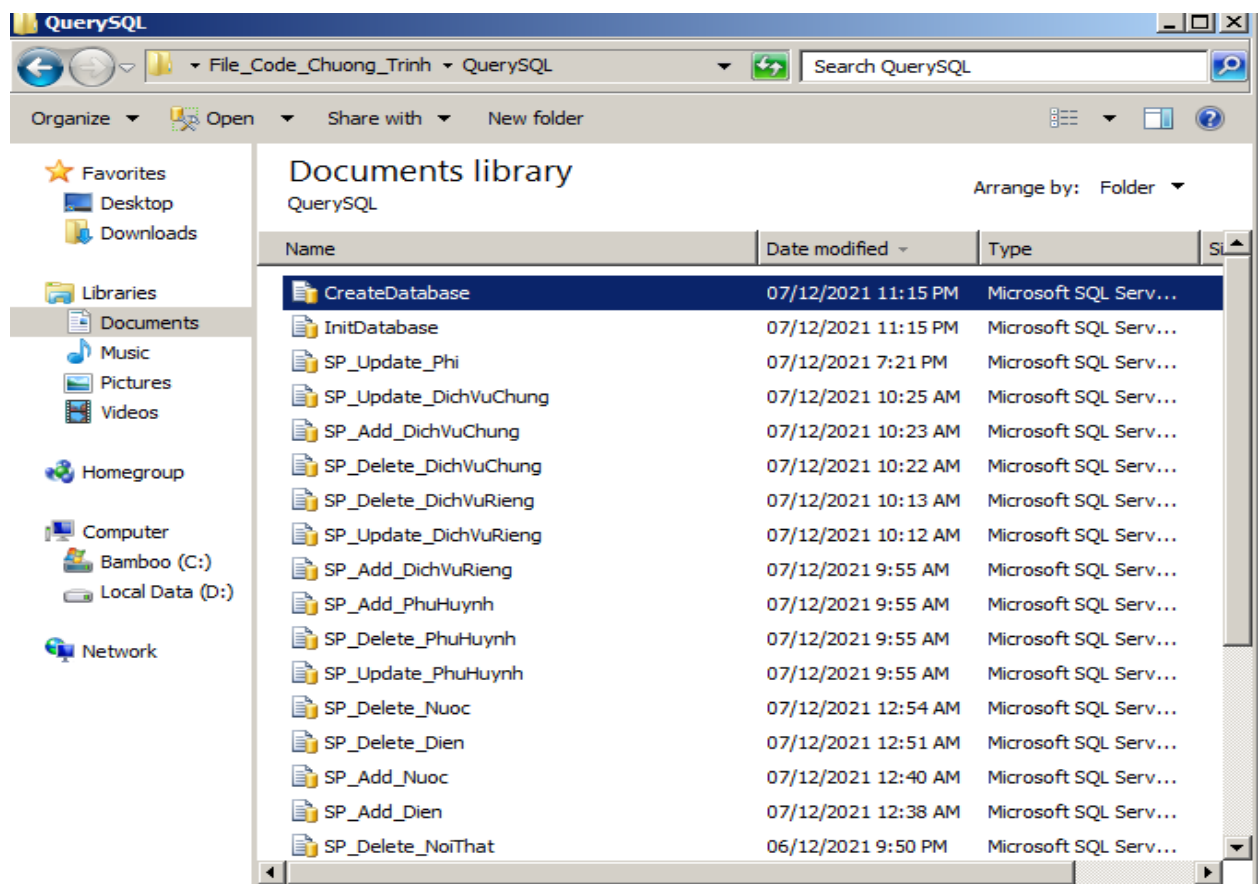
- Chương trình được thiết kế với giao diện người dùng trên Console. Tổng quan cách sử dụng rất đơn giản và dễ hiểu. Tuy nhiên việc khởi tạo Database và thêm các procedure trên SQL Server phải thực hiện thủ công trước khi sử dụng phần mềm.

Đòi hỏi người sử dụng phải có chút kiến thức về hệ quản trị cơ sở dữ liệu SQL Server.

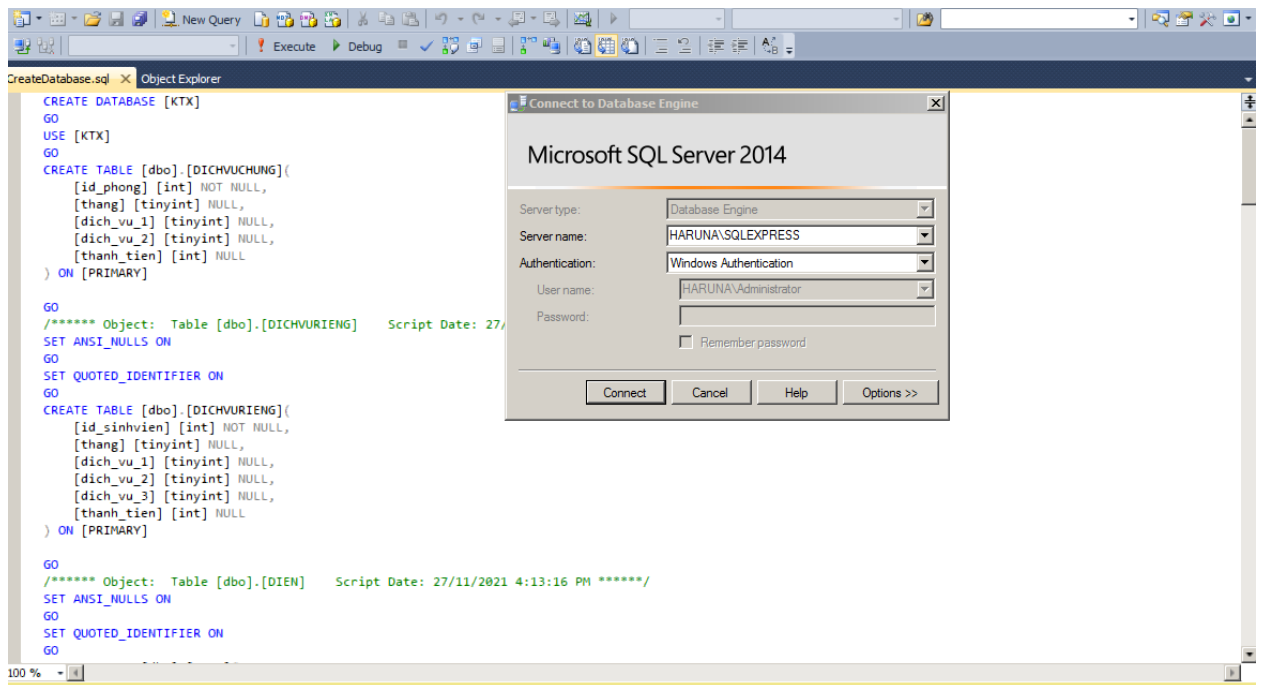
Cách khởi tạo cơ sở dữ liệu trên SQL Server:

B1: Mở thư mục QuerySQL có trong thư mục File_Code_Chuong_Trinh.

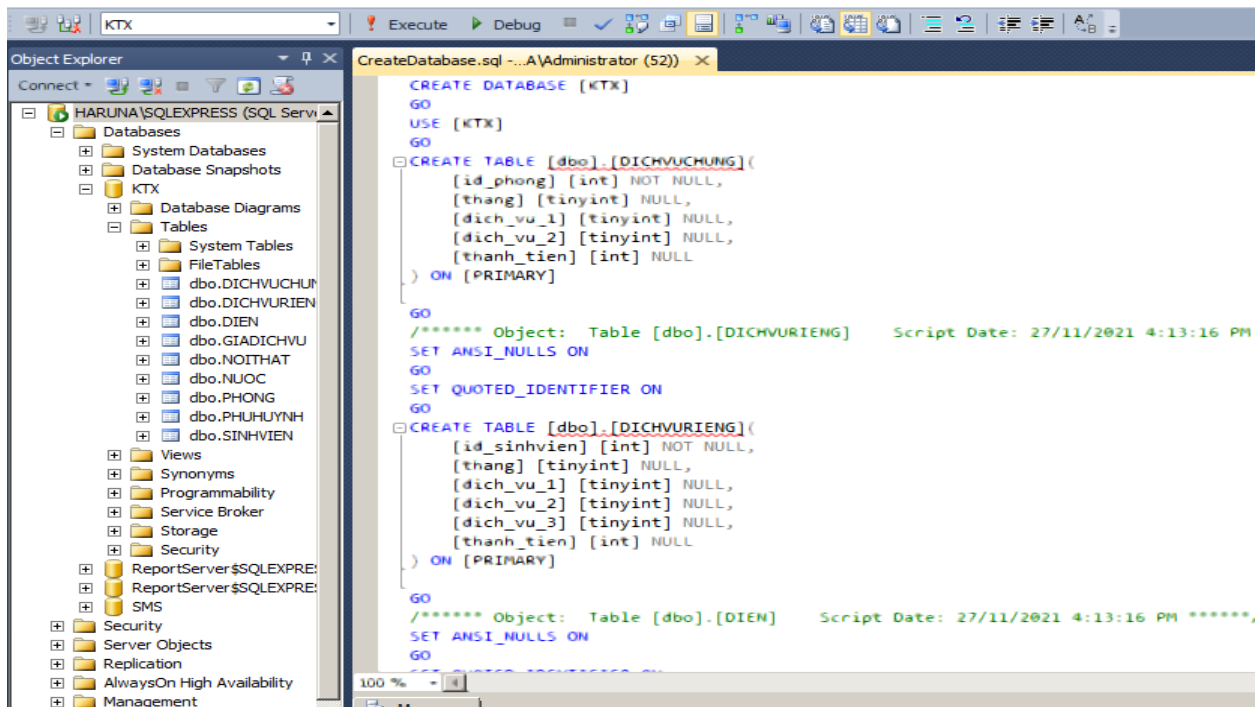
B2: Tìm CreateDatabase.sql, nhấp đúp chuột để mở query đó trong SQL Server Studio Management.



B3: Tại đây ta kết nối tới Database chính trên máy trước bằng cách nhấn “Connect”.



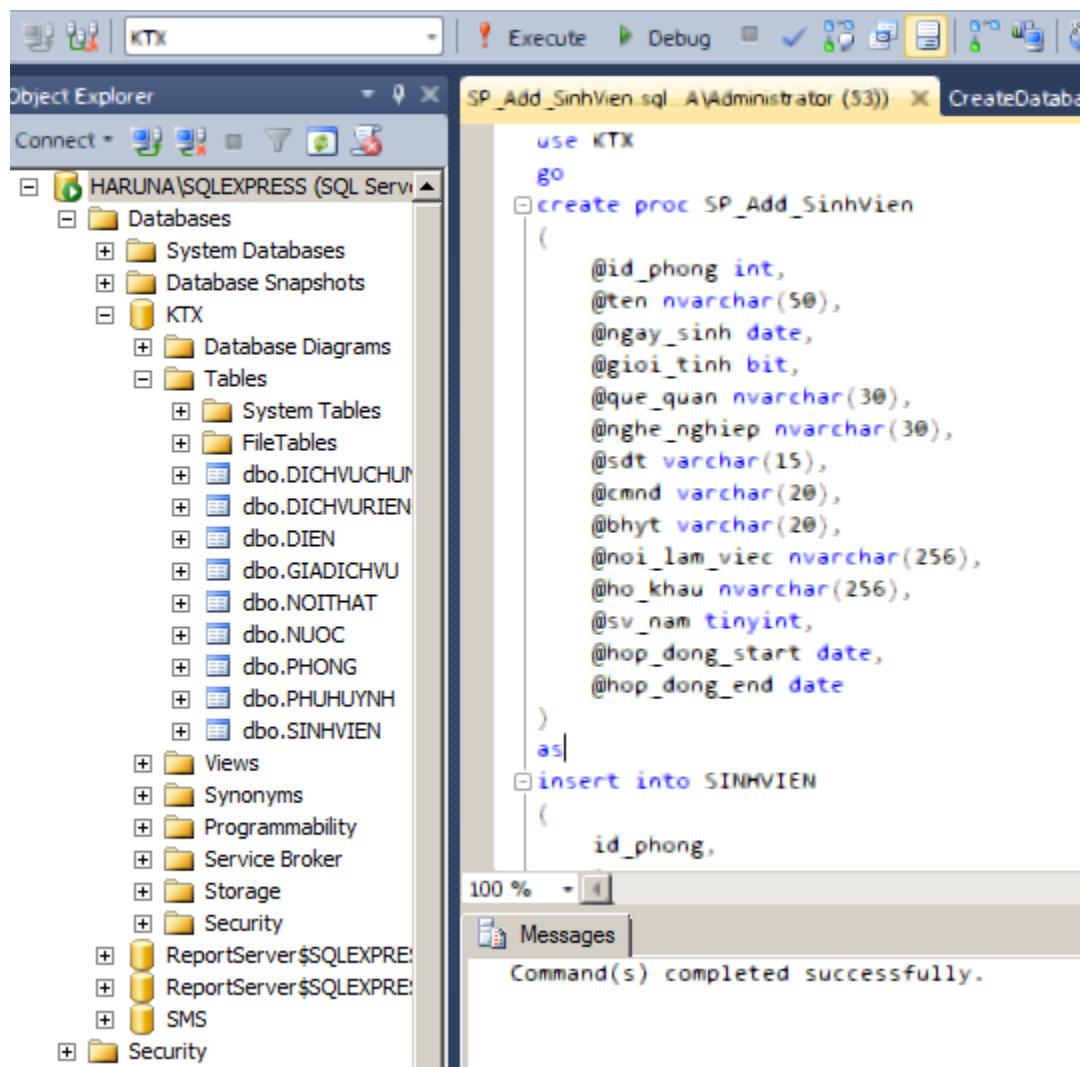
B4: Cuối cùng sau khi đã kết nối xong, ta tìm nút “Execute” để chạy Query trên, vậy là đã khởi tạo xong Database rồi KTX trong SQL Server của máy.



Sau khi đã khởi tạo xong Database KTX, ta cần tiếp tục thêm những Procedure sql vào để sử dụng các Procedure đó tạo tương tác giữa code phần mềm và sql database KTX sau này.

Cách thêm các Procedure sql vào Database hiện có:

B1: Như các bước đã nêu ở trên, ta mở từng file có đầu SP_...trong thư mục QuerySQL để Execute vào trong Database KTX hiện có. Khi hiện “Command(s) completed successfully” là đã thêm được procedure vào Database KTX

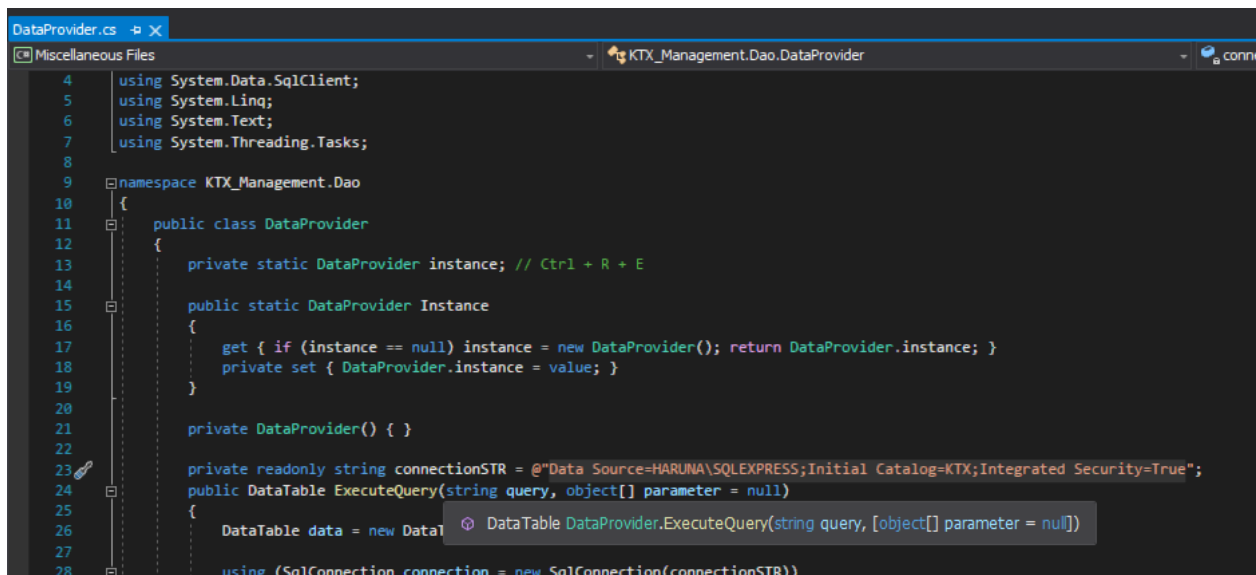


Sau khi đã tạo Database và thêm đầy đủ query SQL, ta dùng Visual Studio để mở Dataprovider.cs trong thư mục Dataprovider và tiến hành lấy chuỗi kết nối tới SQL Server trên máy của bạn để có thể chạy chương trình.

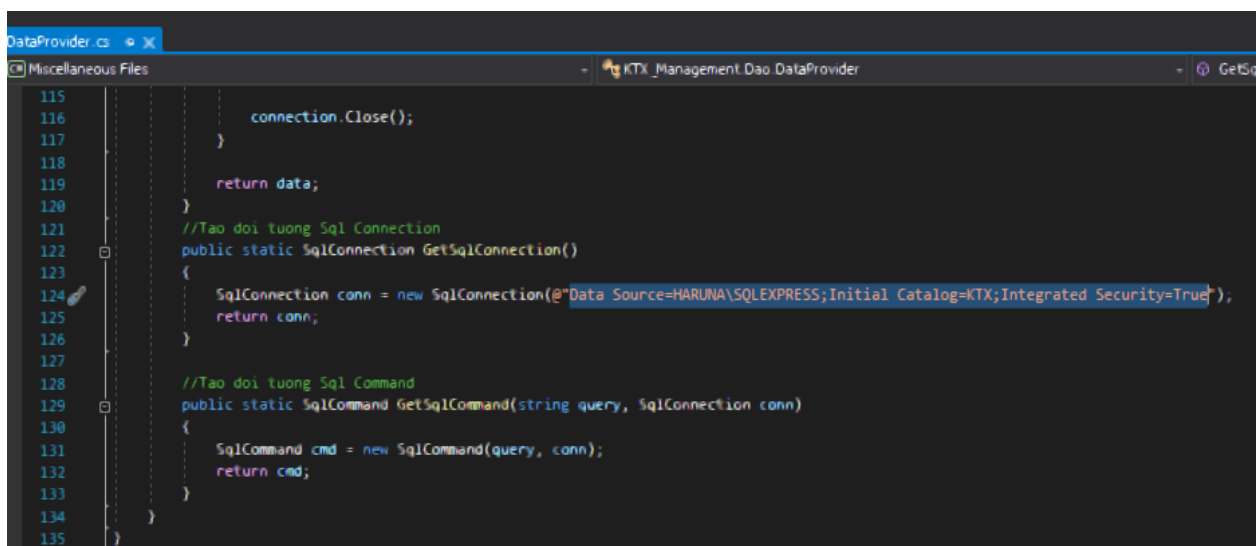
B1: Mở Dataprovider.cs, tìm hai dòng lệnh giống như sau

+ `private readonly string connectionSTR = @"Data Source=...;Initial Catalog=KTX;Integrated Security=True"`

+ `SqlConnection conn = new SqlConnection(@"Data Source=...;Initial Catalog=KTX;Integrated Security=True");`



```
4 using System.Data.SqlClient;
5 using System.Linq;
6 using System.Text;
7 using System.Threading.Tasks;
8
9 namespace KTX_Management.Dao
10 {
11     public class DataProvider
12     {
13         private static DataProvider instance; // Ctrl + R + E
14
15         public static DataProvider Instance
16         {
17             get { if (instance == null) instance = new DataProvider(); return DataProvider.instance; }
18             private set { DataProvider.instance = value; }
19         }
20
21         private DataProvider() { }
22
23         private readonly string connectionSTR = @"Data Source=HARUNA\SQLEXPRESS;Initial Catalog=KTX;Integrated Security=True";
24         public DataTable ExecuteQuery(string query, object[] parameter = null)
25         {
26             DataTable data = new DataTable();
27             using (SqlConnection connection = new SqlConnection(connectionSTR))
28             {
```



```
115         connection.Close();
116     }
117
118     return data;
119 }
120
121 //Tạo đối tượng Sql Connection
122 public static SqlConnection GetSqlConnection()
123 {
124     SqlConnection conn = new SqlConnection(@"Data Source=HARUNA\SQLEXPRESS;Initial Catalog=KTX;Integrated Security=True");
125     return conn;
126 }
127
128 //Tạo đối tượng Sql Command
129 public static SqlCommand GetSqlCommand(string query, SqlConnection conn)
130 {
131     SqlCommand cmd = new SqlCommand(query, conn);
132     return cmd;
133 }
134
135 }
```

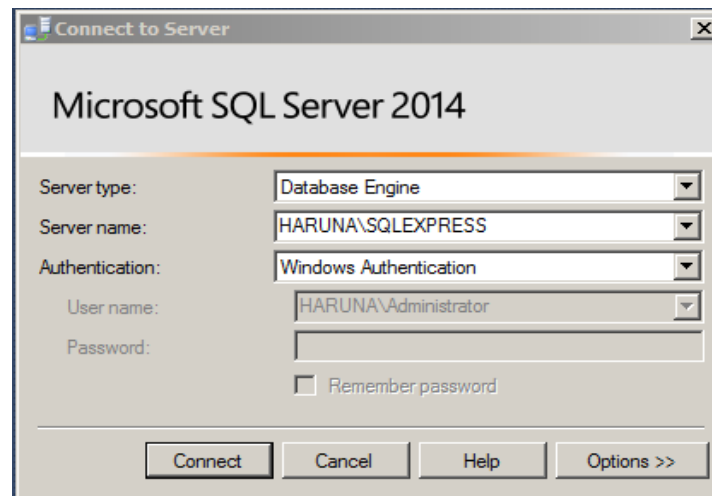
B2: Sau khi đã xác định được hai dòng lệnh trên, ta tiến hành thay thế `Data Source=...` bằng tên của máy chủ SQL Server trên máy của bạn.

Ví dụ trong hình: máy chủ có tên là HARUNA/SQLEXPRESS thì chuỗi kết nối sẽ thành từ:

`Data Source=...;Initial Catalog=KTX;Integrated Security=True`

Thành:

`Data Source=HARUNA/SQLEXPRESS;Initial Catalog=KTX;Integrated Security=True`



B3: Lưu thay đổi, mở file KTX Management C# Project File lên bằng Visual Studio và chọn thư mục Main trong Visual Studio, nhấn chọn KTX_Management.cs để code chương trình chính hiện ra. Ta nhấn Ctrl + F5 thì chương trình sẽ được chạy

Giao diện chương trình quản lý ký túc xá:

Gồm một menu chính với 4 chức năng: quản lý phòng, sinh viên, dịch vụ, xuất file. Mỗi một chức năng sẽ có 1 menu riêng bao gồm tất cả các chức năng phụ liên quan đến một trong bốn mảng trên. Thao tác bằng bàn phím số và nút enter để thực thi chức năng nhập vào.

```
C:\Windows\system32\cmd.exe

===== Chương trình quản lý KTX nhóm 25 =====
==          [1] Quản lý phòng ở KTX              ==
==          [2] Quản lý sinh viên KTX             ==
==          [3] Quản lý dịch vụ KTX              ==
==          [4] Xuất dữ liệu ra file Excel        ==
==          [0] Thoát chương trình                ==
=====
              Mời bạn chọn chức năng chương trình:
```

```
C:\Windows\system32\cmd.exe

===== Xuất dữ liệu sang file Excel =====
==          [1] Bảng dữ liệu tất cả phòng        ==
==          [2] Bảng dữ liệu tiền điện các phòng ==
==          [3] Bảng dữ liệu tiền nước các phòng ==
==          [4] Bảng dữ liệu dịch vụ các phòng   ==
==          [5] Bảng dữ liệu nội thất các phòng  ==
==          [6] Bảng dữ liệu tất cả sinh viên    ==
==          [7] Bảng dữ liệu phụ huynh sinh viên ==
==          [8] Bảng dữ liệu dịch vụ sinh viên   ==
==          [0] Quay lại                          ==
=====
              Mời bạn chọn chức năng:
```

```
C:\Windows\system32\cmd.exe

===== Chức năng quản lý phòng =====
==      [1] Thêm 1 phòng mới                      ==
==      [2] Sửa thông tin phòng                    ==
==      [3] Xoá 1 phòng có sẵn                      ==
==      [4] Thêm nội thất vào 1 phòng              ==
==      [5] Xoá nội thất trong 1 phòng            ==
==      [6] Kiểm tra tình trạng nội thất          ==
==      [7] Cập nhật tình trạng nội thất          ==
==      [8] Thêm ghi chú điện cho phòng           ==
==      [9] Xoá ghi chú điện cho phòng            ==
==     [10] Thêm ghi chú nước cho phòng            ==
==     [11] Xoá ghi chú nước cho phòng            ==
==     [12] Cập nhật phí thu theo tháng            ==
==     [13] Hiển thị tất cả phòng KTX              ==
==     [14] Hiển thị nội thất của phòng           ==
==     [15] Hiển thị phí thu tất cả phòng         ==
==      [0] Quay lại                              ==
=====
      Mời bạn chọn chức năng chương trình:
```

```
C:\Windows\system32\cmd.exe

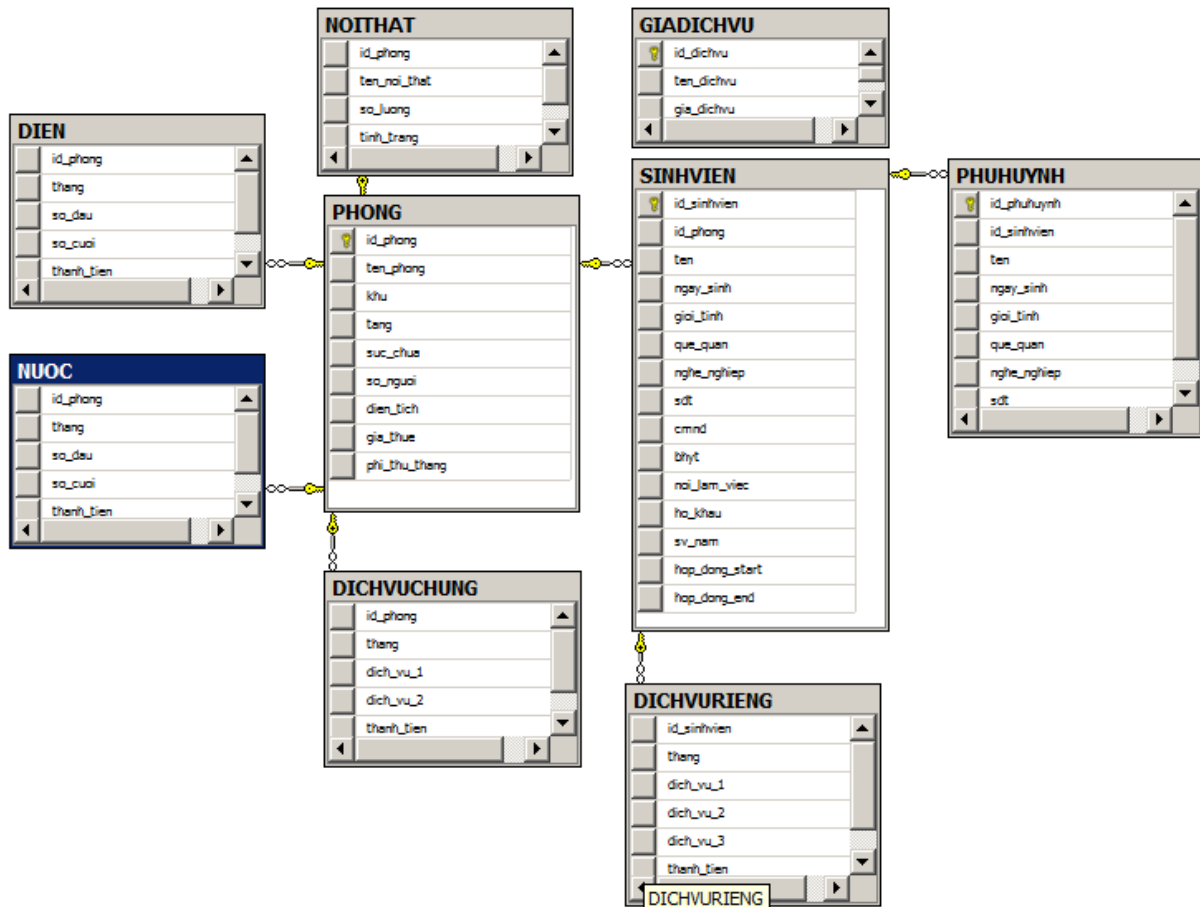
===== Chức năng quản lý sinh viên =====
==      [1] Thêm 1 sinh viên mới                    ==
==      [2] Sửa thông tin 1 sinh viên                ==
==      [3] Xoá 1 sinh viên có sẵn                  ==
==      [4] Thêm thông tin phụ huynh SV            ==
==      [5] Sửa thông tin phụ huynh SV            ==
==      [6] Xoá thông tin phụ huynh SV            ==
==      [7] Cập nhật hợp đồng 1 sinh viên          ==
==      [8] Hiển thị thông tin 1 sinh viên          ==
==      [9] Hiển thị thông tin SV theo phòng       ==
==     [10] Xuất theo thứ tự                      ==
==      [0] Quay lại                              ==
=====
      Mời bạn chọn chức năng chương trình:
```

```
C:\Windows\system32\cmd.exe

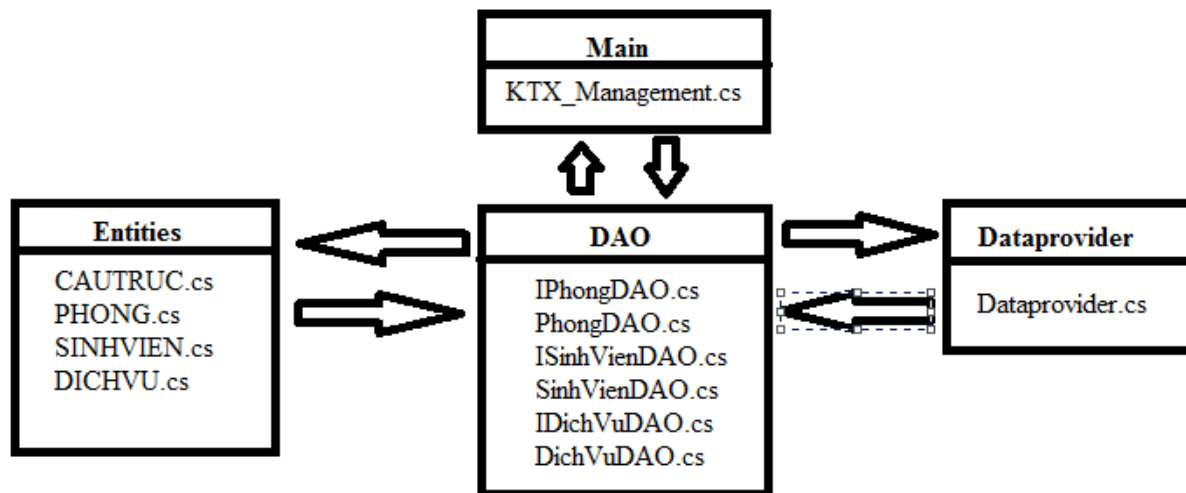
===== Chức năng quản lý dịch vụ =====
==      [1] Thêm ghi chú dịch vụ 1 SV              ==
==      [2] Sửa ghi chú dịch vụ 1 SV              ==
==      [3] Xoá ghi chú dịch vụ 1 SV              ==
==      [4] Thêm ghi chú dịch vụ 1 phòng           ==
==      [5] Sửa ghi chú dịch vụ 1 phòng           ==
==      [6] Xoá ghi chú dịch vụ 1 phòng           ==
==      [7] Hiển thị phí dịch vụ 1 phòng           ==
==      [8] Hiển thị phí dịch vụ 1 sinh viên       ==
==      [0] Quay lại                              ==
=====
      Mời bạn chọn chức năng chương trình:
```


2.4 Sơ đồ khối và code chương trình

Sơ đồ khối về cấu trúc dữ liệu có trong SQL Server:



Sơ đồ khối về cấu trúc chương trình:



- Cấu trúc dữ liệu của chương trình gồm cấu trúc dữ liệu mảng array, cấu trúc dữ liệu struct tự định nghĩa, các lớp, hàm được viết theo hướng đối tượng và các thuật toán sắp xếp mảng để hiển thị.

Cấu trúc dữ liệu struct trong lớp CAUTRUC:

```
public struct NT
{
    public string Ten;
    public short SoLuong;
    public bool TinhTrang; // true neu co hu hong, false neu khong hu hong
};
public struct DIENNUOC
{
    public int SoDau, SoCuoi;
    public int ThanhTien;
};
public struct DV
{
    public int[] LanSuDung; // LanSuDung[i] = so lan su dung dich vu i
    public double ThanhTien;
};
public struct NGUOI
{
    public int ID;
    public string HoTen;
    public string NgaySinh;
    public string GioiTinh;
    public string QueQuan;
    public string NghieNghiep;
    public string SDT;
};
```

Cấu trúc dữ liệu mảng PHONG:

```
List<PHONG> phong
```

```
class PHONG
{
    // Fields
    private int idphong;
    private string tenphong;
    private CAUTRUC.VT vitri;
    private short succhuanguoi;
    private short songuoihientai;
    private CAUTRUC.NT noithat;
    private double dientich;
    private double giathue;
    private CAUTRUC.DIENNUOC dien;
    private CAUTRUC.DIENNUOC nuoc;
    private CAUTRUC.DV dichvuphong;
    private double tongthuthang;
```

Cấu trúc dữ liệu mảng SINHVIEN:

```
SINHVIEN[] sinhvien
```

```
class SINHVIEN
{
    // Fields
    private int idsinhvien;
    private int idphong;
    private CAUTRUC.NGUOI sinhvien;
    private string cmnd;
    private string bhyt;
    private string noilamviec;
    private string diachihk;
    private short svnam;
    private CAUTRUC.NGUOI phuhuynh1, phuhuynh2;
    private string hopdongstart, hopdongend;
    private CAUTRUC.DV dichvurieng;
    // Constructors
```

Một số thuật toán sắp xếp sử dụng trong chương trình:

+ Interchange sort

```
//Interchange sort
{
    Console.Clear();
    Console.WriteLine("Bạn chọn sắp xếp theo ID phòng tăng dần!");
    for (int i = 0; i < phong.Count - 1; i++)
    {
        for (int j = i + 1; j < phong.Count; j++)
            if (phong[i].IDPhong > phong[j].IDPhong)
            {
                PHONG temp = phong[i];
                phong[i] = phong[j];
                phong[j] = temp;
            }
    }
    break;
}
```

+ Bubble sort

```
//Bubble sort
{
    Console.Clear();
    Console.WriteLine("Bạn chọn sắp xếp theo tên phòng A-Z!");
    for (int i = 0; i < phong.Count - 1; i++)
    {
        for (int j = phong.Count - 1; j > i; j--)
        {
            if (phong[j].TenPhong[0] < phong[j - 1].TenPhong[0])
            {
                PHONG temp = phong[j];
                phong[j] = phong[j - 1];
                phong[j - 1] = temp;
            }
        }
    }
    break;
}
```

+ Selection sort

```
//Selection sort
Console.Clear();
Console.WriteLine("Bạn chọn sắp xếp theo ngày hết hạn hợp đồng sớm nhất!");
int min;
for (int i = 0; i < sosinhvien - 1; i++)
{
    min = i;
    for (int j = i + 1; j < sosinhvien; j++)
        if (DateTime.Compare(Convert.ToDateTime(sinhvien[j].HopDongEnd),
            Convert.ToDateTime(sinhvien[min].HopDongEnd)) < 0)
            min = j;
    if (min != i)
    {
        SINHVIEN temp = sinhvien[i];
        sinhvien[i] = sinhvien[min];
        sinhvien[min] = temp;
    }
}
option = "0";
break;
```

+ Insertion sort

```
//Insertion Sort
Console.Clear();
Console.WriteLine("Bạn chọn sắp xếp tăng dần theo niên khoá của sinh viên!");
SINHVIEN x;
int pos;
for (int i = 1; i < sosinhvien; i++)
{
    x = sinhvien[i];
    pos = i;
    while (pos > 0 && x.SVNam < sinhvien[pos - 1].SVNam)
    {
        sinhvien[pos] = sinhvien[pos - 1]; // dời chỗ
        pos--;
    }
    sinhvien[pos] = x;
}
option = "0";
break;
```

Giải thích cấu trúc code:

- Trong thư mục Main bao gồm code giao diện chương trình chính, cách hàm tương tác, nhập xuất dữ liệu giữa người dùng và chương trình quản lý, các thuật toán sắp xếp.
- Trong thư mục Entities bao gồm code chứa lớp các struct cấu trúc dữ liệu, getter và setter, methods của C#.
- Trong thư mục DAO bao gồm các code về interface và phương thức hoạt động giữa chương trình và SQL Server để thêm, chỉnh sửa hoặc xóa dữ liệu có trong Database.
- Trong thư mục Dataprovider chứa code cho các giao thức kết nối, giao thức khởi Query yêu cầu giữa chương trình và SQL Server.
- Khi người dùng mở chương trình lên, các code trong Main sẽ cung cấp giao diện tương tác giữa người dùng và chương trình. Khi người dùng chọn 1 chức năng bất kỳ để chạy, nhập xuất dữ liệu thông tin vào chương trình, hàm trong Main sẽ lưu trữ dữ liệu người dùng bằng các struct có sẵn trong Entities nếu có. Rồi sẽ chạy các phương thức tương tác giữa phần mềm và SQL Server, sử dụng các giao thức kết nối, giao thức khởi tạo Query từ Dataprovider để thực thi câu lệnh. Rồi trả về kết quả dữ liệu được chỉnh sửa trong SQL Server và Console chương trình.

3. Phạm vi giới hạn và hướng phát triển đề tài

3.1 Phạm vi giới hạn

- Do thời gian có hạn, cũng như nhiều thiếu sót, trục trặc về kĩ thuật nên nhóm 24 thực hiện bài “Xây dựng chương trình hỗ trợ quản lý ký túc xá” còn nhiều thiếu sót. Vẫn còn nhiều chức năng cần thêm vào để phục vụ và khai thác hết tiềm năng của chương trình đề tài.

3.2 Hướng phát triển đề tài

- Để phát triển đề tài tốt hơn trong tương lai, ta có thể thêm các chức năng mới vào trong chương trình, tinh chỉnh những chức năng cũ. Có thể nâng cấp giao diện từ Console thành Window Form giúp người dùng tương tác, xem thông tin dữ liệu người dùng được trực quan, dễ dàng hơn trong tương lai.

III. KẾT LUẬN

- Đây là một đề tài tương đối phức tạp và yêu cầu cao về kỹ thuật lập trình, cách xây dựng cấu trúc dữ liệu phù hợp và kỹ năng làm việc nhóm ở mức cao. Nên nhóm chúng em cũng gặp một số hạn chế về phân công công việc, phần cứng hỗ trợ phát triển và về áp dụng những kiến thức mới về quản trị cơ sở dữ liệu từ chương trình C# vào SQL Server. Tuy nhiên nói chung chúng em đã khắc phục được những vấn đề đã gặp, dựa vào kiến thức được học từ thầy và tham khảo trên nhiều nguồn khác nhau đã cho ra thành phẩm ban đầu khá ổn về một chương trình hỗ trợ quản lý ký túc xá cơ bản.

- Trong suốt quá trình thực hiện đề tài đã giúp chúng em học hỏi, áp dụng được nhiều kiến thức bổ ích từ thầy đã dạy, khắc phục những lỗ hổng, thiếu sót trong quản lý làm việc nhóm, xây dựng một cấu trúc dữ liệu và một chương trình máy tính hoàn chỉnh. Đó là một trải nghiệm khá thú vị mà chúng em đã được trải nghiệm trong môn học này, và là kinh nghiệm quý báu cho kỹ năng lập trình, xây dựng cấu trúc dữ liệu, kỹ năng làm việc nhóm một cách hiệu quả cao.