Macau University of Science and Technology

Title: Online Grading System Report

Name : 王虹博 , Wang Hong Bo

Student ID: 18098532-I011-0018

Course name: Advanced Database Systems

Course number: CS108

Date : 2020/12/27

INTRODUCTION

Online Grading System Design is a web application developed by ASP.NET Web application (.NET Framework) in C# Language. In this project, I used two software to build it Visual Studio 2019 and SQL Server Management Studio 18. Besides, the test software is Google Chrome.

This online grading system is a web-based application will be developed using ASP.NET and SQL Server. The main purpose of this system is to facilitate calculation of the average marks or grades of a student through online.

This report mainly describes:

1. Table designed in SQL Server Management
2. GUI designed in ASP.NET and each website main function
3. Interface Operation Manual by user

Content

SQL Design

Create Database and Schema

The first work is creating a database, I create a database called whb. Secondly, we open the whb database find security -> schema -> new schema then we add four schemas Person, Student, Teacher, Homework in the whb database

Create Table

Table 1: Person.Person

Description:

This table is used to create school personnel information, there has five information in this table, SchoolEntityID, PersonType, FirstName, LastName and ModifiedDate.

SchoolEntityID(Primary Key): It is used to indicate a person's ID in school.

PersonType: It is used to indicate whether a person is a teacher or a student.

FirstName: It is used to form a name and represents a person's first name.

LastName: It is used to form a name and represents a person's last name.

ModifiedDate: It is used to record the date of change.

Design Table: (Bold text is the primary key)

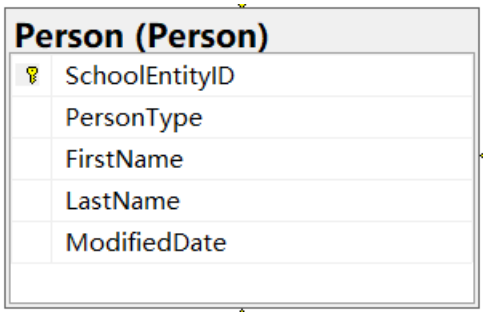
Number	Column Name	Type	Allow Null values
1	**SchoolEntityID**	int	×
2	PersonType	nchar(2)	×
3	FirstName	nvarchar(50)	×
4	LastName	nvarchar(50)	×
5	ModifiedDate	datetime	√

Code:

```
       CREATE TABLE [Person].[Person](
    [SchoolEntityID] [int] NOT NULL,
    [PersonType] [nchar](2) NOT NULL,
    [FirstName] [nvarchar](50) NOT NULL,
    [LastName] [nvarchar](50) NOT NULL,
    [ModifiedDate] [datetime] NULL,
 CONSTRAINT [PK_Person] PRIMARY KEY CLUSTERED
(
    [SchoolEntityID] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTI
AL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
```

Structure: This is the structure of Person.Person and the yellow key is PRIMARY KEY



Picture 1

Table 2: Teacher.Teacher

Description:

This table is used to create teachers in school information, there has seven information in this table, SchoolEntityID, TeacherID, FirstName, LastName, Gender, BirthDate and ModifiedDate.

SchoolEntityID(Primary Key): It is used to indicate a teacher ID in school.

TeacherID(Foreign Key): It is used to indicate a teacher's ID and it is login id.

FirstName: It is used to form a name and represents a teacher's first name.

LastName: It is used to form a name and represents a teacher's last name.

Gender: It is used to indicate a teacher's sex.

BirthDate: It is used to indicate a teacher's Birthday.

ModifiedDate: It is used to record the date of change.

Design Table: (Bold text is the primary key)

Number	Column Name	Type	Allow Null values
1	**SchoolEntityID**	int	×
2	TeacherID	nvarchar(256)	×
3	FirstName	nvarchar(50)	×
4	LastName	nvarchar(50)	×
5	Gender	nchar(1)	×
6	BirthDate	date	√
7	ModifiedDate	datetime	√

Code:

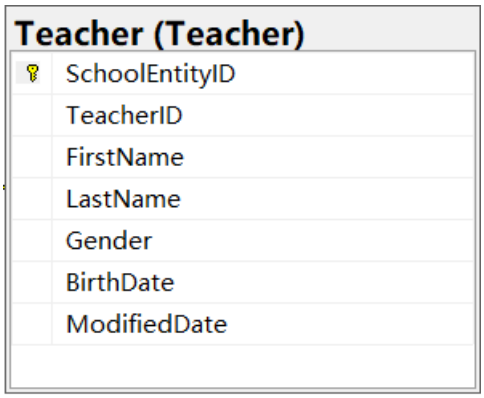
```
CREATE TABLE [Teacher].[Teacher](
```

```
    [SchoolEntityID] [int] NOT NULL,
    [TeacherID] [nvarchar](256) NOT NULL,
    [FirstName] [nvarchar](50) NOT NULL,
    [LastName] [nvarchar](50) NOT NULL,
    [Gender] [nchar](1) NOT NULL,
    [BirthDate] [date] NULL,
    [ModifiedDate] [datetime] NULL,
 CONSTRAINT [PK_Teacher] PRIMARY KEY CLUSTERED
(
    [SchoolEntityID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTI
AL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [Teacher].[Teacher]  WITH CHECK ADD  CONSTRAINT [FK_Teacher
_Person] FOREIGN KEY([SchoolEntityID])
REFERENCES [Person].[Person] ([SchoolEntityID])
GO

ALTER TABLE [Teacher].[Teacher] CHECK CONSTRAINT [FK_Teacher_Person]
GO
```

Structure: This is the structure of Teacher.Teacher and the yellow key is PRIMARY KEY



Picture 2

Table 3: Student.Student

Description:

This table is used to create students in school information, there has eight information in this table, SchoolEntityID, StudentID, FirstName, LastName, Gender, BirthDate, EmailAddress and ModifiedDate.

SchoolEntityID(Primary Key): It is used to indicate a student ID in school.
StudentID(Foreign Key): It is used to indicate a student's ID and it is login id.
FirstName: It is used to form a name and represents a student's first name.
LastName: It is used to form a name and represents a student's last name.
Gender: It is used to indicate a student's sex.
BirthDate: It is used to indicate a student's Birthday.
EmailAddress: It is used to indicate a student's Email address.
ModifiedDate: It is used to record the date of change.

Design Table: (Bold text is the primary key)

Number	Column Name	Type	Allow Null values
1	**SchoolEntityID**	int	×
2	TeacherID	nvarchar(256)	×
3	FirstName	nvarchar(50)	×
4	LastName	nvarchar(50)	×
5	Gender	nchar(1)	×
6	BirthDate	nvarchar(50)	√
7	EmailAddress	nvarchar(50)	√
8	ModifiedDate	datetime	√

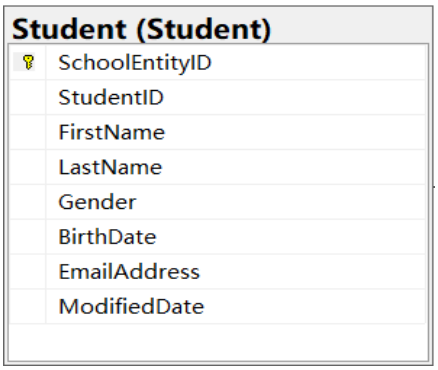
Code:

```
       CREATE TABLE [Student].[Student](
    [SchoolEntityID] [int] NOT NULL,
    [StudentID] [nvarchar](256) NOT NULL,
    [FirstName] [nvarchar](50) NOT NULL,
    [LastName] [nvarchar](50) NOT NULL,
    [Gender] [nchar](1) NOT NULL,
    [BirthDate] [nvarchar](50) NULL,
    [EmailAddress] [nvarchar](50) NULL,
    [ModifiedDate] [datetime] NULL,
 CONSTRAINT [PK_Student] PRIMARY KEY CLUSTERED
(
    [SchoolEntityID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTI
AL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [Student].[Student]  WITH CHECK ADD  CONSTRAINT [FK_Student
_Person] FOREIGN KEY([SchoolEntityID])
REFERENCES [Person].[Person] ([SchoolEntityID])
GO
```

```
ALTER TABLE [Student].[Student] CHECK CONSTRAINT [FK_Student_Person]
GO
```

Structure: This is the structure of Student.Student and the yellow key is PRIMARY KEY



Picture3

Table 4: Person.Password

Description:

This table is used to create school personnel password information, there has five information in this table, SchoolEntityID, LoginID, Password, PersonType and ModifiedDate.

SchoolEntityID(Primary Key): It is used to indicate a person's ID in school.

LoginID: It is a person's ID, use it can login online grading system

Password: It is a person's password, use it can login online grading system

PersonType: It is used to indicate whether a person is a teacher or a student.

ModifiedDate: It is used to record the date of change.

Design Table: (Bold text is the primary key)

Number	Column Name	Type	Allow Null values
1	**SchoolEntityID**	int	×
2	LoginID	nvarchar(256)	×
3	Password	varchar(10)	×
4	PersonType	nchar(2)	×
5	ModifiedDate	datetime	√

Code:

```
CREATE TABLE [Person].[Password](
    [SchoolEntityID] [int] NOT NULL,
    [LoginID] [nvarchar](256) NOT NULL,
    [Password] [varchar](10) NOT NULL,
    [PersonType] [nchar](2) NOT NULL,
    [ModifiedDate] [datetime] NULL,
 CONSTRAINT [PK_Password] PRIMARY KEY CLUSTERED
```

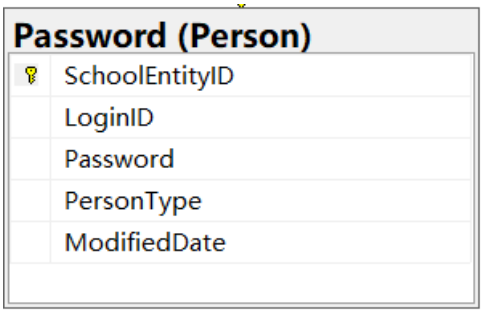
```
(
    [SchoolEntityID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTI
AL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [Person].[Password]  WITH CHECK ADD  CONSTRAINT [FK_Passwor
d_Person] FOREIGN KEY([SchoolEntityID])
REFERENCES [Person].[Person] ([SchoolEntityID])
GO

ALTER TABLE [Person].[Password] CHECK CONSTRAINT [FK_Password_Person]
GO

EXEC sys.sp_addextendedproperty @name=N'MS_Description', @value=N'Rando
m value concatenated with the password string before the password is ha
shed.' , @level0type=N'SCHEMA',@level0name=N'Person', @level1type=N'TAB
LE',@level1name=N'Password', @level2type=N'COLUMN',@level2name=N'Passwo
rd'
GO
```

Structure: This is the structure of Person.Password and the yellow key is PRIMARY KEY



Picture4

Table 5: Teacher.Course

Description:

This table is used to create courses by teachers, there has three information in this table, CourseID, CourseName, TeacherID.

CourseID(Primary Key): It is used to indicate a course's ID in school.

CourseName: It is used to record teacher's course.

TeacherID: It is used to indicate a teacher's ID and it is login id.

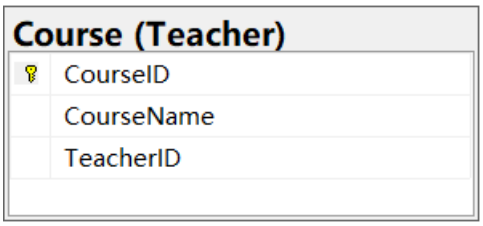
Design Table: (Bold text is the primary key)

Number	Column Name	Type	Allow Null values
1	**CourseID**	int	×
2	CourseName	nvarchar(50)	×
3	TeacherID	nvarchar(256)	×

Code:

```
CREATE TABLE [Teacher].[Course](
    [CourseID] [int] NOT NULL,
    [CourseName] [nvarchar](50) NOT NULL,
    [TeacherID] [nvarchar](256) NOT NULL,
 CONSTRAINT [PK_Course] PRIMARY KEY CLUSTERED
(
    [CourseID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTI
AL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
```

Structure: This is the structure of Teacher.Course and the yellow key is PRIMARY KEY



Picture 5

Table 6: Student.Course

Description:

This table is used to create student's courses by teachers, there has three information in this table, CourseID, Course, StudentID.

CourseID(Primary Key): It is used to indicate a course's ID in school.

Course: It is used to record student's course.

StudentID: It is used to indicate a student's ID and it is login id.

Design Table: (Bold text is the primary key)

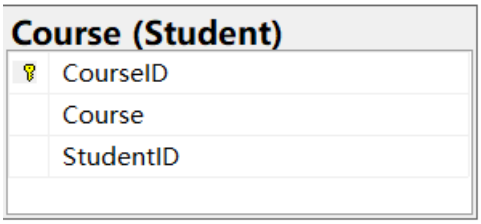
Number	Column Name	Type	Allow Null values
1	**CourseID**	int	×
2	Course	nvarchar(50)	×

3	StudentID	nvarchar(256)	×

Code:

```
CREATE TABLE [Student].[Course](
    [CourseID] [int] NOT NULL,
    [Course] [nvarchar](50) NOT NULL,
    [StudentID] [nvarchar](256) NOT NULL,
 CONSTRAINT [PK_Course] PRIMARY KEY CLUSTERED
(
    [CourseID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTI
AL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
```

Structure: This is the structure of Student.Course and the yellow key is PRIMARY KEY



Picture 6

Table 7: Teacher.Homework

Description:

This table is used to create homework by teachers, there has fifteen information in this table, HomeworkID, TeacherID, Course, Title, Homework1, Homework1_Weight, Homework2, Homework2_Weight, Homework3, Homework3_Weight, Homework4, Homework4_Weight, Homework5 and Homework5_Weight

HomeworkID (Primary Key): It is used to indicate a homework's ID by teacher.

TeacherID: It is used to record which teacher left the homework.

Course: It is used to record which class left the homework.

Title: The homework title.

Homework1: The teacher left the first topic that students need to complete.

Homework1_Weight: The points of first topic.

Homework2: The teacher left the second topic that students need to complete.

Homework2_Weight: The points of second topic.

Homework3: The teacher left the third topic that students need to complete.

Homework3_Weight: The points of third topic.

Homework4: The teacher left the fourth that students need to complete.

Homework4_Weight: The points of fourth topic.

Homework5: The teacher left the fifth topic that students need to complete.

Homework5_Weight: The points of fifth topic.

DueDate: It is used to record the student Submission Assignment time.

Design Table: (Bold text is the primary key)

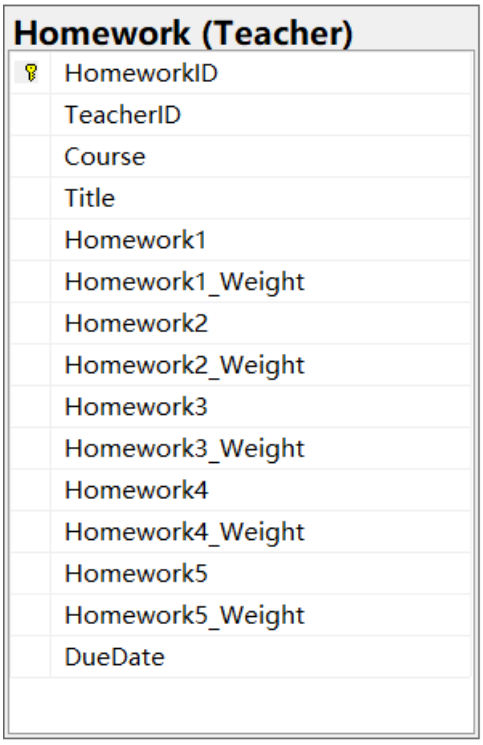
Number	Column Name	Type	Allow Null values
1	**HomeworkID**	int	×
2	TeacherID	nvarchar(256)	×
3	Course	nvarchar(50)	×
4	Title	nvarchar(100)	×
5	Homework1	nvarchar(MAX)	×
6	Homework1_Weight	float	×
7	Homework2	nvarchar(MAX)	√
8	Homework2_Weight	float	√
9	Homework3	nvarchar(MAX)	√
10	Homework3_Weight	float	√
11	Homework4	nvarchar(MAX)	√
12	Homework4_Weight	float	√
13	Homework5	nvarchar(MAX)	√
14	Homework5_Weight	float	√
15	DueDate	datetime	√

Code:

```
CREATE TABLE [Teacher].[Homework](
    [HomeworkID] [int] NOT NULL,
    [TeacherID] [nvarchar](256) NOT NULL,
    [Course] [nvarchar](50) NOT NULL,
    [Title] [nvarchar](100) NOT NULL,
    [Homework1] [nvarchar](max) NOT NULL,
    [Homework1_Weight] [float] NOT NULL,
    [Homework2] [nvarchar](max) NULL,
    [Homework2_Weight] [float] NULL,
    [Homework3] [nvarchar](max) NULL,
    [Homework3_Weight] [float] NULL,
    [Homework4] [nvarchar](max) NULL,
    [Homework4_Weight] [float] NULL,
    [Homework5] [nvarchar](max) NULL,
    [Homework5_Weight] [float] NULL,
    [DueDate] [datetime] NULL,
 CONSTRAINT [PK_Homework] PRIMARY KEY CLUSTERED
(
    [HomeworkID] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTI
AL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
```

Structure: This is the structure of Teacher.Homework and the yellow key is PRIMARY KEY



Picture 7

Table 8: Homework.Student

Description:

This table is used to the homework and it is need answer by students, there has twenty fourth information in this table, ID, HomeworkID, StudentID, TeacherID, Course, Title,TotalScore,Answer1,Answer1_Score,Answer1_Evaluation,Answer2,Answer2_Score,Answer2_Evaluation,Answer3,Answer3_Score,Answer3_Evaluation,Answer4,Answer4_Score,Answer4_Evaluation, Answer5, Answer5_Score, Answer5_Evaluation.

ID(Primary Key): It is used to indicate homework ID each update homework students.

HomeworkID: It is used to record which teacher left the homework.

StudentID: It is used to record which student update the homework.

TeacherID: It is used to record which teacher left the homework.

Course: The class number of the homework.

Title: The title of the homework.

TotalScore: The score of the whole homework.

Answer1: The student who answer the first question

Answer1_Score: The student gets the point from the first question.

Answer1_Evaluation: The teacher evaluation the first question which is answer by student

Answer2: The student who answer the second question

Answer2_Score: The student gets the point from the second question.

Answer2_Evaluation: The teacher evaluation the second question which is answer by student

Answer3: The student who answer the third question

Answer3_Score: The student gets the point from the third question.

Answer3_Evaluation: The teacher evaluation the third question which is answer by student

Answer4: The student who answer the fourth question

Answer4_Score: The student gets the point from the fourth question.

Answer4_Evaluation: The teacher evaluation the fourth question which is answer by student

Answer5: The student who answer the fifth question

Answer5_Score: The student gets the point from the fifth question.

Answer5_Evaluation: The teacher evaluation the fifth question which is answer by student

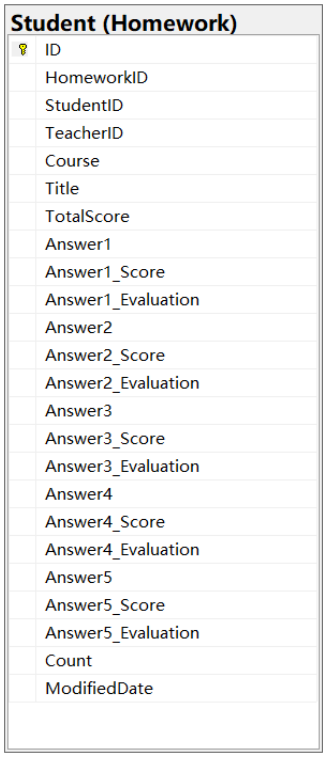
Design Table: (Bold text is the primary key)

Number	Column Name	Type	Allow Null values
1	**ID**	int	×
2	HomeworkID	int	×
3	StudentID	nvarchar(256)	×
4	TeacherID	nvarchar(256)	×
5	Course	nvarchar(50)	√
6	Title	nvarchar(100)	√
7	TotalScore	float	√
8	Answer1	nvarchar(MAX)	√
9	Answer1_Score	nvarchar(MAX)	√
10	Answer1_Evaluation	float	√
11	Answer2	nvarchar(MAX)	√
12	Answer2_Score	float	√
13	Answer2_Evaluation	nvarchar(MAX)	√
14	Answer3	nvarchar(MAX)	√
15	Answer3_Score	float	√
16	Answer3_Evaluation	nvarchar(MAX)	√
17	Answer4	nvarchar(MAX)	√
18	Answer4_Score	float	√
19	Answer4_Evaluation	nvarchar(MAX)	√
20	Answer5	nvarchar(MAX)	√
21	Answer5_Score	float	√
22	Answer5_Evaluation	nvarchar(MAX)	√
23	Count	int	√
24	ModifiedDate	datetime	√

Code:

```
CREATE TABLE [Homework].[Student](
    [ID] [int] NOT NULL,
    [HomeworkID] [int] NOT NULL,
    [StudentID] [nvarchar](256) NOT NULL,
    [TeacherID] [nvarchar](256) NOT NULL,
    [Course] [nvarchar](50) NULL,
    [Title] [nvarchar](100) NULL,
    [TotalScore] [float] NULL,
    [Answer1] [nvarchar](max) NULL,
    [Answer1_Score] [float] NULL,
    [Answer1_Evaluation] [nvarchar](max) NULL,
    [Answer2] [nvarchar](max) NULL,
    [Answer2_Score] [float] NULL,
    [Answer2_Evaluation] [nvarchar](max) NULL,
    [Answer3] [nvarchar](max) NULL,
    [Answer3_Score] [float] NULL,
    [Answer3_Evaluation] [nvarchar](max) NULL,
    [Answer4] [nvarchar](max) NULL,
    [Answer4_Score] [float] NULL,
    [Answer4_Evaluation] [nvarchar](max) NULL,
    [Answer5] [nvarchar](max) NULL,
    [Answer5_Score] [float] NULL,
    [Answer5_Evaluation] [nvarchar](max) NULL,
    [Count] [int] NULL,
    [ModifiedDate] [datetime] NULL,
 CONSTRAINT [PK_Student_1] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTI
AL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
```

Structure: This is the structure of Homework.Student and the yellow key is PRIMARY KEY

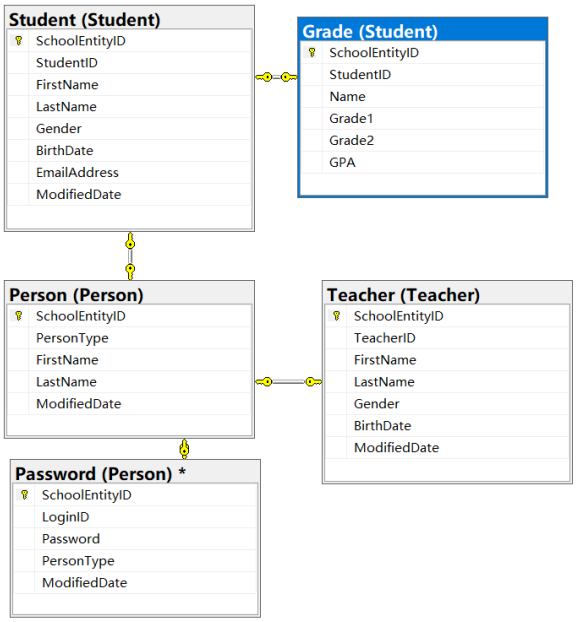


Picture 8

Combine Tables

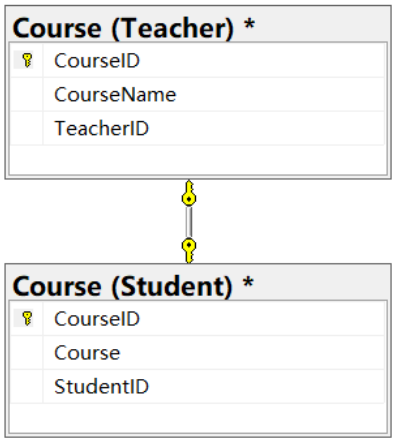
There have three tables need combined.

Combine Table1: Table Person.Person, Teacher.Teacher, Student.Student, Person.Password, Student.Grade are combined.



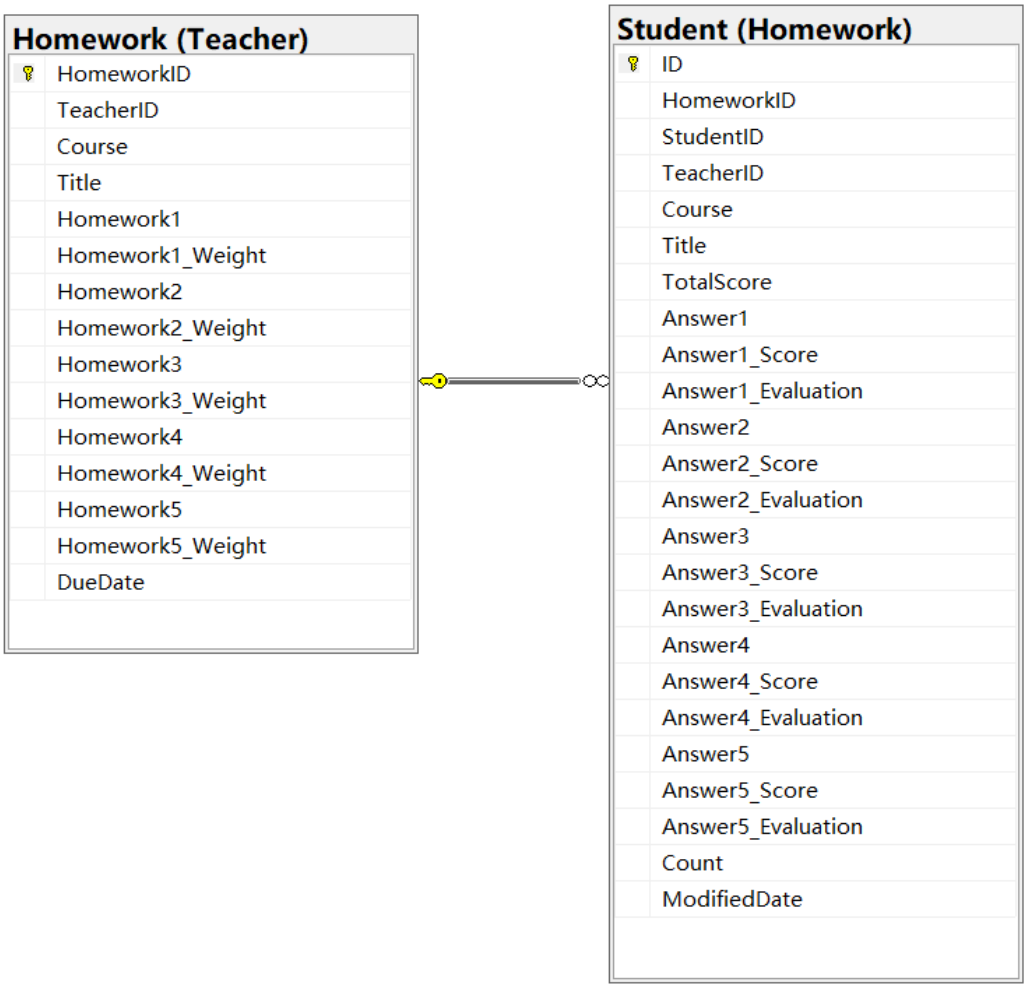
Picture 9

Combine Table2: Table Teacher.Course and Student.Course are combined.



Picture 10

Combine Table3: Teacher.Homework and Homework.Student are combined.



Picture 11: two tables are combined they have already connet

ASP.NET Design

Create ASP.NET

Firstly, Open the Visual Studio 2019. Secondly, Create ASP.NET Web application (.NET Framework) in C# language then Select Add -> New Item -> Web form , input name and add it create a new web.

Create Website

Website name: Login.aspx

Description:

This page is a login page, teachers and students can use their ID(StudentID, TeacherID) and password to login Online Grading System.

Realized function:

1. Get username and password from Front-end.

```
string username = TextBox1.Text;
string password = TextBox2.Text;
```

2. By change the RadioButtonList to realize the different login interfaces for students and teachers.

```
radSale.Equals("0") == true
```

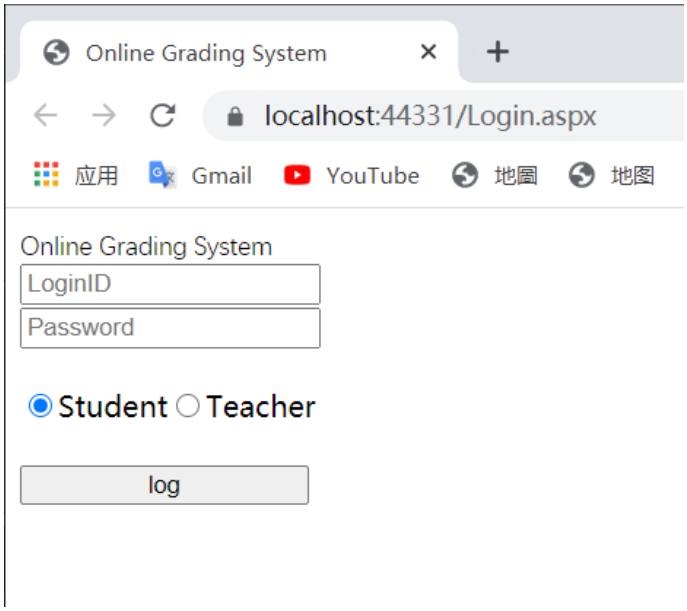
3. Data read from the database and data read from the Front-end match each other.

```
    SqlConnection con = new SqlConnection(@"Server=(local);Data Source = LAP
TOP-BSO7LT4F\SQLEXPRESS;Initial Catalog=whb;Integrated Security=True");
    SqlCommand cmd = new SqlCommand(sql);
    cmd.Connection = con;
    con.Open();
    SqlDataReader data = cmd.ExecuteReader();
```

4. Page jump with username.

```
Response.Redirect("TeacherPage_Home.aspx?name="+username);
```

GUI design:



Picture 12: Login in interface

Website name: TeacherPage_Home.aspx

Description:

This page is Teacher page, teachers can Modify student information and it is also can jump to Student Account Creation Page, Management of the Assignments Page, Assignment Marking Page and Management of Student Course Page.

Realized function:

1. Use sqlDataSource to connect to the database and use GridView to display it.

```
<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="
False" DataKeyNames="…" DataSourceID="SqlDataSource1" CellPadding
="…" ForeColor="…" GridLines="None"></asp:GridView>

<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionS
tring="…" OnSelecting="…" SelectCommand="…" ConflictDetection="…"
 DeleteCommand="…">
```

2. Teachers can Update student information through GridView and update to the database.

```
<UpdateParameters>
    <asp:Parameter Name="StudentID" Type="String" />
    <asp:Parameter Name="FirstName" Type="String" />
    …
</UpdateParameters>
```

3. Teachers can Delete student information through GridView and update to the database.

```
<DeleteParameters>
    <asp:Parameter Name="original_SchoolEntityID" Type="Int32" />
    <asp:Parameter Name="original_StudentID" Type="String" />
```

```
    …
</DeleteParameters>
```

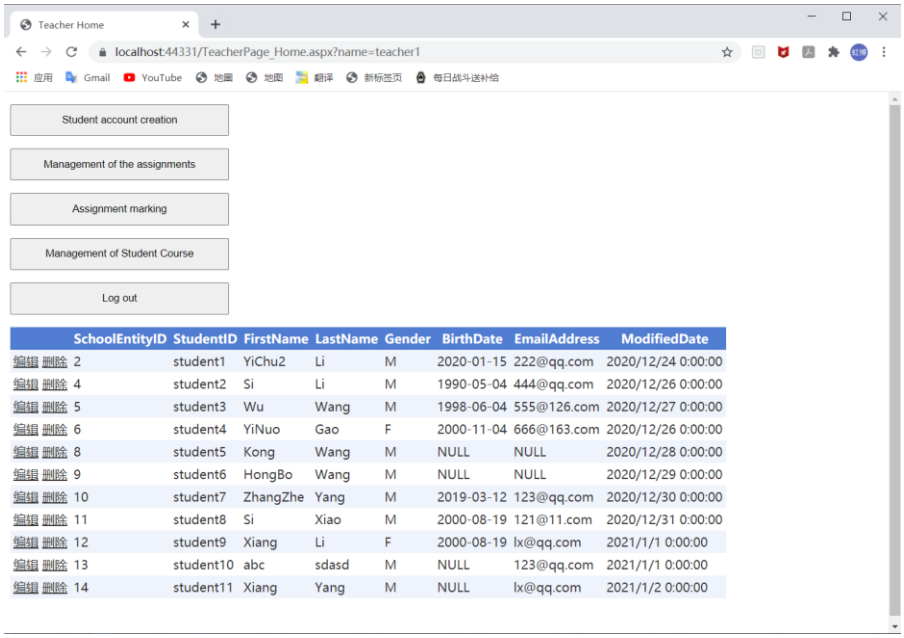
4. Page jump with username.

```
Response.Redirect("…/aspx?name="+username);
```

5. Logout button.

```
Response.Redirect("Login.aspx");
```

GUI design:



Picture 13: Teacher Page: Teacher Home

Website name: TeacherPage_StudentAccountCreation.aspx

Description:

This page is Teacher page. The teacher can create an account for each student. The teacher goes to a website and enters the student information into the database. The student information can be the student name, email address and birthday, etc. Student ID will be automatically generated, of course, if the teacher or student wants to change it can also be changed in the GridView.

Realized function:

1. Get user input from textbox.

```
string ... = (...).Text;
```

2. StudentID and SchoolEntityID is automatically generated.

```
//Get max SchoolEntityID
int maxid;
sql = "SELECT   MAX(SchoolEntityID) AS MAXID FROM   Person.Person";
SqlCommand cmd = new SqlCommand(sql);
cmd.Connection = con;
```

```
maxid = (int)cmd.ExecuteScalar() + 1;//The new student SchoolEntityID
```

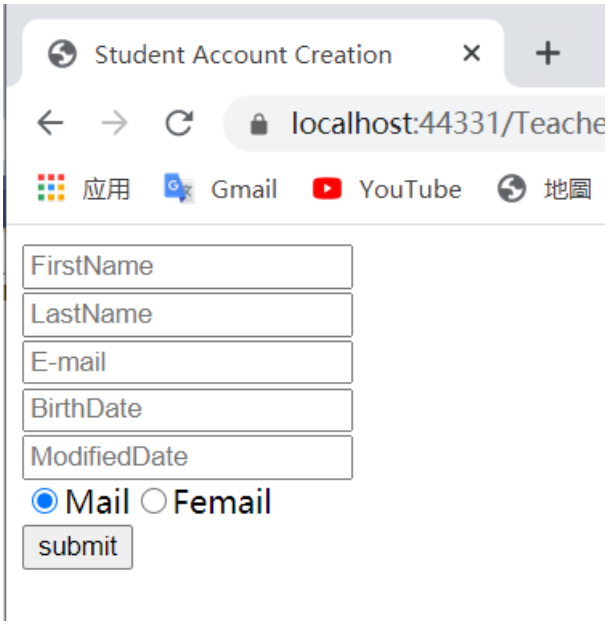
3. At the same time insert the data in Person.Person, Student.Student, Person.Password.

```
//Person.Person
sql = "INSERT INTO Person.Person(...) values('"+...+"','"+...+"')";
SqlCommand cmd_insert_pp = new SqlCommand(sql);
cmd_insert_pp.Connection = con;
cmd_insert_pp.ExecuteNonQuery();
//Student.Student
...
//Person.Password
...
```

4. If successfully inserted, there will be a success prompt.

```
Response.Write("<script>alert('success')</script>");
```

GUI design:



Picture 14: Teacher Page: Student Account Creation

Website name: TeacherPage_ManagementOfTheAssignment.aspx

Description:

This page is Teacher page. The teacher can create an assignment to the students. An assignment has several tasks or questions, for each question there is a weight associated with it. The teacher can go to a website and enters the questions with the title and the weight for each assignment.

Realized function:

1. Teacher can click the calendar to select a assignment deadline date.

```
string str = Calendar1.SelectedDate.ToString("yyyy-MM-dd");
```

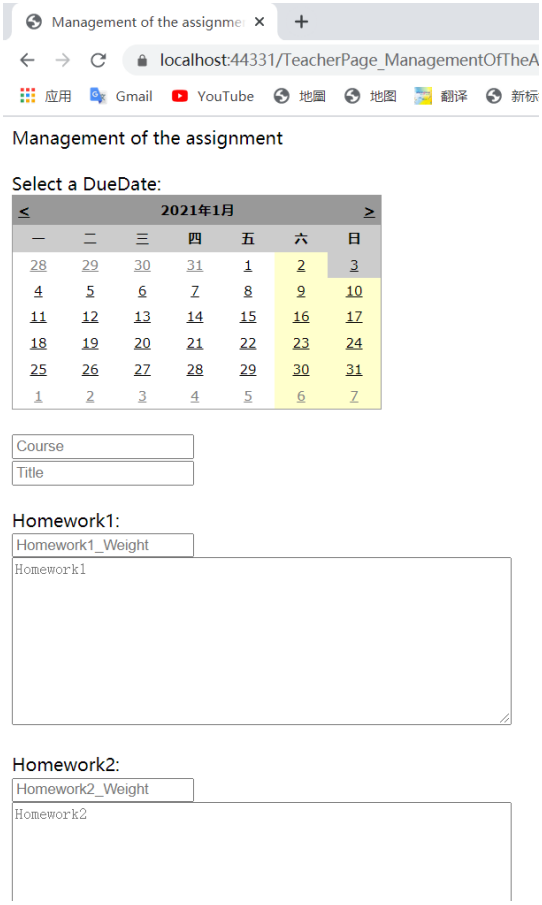
2. Get user input from textbox.

```
string ... = (...).Text;
```

3. Insert the assignment and course into SQL table Teacher.Homework and Teacher.Course.

```
string sql;
sql = "INSERT INTO Teacher.Course(...) values('" + ... + "','" + ... + "')";
SqlCommand cmd = new SqlCommand(sql);
cmd.Connection = con; cmd.ExecuteNonQuery();
```

GUI design:



Picture 15: Teacher Page: Management of the assignment

Website name: TeacherPage_AssignmentMarking.aspx

Description:

This page is Teacher page. When the students submit their assignment, the teacher can give a grade as well as comments for each question or assignment.

Realized function:

1. Use sqlDataSource to connect to the database and use GridView to display it.

```
<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="
False" DataKeyNames="..." DataSourceID="SqlDataSource1" CellPadding
="..." ForeColor="..." GridLines="None"></asp:GridView>
```

```
<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionS
tring="..." OnSelecting="..." SelectCommand="..." ConflictDetection="..."
 DeleteCommand="...">
```

2. Teacher can click Details then can check student Assignment.

```
<asp:ButtonField CommandName="Details" Text="Details" />
```

```
protected void btnOpenClick(object sender, GridViewCommandEventArgs e)
{
    if (e.CommandName == "Details")
    {
        int RowIndex = Convert.ToInt32(e.CommandArgument);
        string ID = GridView1.Rows[RowIndex].Cells[0].Text;
        string HomeworkID = GridView1.Rows[RowIndex].Cells[1].Text;
        string StudentID = GridView1.Rows[RowIndex].Cells[2].Text;
        string userName = Request.QueryString["name"];
        Response.Redirect("(...).aspx");
    }
}
```

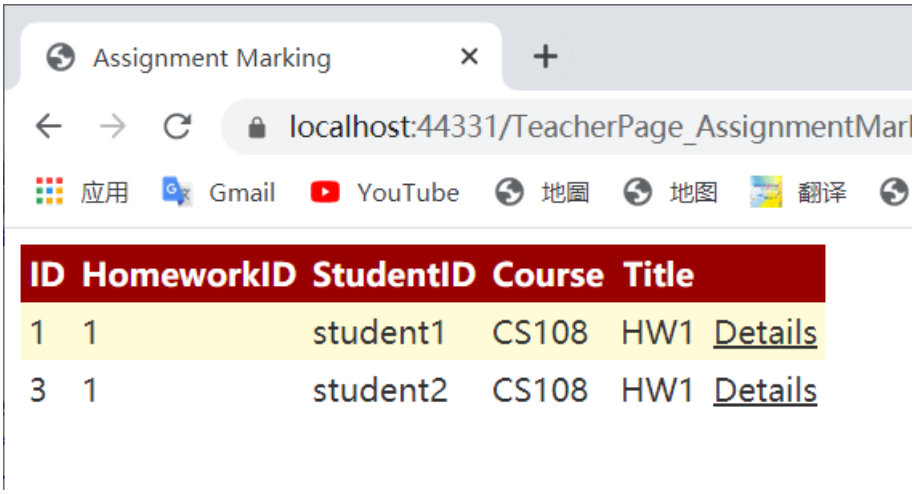
3. Teacher can only see your student Assignment.

```
<asp:QueryStringParameter Name="TeacherID" QueryStringField="name
" Type="String" />
```

4. Page jump to Homework_Teacher.aspx with username, StudentID and HomeworkID.

```
Response.Redirect("Homework_Teacher.aspx?name=" + ...
+"&StudentID="+ ... + "&ID=" + ID+ "&HomeworkID="+...);
```

GUI design:



Picture 16: Teacher Page: Management of the assignment

Website name: TeacherPage_ManagementOfStudentCourse.aspx

Description:

This page is Teacher page. Teacher can add student who need attend class. Then Student can find

the Homework assigned by the teacher in their summary assignment Page.

Realized function:

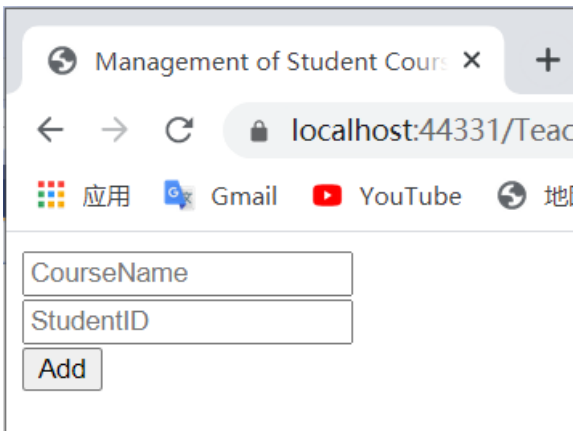
1. Get user input from textbox.

```
string ... = (...).Text;
```

2. Insert the student course into SQL table Student.Course.

```
string sql;
sql = "INSERT INTO Student.Course(...) values('" + ... + "','" + ... + "')";
SqlCommand cmd = new SqlCommand(sql);
cmd.Connection = con; cmd.ExecuteNonQuery();
```

GUI design:



Picture 17: Teacher Page: Management of Student Course

Website name: StudentPage_Home.aspx

Description:

This page is Student page. Students can Modify themselves information and it is also can jump to Assignment submission Page, Summary of the assignment Page and Logout Page.

Realized function:

1. Use sqlDataSource to connect to the database and use GridView to display it.

```
<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="
False" DataKeyNames="…" DataSourceID="SqlDataSource1" CellPadding
="…" ForeColor="…" GridLines="None"></asp:GridView>

<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionS
tring="…" OnSelecting="…" SelectCommand="…" ConflictDetection="…"
 DeleteCommand="…">
```

2. Students can Update himself information through GridView and update in the database.

```
<UpdateParameters>
    <asp:Parameter Name="StudentID" Type="String" />
    <asp:Parameter Name="FirstName" Type="String" />
```

```
    …
</UpdateParameters>
```

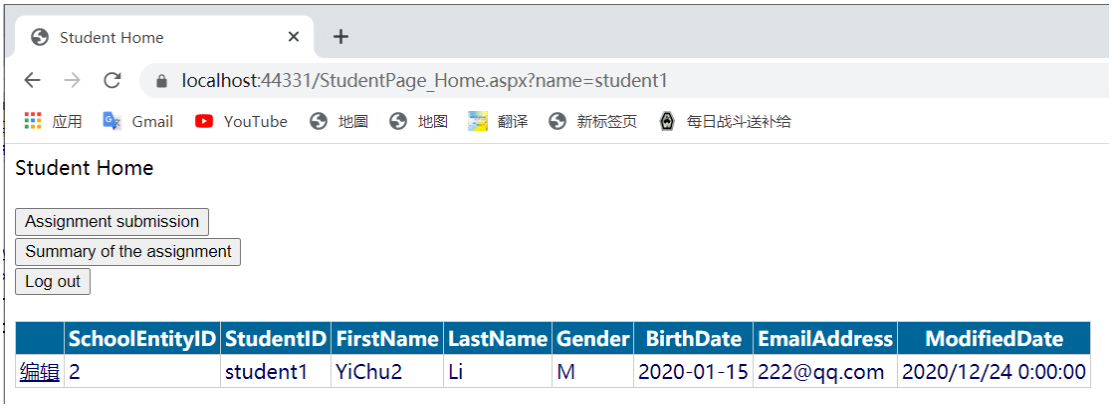
3. Students can only see themselves student information.

```
<asp:QueryStringParameter Name="StudentID" QueryStringField="name
" Type="String" />
```

4. Page jump to Homework_Teacher.aspx with username.

```
Response.Redirect("(…).aspx?name=" + ...);
```

GUI design:



Picture 18: Student Page: Student Home

Website name: StudentPage_AssignmentSubmission.aspx

Description:

This page is Student page. The student can find their course in this page.

Realized function:

1. Use sqlDataSource to connect to the database and use GridView to display it.

```
<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="
False" DataKeyNames="…" DataSourceID="SqlDataSource1" CellPadding
="…" ForeColor="…" GridLines="None"></asp:GridView>

<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionS
tring="…" OnSelecting="…" SelectCommand="…" ConflictDetection="…"
 DeleteCommand="…">
```

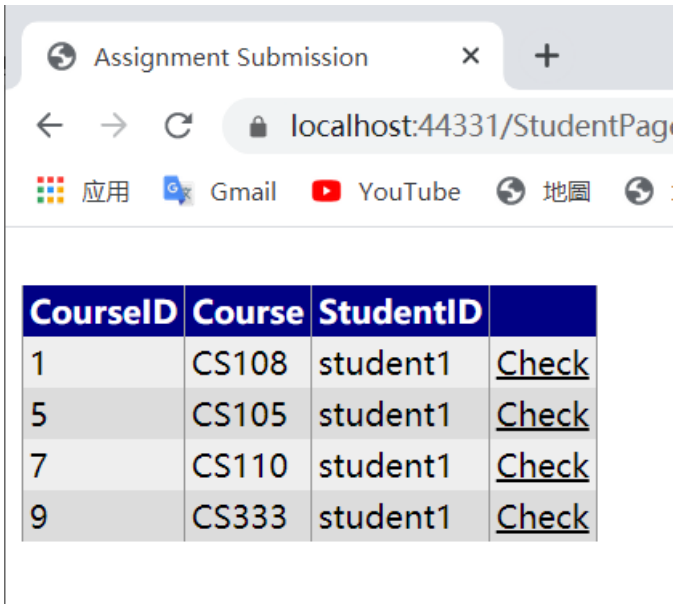
2. Students can only see themselves course information.

```
<asp:QueryStringParameter Name="StudentID" QueryStringField="name
" Type="String" />
```

3. Page jump to Homework_Course.aspx with username and courseName.

```
Response.Redirect("(…).aspx?name=" + ... +"&courseName="+...);
```

GUI design:



Picture 19: Student Page: Assignment Submission

Website name: StudentPage_SummaryOfTheAssignment.aspx

Description:

This page is Student page. The students can see the grade and the comments of their assignments. The system will calculate the average marks and the GPA and display them to the students.

Realized function:

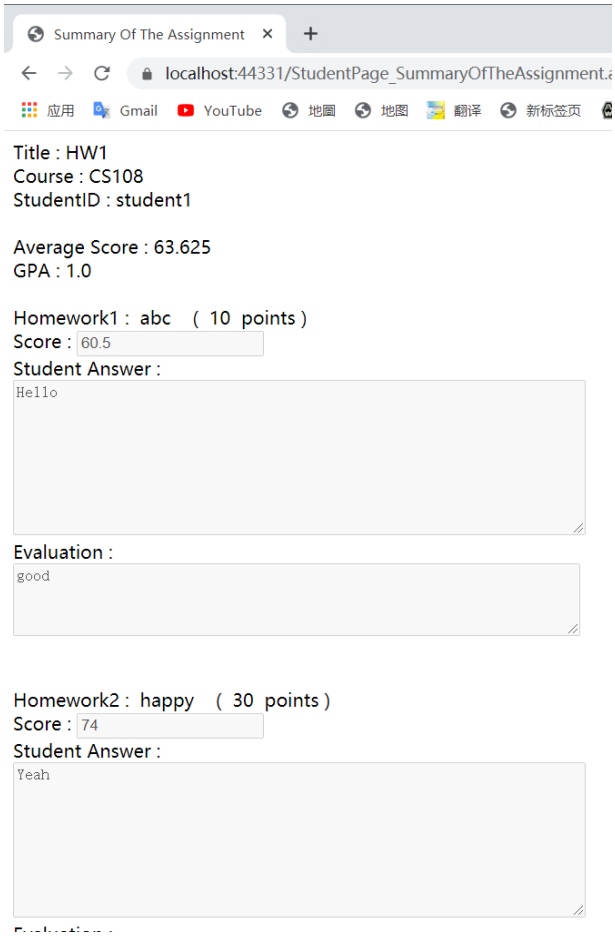
1. Student cannot change the textbox information .

```
disabled="disabled"
```

2. Use Label to change Information

```
[Label].Text =(...);
```

GUI design:



Picture 20: Student Page: Summary of Assignment

Website name: StudentPage_Check.aspx

Description:

This page is Student page.

This page is Student page. In this page you can get the homework grade and details button can jump to a new page which you can see details about your homework, like weight, GPA and teacher's evaluations.

Realized function:

1. Use sqlDataSource to connect to the database and use GridView to display it.

```
<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="
False" DataKeyNames="…" DataSourceID="SqlDataSource1" CellPadding
="…" ForeColor="…" GridLines="None"></asp:GridView>

<asp:SqlDataSource ID="SqlDataSource1" runat="server" ConnectionS
tring="…" OnSelecting="…" SelectCommand="…" ConflictDetection="…"
 DeleteCommand="…">
```

2. Students can only see themselves Homework grade information.

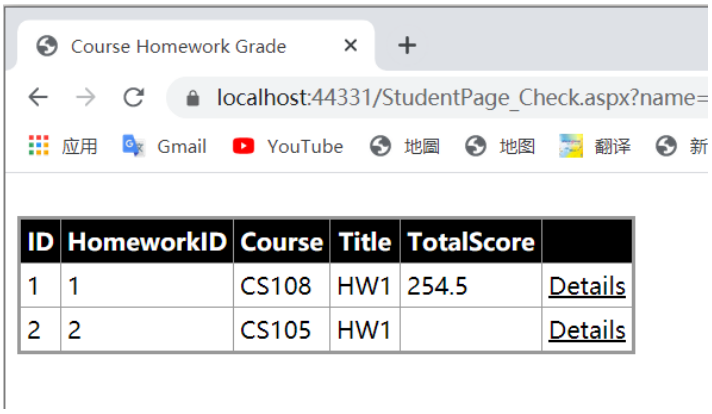
```
<asp:QueryStringParameter Name="StudentID" QueryStringField="name
" Type="String" />
```

3. Students can click Details then can check details about Assignment.

```
<asp:ButtonField CommandName="Details" Text="Details" />
```

```
protected void btnOpenClick(object sender, GridViewCommandEventArgs e)
{
    if (e.CommandName == "Details")
    {
        int RowIndex = Convert.ToInt32(e.CommandArgument);
        string ID = GridView1.Rows[RowIndex].Cells[0].Text;
        string HomeworkID = GridView1.Rows[RowIndex].Cells[1].Text;
        string StudentID = GridView1.Rows[RowIndex].Cells[2].Text;
        string userName = Request.QueryString["name"];
        Response.Redirect("(...).aspx");
    }
}
```

GUI design:



ID	HomeworkID	Course	Title	TotalScore	
1	1	CS108	HW1	254.5	Details
2	2	CS105	HW1		Details

Picture 21: Student Page: Course Homework Grade

Website name: Homework_Course.aspx

Description:

This page is Student page. The student can find how many homework do the teacher leave in the same class, then you can click Details to jump to a new Page which you can do your homework.

Realized function:

1. Use sqlDataSource to connect to the database and use GridView to display it.

```
<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="
False" DataKeyNames="…" DataSourceID="SqlDataSource1" CellPadding
="…" ForeColor="…" GridLines="None"></asp:GridView>
```

```
<asp:SqlDataSource ID=”SqlDataSource1” runat=”server” ConnectionS
tring=”…” OnSelecting=”…” SelectCommand=”…” ConflictDetection=”…”
 DeleteCommand=”…”>
```

2. Students can only see themselves Homework grade information.

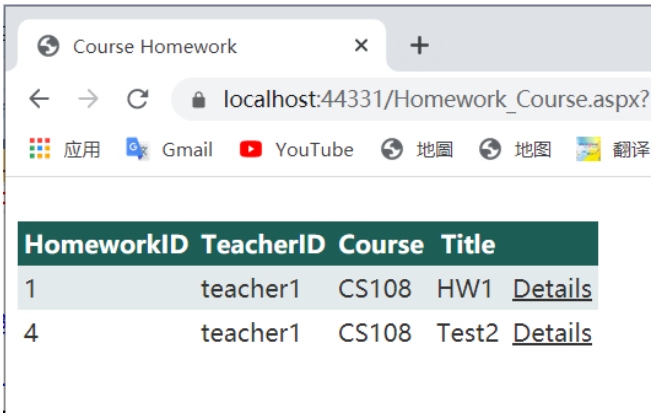
```
<asp:QueryStringParameter Name="StudentID" QueryStringField="name
" Type="String" />
```

3. Students can click Details then can check details about Assignment.

```
<asp:ButtonField CommandName="Details" Text="Details" />
```

```
protected void btnOpenClick(object sender, GridViewCommandEventArgs e)
{
    if (e.CommandName == "Details")
    {
        int RowIndex = Convert.ToInt32(e.CommandArgument);
        string ID = GridView1.Rows[RowIndex].Cells[0].Text;
        string HomeworkID = GridView1.Rows[RowIndex].Cells[1].Text;
        string StudentID = GridView1.Rows[RowIndex].Cells[2].Text;
        string userName = Request.QueryString["name"];
        Response.Redirect("(...).aspx");
    }
}
```

GUI design:



Picture 22: Student Page: Course Homework

Website name: Homework_Student.aspx

Description:

This page is Student page. The students can answer the questions online

Realized function:

1. Use Label to change Information

```
[Label].Text =(...);
```

2. Get the Homework title and homework weight information from SQL table Teacher.Homework

```
protected void getHomeWork1()
{
    string sql;
    sql = "SELECT Homework(...)) FROM Teacher.Homework WHERE HomeworkID = " + ...;
    SqlCommand cmd = new SqlCommand(sql);
    cmd.Connection = con;
    homeWork1_Text = cmd.ExecuteScalar().ToString();
}
```

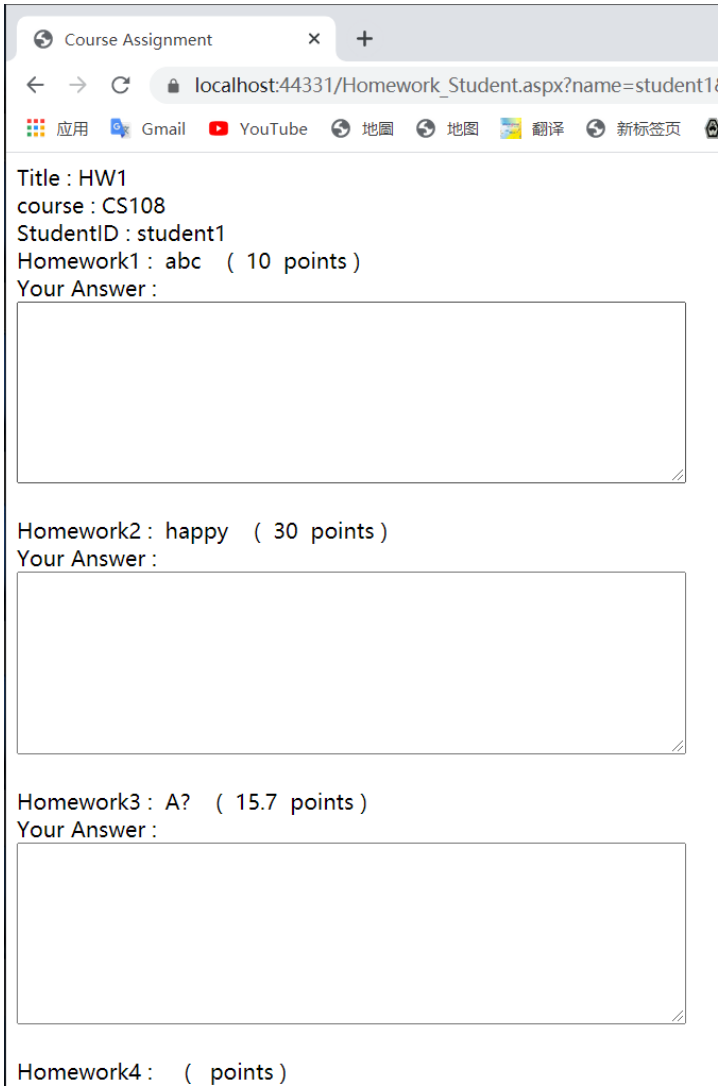
3. Insert the student answer into SQL Table Homework.Student

```
if (Homework1Answer != null)
{
    sql = "UPDATE [Homework].[Student] SET Answer1 ='" + Homework
1Answer + "' WHERE ID = " + _HomeworkRow + " ";
    SqlCommand cmd_HW1 = new SqlCommand(sql);
    cmd_HW1.Connection = con;
    cmd_HW1.ExecuteNonQuery();
}
```

4. Can automatically arrange the assignment ID of each student

```
protected void HomeworkID()
{
    int HomeworkRow;
    string sql;
    sql = "SELECT   COUNT(*) FROM Homework.Student ";
    SqlCommand cmd_sr = new SqlCommand(sql);
    cmd_sr.Connection = con;
    HomeworkRow = (int)cmd_sr.ExecuteScalar() + 1;
    search_HomeworkRow = (HomeworkRow-1).ToString();
    _HomeworkRow = HomeworkRow.ToString();
}
```

GUI design:



Picture 23: Student Page: Course Assignment

Website name: Homework_Teacher.aspx

Description:

This page is Teacher page. When the students submit their assignment, the teacher can give a grade as well as comments for each question or assignment.

Realized function:

1. Use Label to change Information

```
[Label].Text =(...);
```

2. Get the Homework title, homework weight and Student Answer information from SQL table Homework.Student

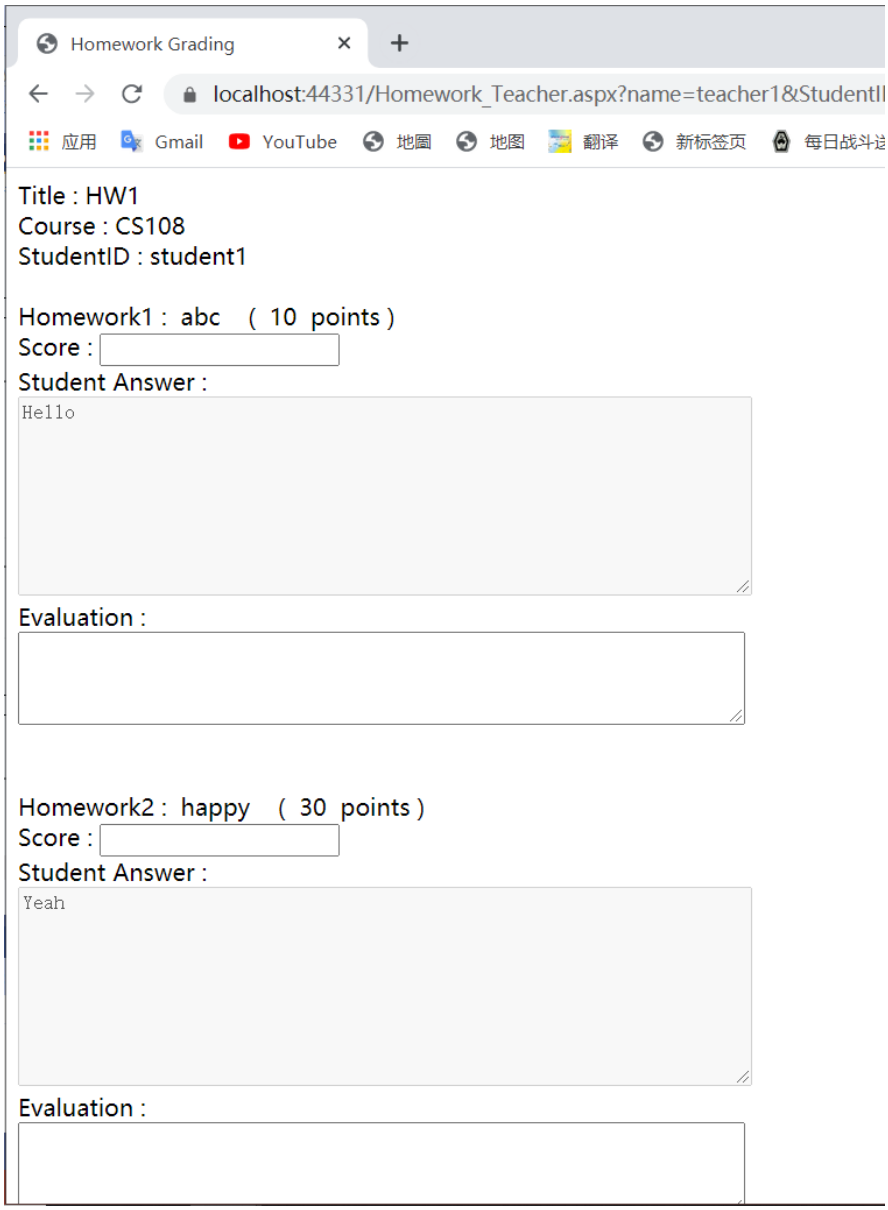
```
protected void getHomeWork1()
{
    string sql;
    sql = "SELECT Homework(...)) FROM Homework.Student WHERE HomeworkID = " + ...;
```

```
    SqlCommand cmd = new SqlCommand(sql);
    cmd.Connection = con;
    homeWork1_Text = cmd.ExecuteScalar().ToString();
}
```

3. Insert the evaluation and point into SQL Table Homework.Student

```
    sql = "UPDATE [Homework].[Student] SET Answer1_Score ='" + Sc
ore + "' WHERE ID = " + ID + " ";
    SqlCommand cmd_HW1 = new SqlCommand(sql);
    cmd_HW1.Connection = con;
    cmd_HW1.ExecuteNonQuery();
```

GUI design:



Picture 24: Student Page: Homework Grading

System Interface operation method

Teacher operation

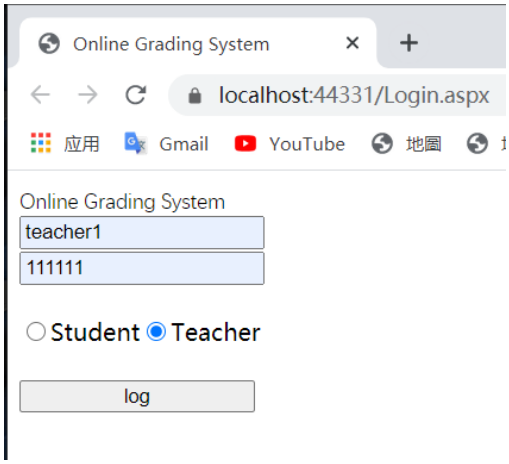
Introduction

In this Online Grading System, the teacher has six pages. Teacher can creation student count, management of the assignment, Assignment marking and Management of Student Course.

Demonstrate

First Page of Teacher:

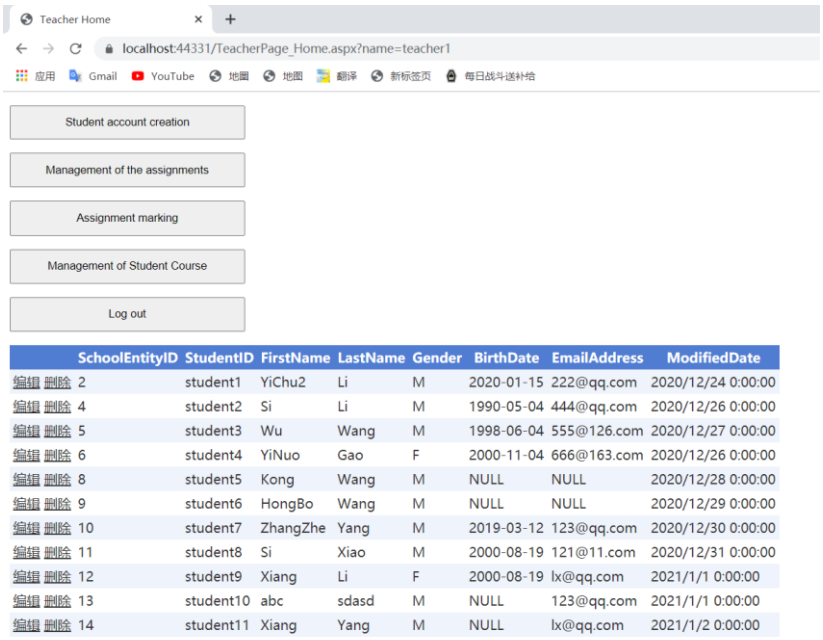
This the first page of teacher, Login interface. Teacher can input 'TeacherID' and 'Password' to login, then teacher need choose 'Teacher' button and click log button. After that teacher can get into the system.



Picture 1: Online Grading System login interface

Second Page of Teacher:

After teacher success login the system, the page will be look like this. Teacher can change student information in the table. If the teacher clicks the last button [Log out], the user will be logout.



	SchoolEntityID	StudentID	FirstName	LastName	Gender	BirthDate	EmailAddress	ModifiedDate
编辑 删除	2	student1	YiChu2	Li	M	2020-01-15	222@qq.com	2020/12/24 0:00:00
编辑 删除	4	student2	Si	Li	M	1990-05-04	444@qq.com	2020/12/26 0:00:00
编辑 删除	5	student3	Wu	Wang	M	1998-06-04	555@126.com	2020/12/27 0:00:00
编辑 删除	6	student4	YiNuo	Gao	F	2000-11-04	666@163.com	2020/12/26 0:00:00
编辑 删除	8	student5	Kong	Wang	M	NULL	NULL	2020/12/28 0:00:00
编辑 删除	9	student6	HongBo	Wang	M	NULL	NULL	2020/12/29 0:00:00
编辑 删除	10	student7	ZhangZhe	Yang	M	2019-03-12	123@qq.com	2020/12/30 0:00:00
编辑 删除	11	student8	Si	Xiao	M	2000-08-19	121@11.com	2020/12/31 0:00:00
编辑 删除	12	student9	Xiang	Li	F	2000-08-19	lx@qq.com	2021/1/1 0:00:00
编辑 删除	13	student10	abc	sdasd	M	NULL	123@qq.com	2021/1/1 0:00:00
编辑 删除	14	student11	Xiang	Yang	M	NULL	lx@qq.com	2021/1/2 0:00:00

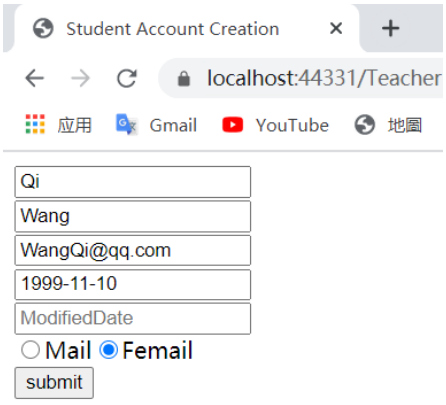
Picture2: Teacher Home Page

If the teacher clicks the last button [edit], the teacher can change student information.

	SchoolEntityID	StudentID	FirstName	LastName	Gender	BirthDate	EmailAddress	ModifiedDate
编辑 删除	2	student1	YiChu2	Li	M	2020-01-15	222@qq.com	2020/12/24 0:00:00
编辑 删除	4	student2	Si	Li	M	1990-05-04	444@qq.com	2020/12/26 0:00:00
编辑 删除	5	student3	Wu	Wang	M	1998-06-04	555@126.com	2020/12/27 0:00:00
更新 取消	6	student4	YiNuo	Gao	F	2000-11-04	666@163.com	2020/12/26 0:00:00
编辑 删除	8	student5	Kong	Wang	M	NULL	NULL	2020/12/28 0:00:00
编辑 删除	9	student6	HongBo	Wang	M	NULL	NULL	2020/12/29 0:00:00
编辑 删除	10	student7	ZhangZhe	Yang	M	2019-03-12	123@qq.com	2020/12/30 0:00:00
编辑 删除	11	student8	Si	Xiao	M	2000-08-19	121@11.com	2020/12/31 0:00:00
编辑 删除	12	student9	Xiang	Li	F	2000-08-19	lx@qq.com	2021/1/1 0:00:00
编辑 删除	13	student10	abc	sdasd	M	NULL	123@qq.com	2021/1/1 0:00:00
编辑 删除	14	student11	Xiang	Yang	M	NULL	lx@qq.com	2021/1/2 0:00:00
编辑 删除	15	student12	Qi	Wang	F	1999-11-10	WangQi@qq.com	2021/1/3 0:00:00

Third Page of Teacher:

If the teacher clicks the first button [student account creation], the page will be look like this.



Picture3: Teacher can add student information

Then teacher can input the student information into the textbox, choose the mail button or female button. Finally, click the button[submit] the page will look like this and information will add into SQL Table Person.Person, Person.Password, Student.Student

Picture4: The information is succeed adding into SQL

	SchoolEntityID	StudentID	FirstName	LastName	Gender	BirthDate	EmailAddress	ModifiedDate
1	2	student1	YiChu2	Li	M	2020-01-15	222@qq.com	2020-12-24 00:00:00.000
2	4	student2	Si	Li	M	1990-05-04	444@qq.com	2020-12-26 00:00:00.000
3	5	student3	Wu	Wang	M	1998-06-04	555@126.com	2020-12-27 00:00:00.000
4	6	student4	YiNuo	Gao	F	2000-11-04	666@163.com	2020-12-26 00:00:00.000
5	8	student5	Kong	Wang	M	NULL	NULL	2020-12-28 00:00:00.000
6	9	student6	HongBo	Wang	M	NULL	NULL	2020-12-29 00:00:00.000
7	10	student7	ZhangZhe	Yang	M	2019-03-12	123@qq.com	2020-12-30 00:00:00.000
8	11	student8	Si	Xiao	M	2000-08-19	121@11.com	2020-12-31 00:00:00.000
9	12	student9	Xiang	Li	F	2000-08-19	lx@qq.com	2021-01-01 00:00:00.000
10	13	student10	abc	sdasd	M	NULL	123@qq.com	2021-01-01 00:00:00.000
11	14	student11	Xiang	Yang	M	NULL	lx@qq.com	2021-01-02 00:00:00.000
12	15	student12	Qi	Wang	F	1999-11-10	WangQi@qq.com	2021-01-03 00:00:00.000

查询已成功执行。 LAPTOP-BSO7LT4F\SQLEXPRESS ... L

Picture5: The information is succeed adding into SQL Table Student.Student

结果 消息

	SchoolEntityID	LoginID	Password	PersonType	ModifiedDate
3	3	teacher2	333333	TC	2020-12-27 00:00:00.000
4	4	student2	444444	SD	2020-12-27 00:00:00.000
5	5	student3	555555	SD	2020-12-27 00:00:00.000
6	6	student4	666666	SD	2020-12-28 00:00:00.000
7	7	teacher3	777777	TC	2020-12-28 00:00:00.000
8	8	student5	888888	SD	2020-12-28 00:00:00.000
9	9	student6	999999	SD	2020-12-29 00:00:00.000
10	10	student7	000000	SD	2020-12-30 00:00:00.000
11	11	student8	111111	SD	2020-12-31 00:00:00.000
12	12	student9	222222	SD	2021-01-01 00:00:00.000
13	13	student10	333333	SD	2021-01-01 00:00:00.000
14	14	student11	444444	SD	2021-01-02 00:00:00.000
15	15	student12	555555	SD	2021-01-03 00:00:00.000

查询已成功执行。 LAP

Picture6: The information is succeed adding into SQL Table Person.Password

结果 消息

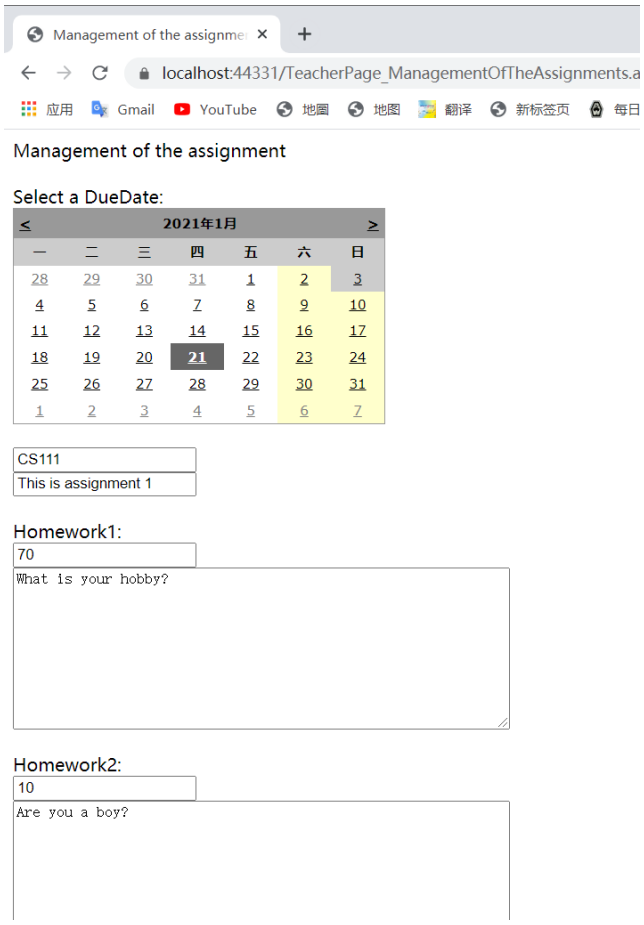
	SchoolEntityID	PersonType	FirstName	LastName	ModifiedDate
3	3	TC	San	Zhang	2020-12-26 00:00:00.000
4	4	SD	Si	Li	2020-12-26 00:00:00.000
5	5	SD	Wu	Wang	2020-12-26 00:00:00.000
6	6	SD	YiNuo	Gao	2020-12-26 00:00:00.000
7	7	TC	Qi	Zhang	2020-12-27 00:00:00.000
8	8	SD	Kong	Wang	2020-12-28 00:00:00.000
9	9	SD	HongBo	Wang	2020-12-29 00:00:00.000
10	10	SD	ZhangZhe	Yang	2020-12-30 00:00:00.000
11	11	SD	Si	Xiao	2020-12-31 00:00:00.000
12	12	SD	Xiang	Li	2021-01-01 00:00:00.000
13	13	SD	abc	sdasd	2021-01-01 00:00:00.000
14	14	SD	Xiang	Yang	2021-01-02 00:00:00.000
15	15	SD	Qi	Wang	2021-01-03 00:00:00.000

查询已成功执行。

Picture7: The information is succeed adding into SQL Table Person.Person

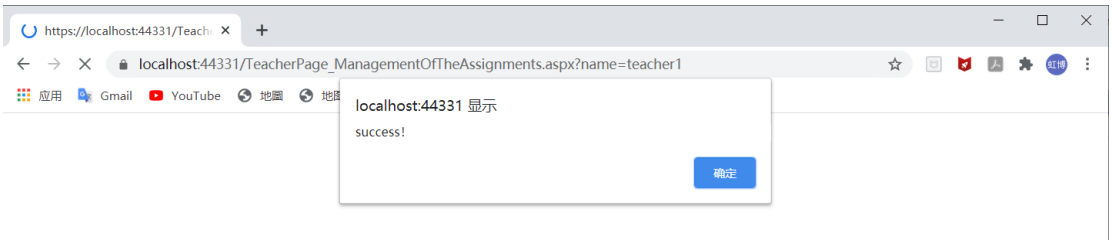
Fourth Page of Teacher:

If the teacher clicks the second button [Management of the assignments], the page will be look like this.

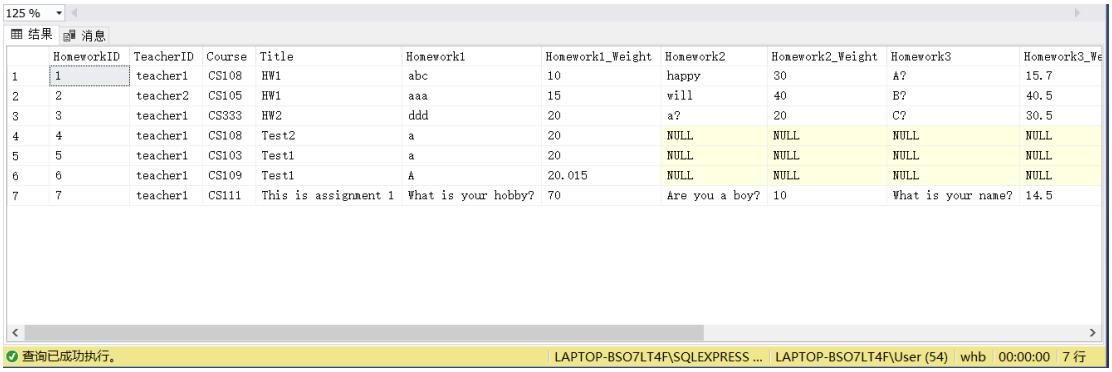


Picture8: The page of management of the assignment

Then teacher can input the assignment information into the textbox and select a due date in the calendar. Finally, click the button[create] the page will look like this and assignment information will add into SQL Table Teacher.Homework



Picture9: The information is succeed adding into SQL

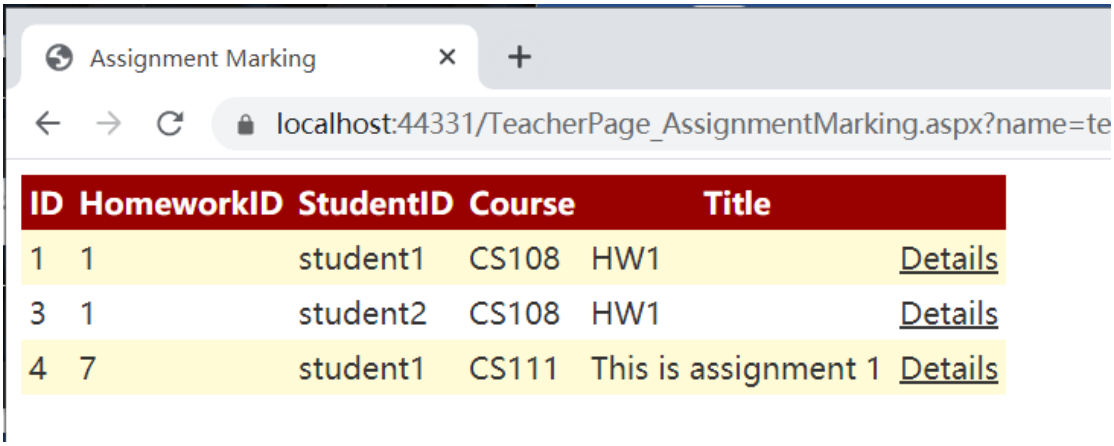


	HomeworkID	TeacherID	Course	Title	Homework1	Homework1_Weight	Homework2	Homework2_Weight	Homework3	Homework3_We
1	1	teacher1	CS108	HW1	abc	10	happy	30	A?	15.7
2	2	teacher2	CS105	HW1	aaa	15	will	40	B?	40.5
3	3	teacher1	CS333	HW2	ddd	20	a?	20	C?	30.5
4	4	teacher1	CS108	Test2	a	20	NULL	NULL	NULL	NULL
5	5	teacher1	CS103	Test1	a	20	NULL	NULL	NULL	NULL
6	6	teacher1	CS109	Test1	A	20.015	NULL	NULL	NULL	NULL
7	7	teacher1	CS111	This is assignment 1	What is your hobby?	70	Are you a boy?	10	What is your name?	14.5

Picture10: The information is succeed adding into SQL Table Teacher.Homework

Fifth Page of Teacher:

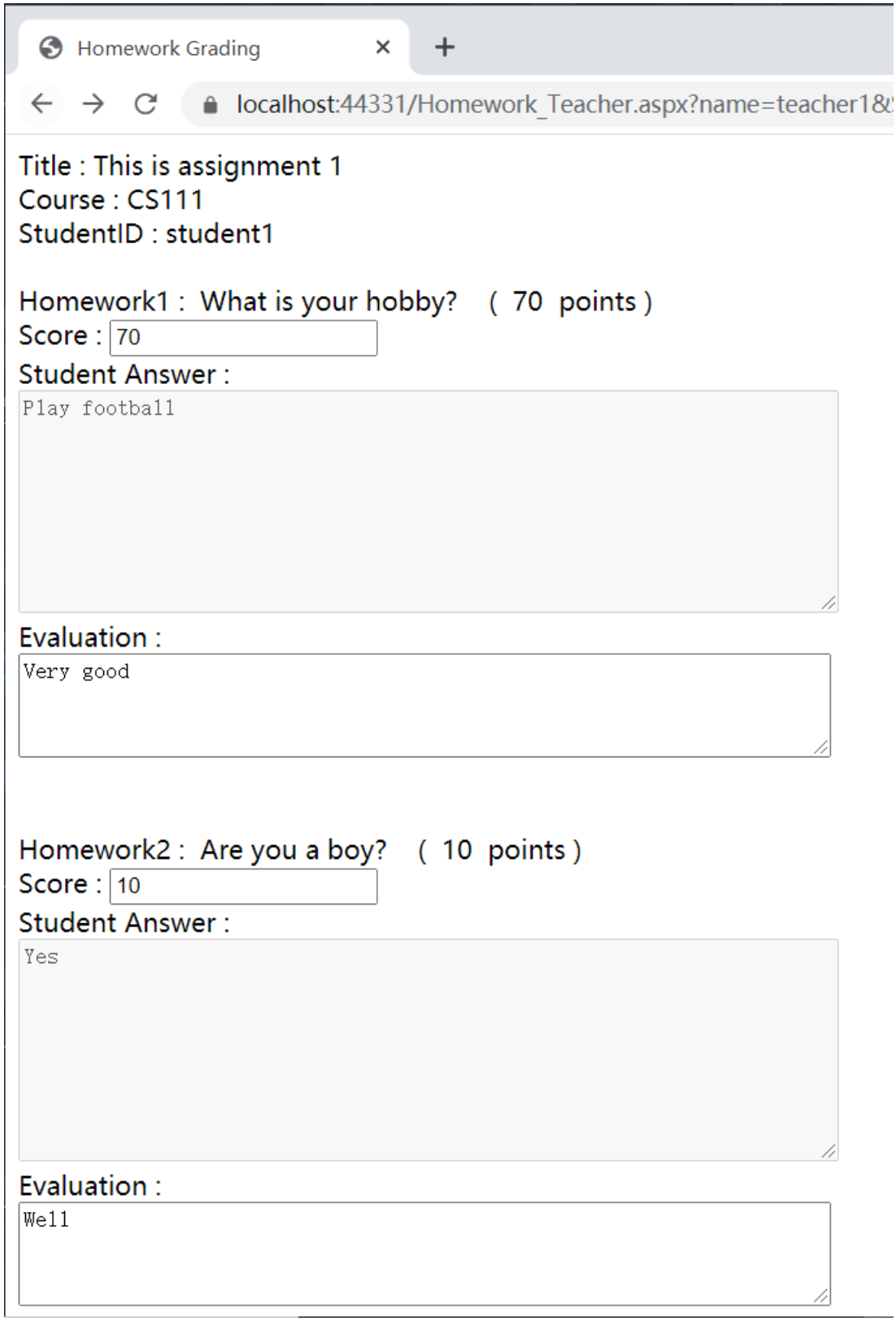
If the teacher clicks the third button [Assignment marking], the page will be look like this.



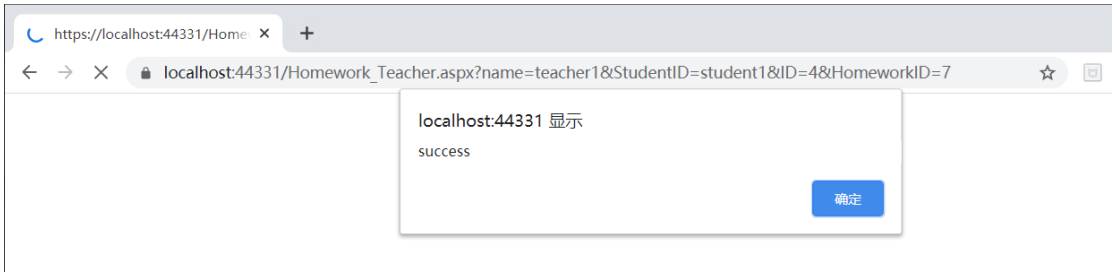
Picture8: Teacher can click Details to mark assignment

Teacher can click Details to mark assignment. Then, click the button[Upload] the grade and

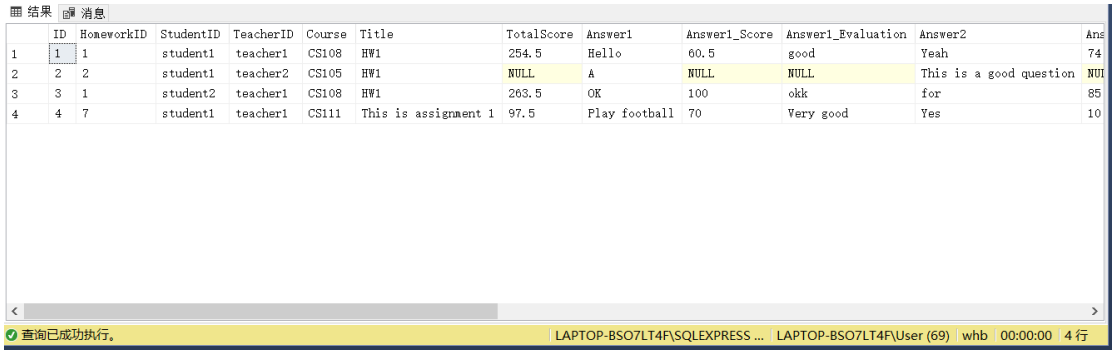
evaluation information will add into SQL Table Homework.Student



Picture9: Teacher marks assignment and evaluate to student



Picture9: Teacher upload the score and evaluation successfully

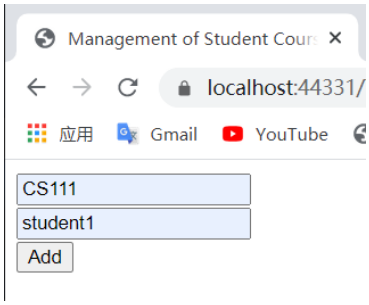


	ID	HomeworkID	StudentID	TeacherID	Course	Title	TotalScore	Answer1	Answer1_Score	Answer1_Evaluation	Answer2	Ans
1	1	1	student1	teacher1	CS108	HW1	254.5	Hello	60.5	good	Yeah	74
2	2	2	student1	teacher2	CS105	HW1	NULL	A	NULL	NULL	This is a good question	NUI
3	3	1	student2	teacher1	CS108	HW1	263.5	OK	100	okk	for	85
4	4	7	student1	teacher1	CS111	This is assignment 1	97.5	Play football	70	Very good	Yes	10

Picture10: The information is succeed adding into SQL Table Homework.Student

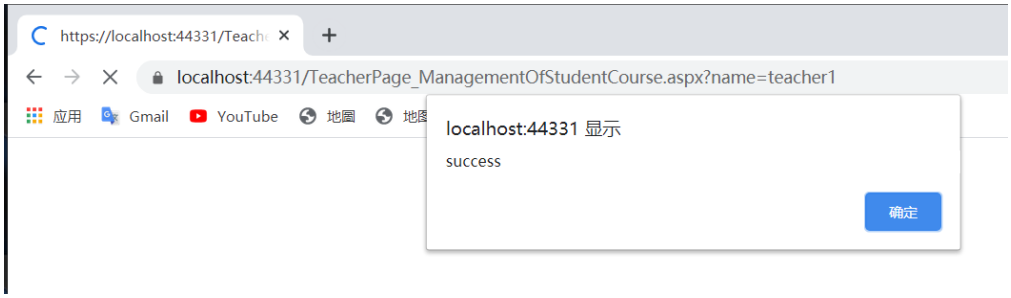
Sixth Page of Teacher:

If the teacher clicks the fourth button [Management of Student Course], the page will be look like this.

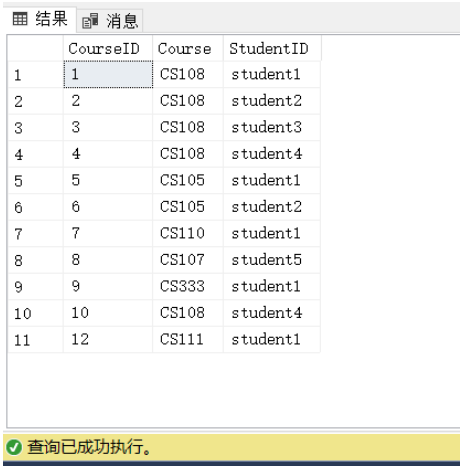


Picture 11: Teacher is adding student1 to CS111 course

Then teacher can input the course and student is into the textbox. Finally, click the button[add] the page will look like this and assignment information will add into SQL Table Student.Course



Picture 12: The information is succeed adding into SQL



	CourseID	Course	StudentID
1	1	CS108	student1
2	2	CS108	student2
3	3	CS108	student3
4	4	CS108	student4
5	5	CS105	student1
6	6	CS105	student2
7	7	CS110	student1
8	8	CS107	student5
9	9	CS333	student1
10	10	CS108	student4
11	12	CS111	student1

Picture13: The information is succeed adding into SQL Table Student.Course

Student operation

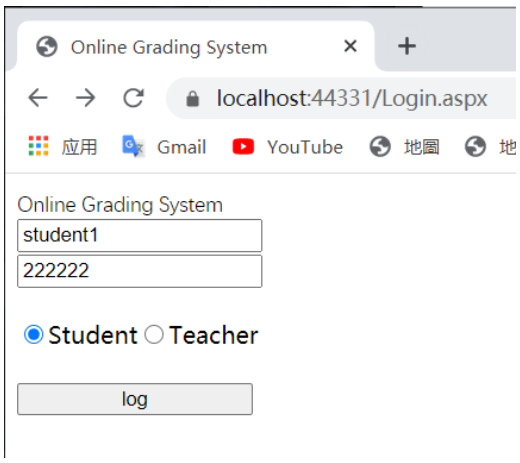
Introduction

In this Online Grading System, the Student has four pages. Student can assignment submission and summary of the assignment.

Demonstrate

First Page of Student:

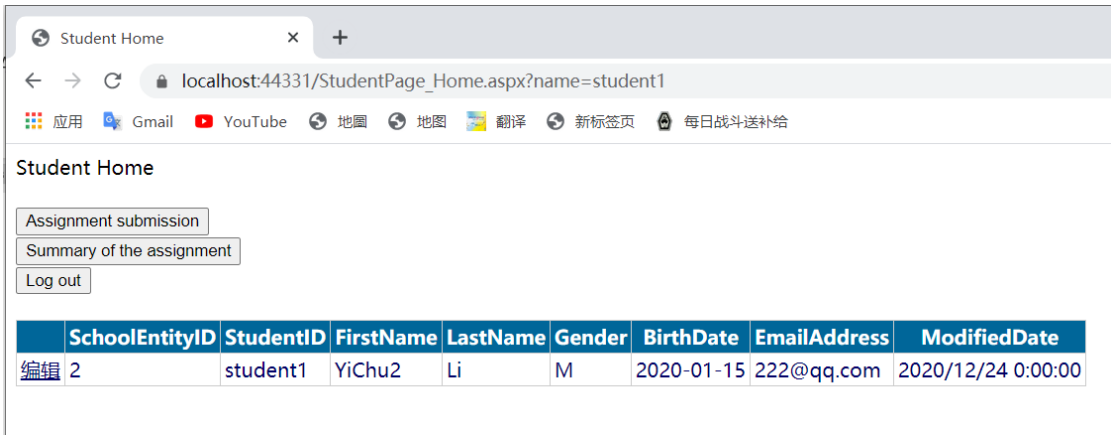
This the first page of student, Login interface. Student can input ‘StudentID’ and ‘Password’ to login, then students need choose ‘Student’ button and click log button. After that students can get into the system.



Picture 14: Online Grading System login interface

Second Page of Student:

After student success login the system, the page will be look like this. Student can change themselves information in the table. If the student clicks the last button [Log out], the user will be logout.



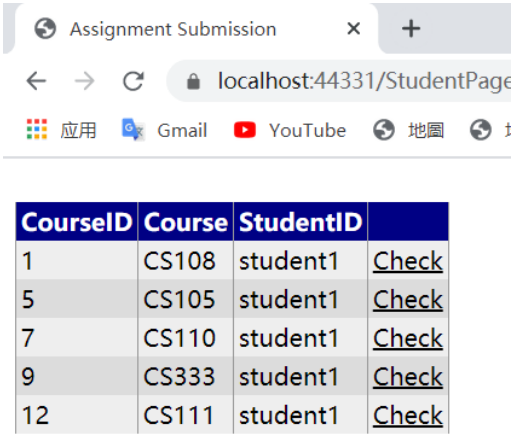
Picture15: Student Home Page

If the student clicks the last button [edit], the student can change his information.

Picture16: Student can change themselves information

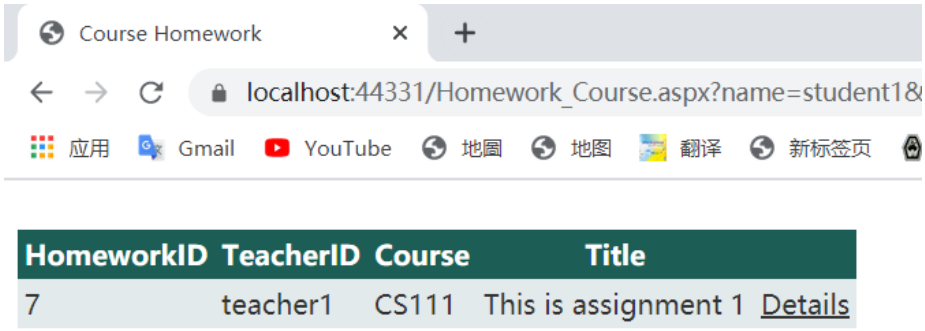
Third Page of Student:

If the student clicks the first button [Assignment submission], the page will be look like this.

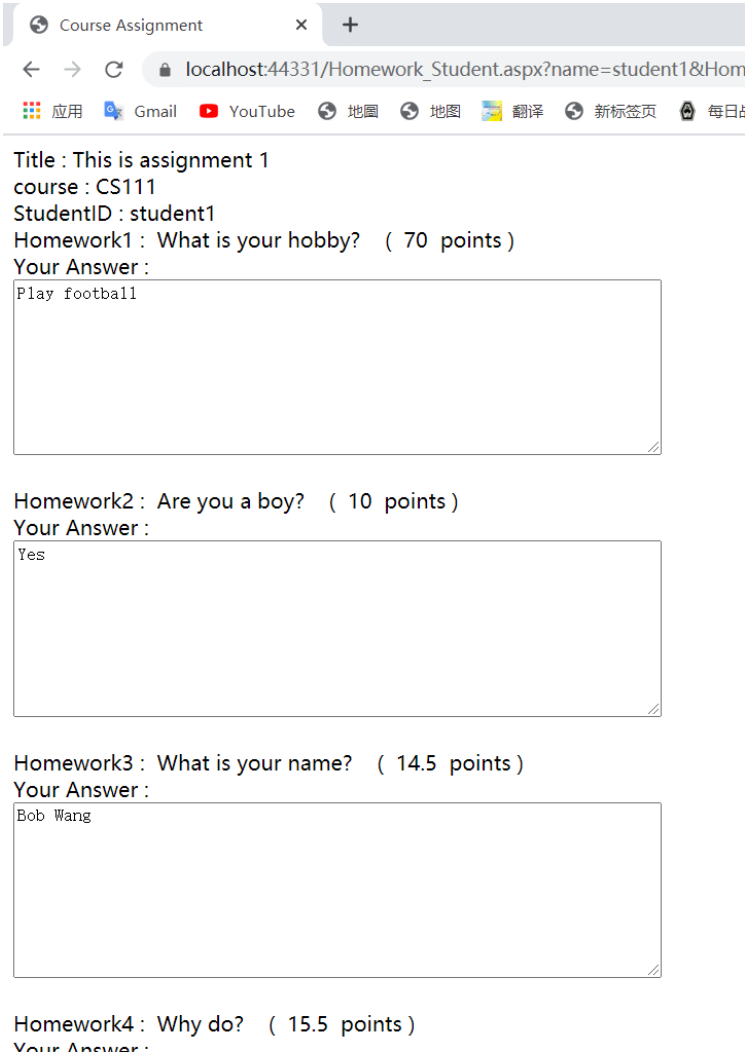


Picture17: Student can click check button to do course assignment

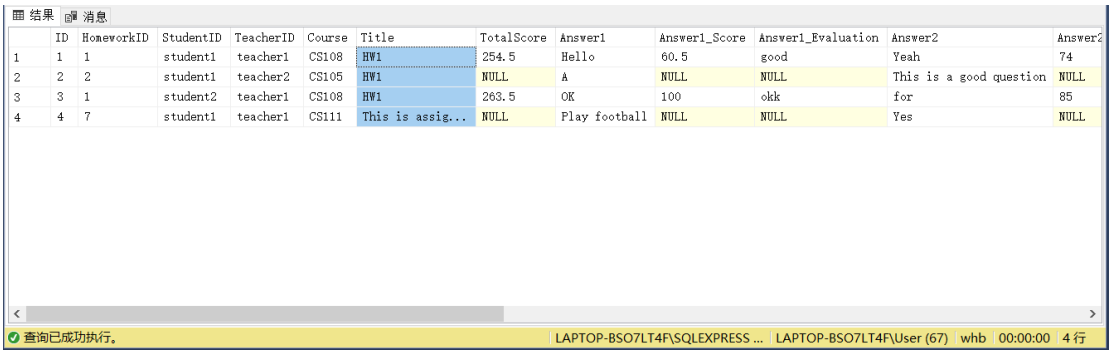
Then student can choose a course, then the student can select which assignment will do. Then the student can click details and he will get into course assignment page to assign submission and the answer will add into SQL Table Homework.Student



Picture18: Student can click Details button to open assignment



Picture19: Student is completing homework



结果 消息

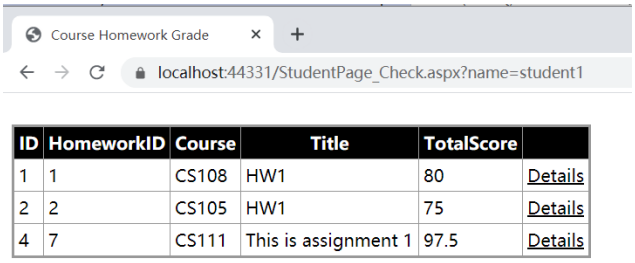
	ID	HomeworkID	StudentID	TeacherID	Course	Title	TotalScore	Answer1	Answer1_Score	Answer1_Evaluation	Answer2	Answer2
1	1	1	student1	teacher1	CS108	HW1	254.5	Hello	60.5	good	Yeah	74
2	2	2	student1	teacher2	CS105	HW1	NULL	A	NULL	NULL	This is a good question	NULL
3	3	1	student2	teacher1	CS108	HW1	263.5	OK	100	okk	for	85
4	4	7	student1	teacher1	CS111	This is assig...	NULL	Play football	NULL	NULL	Yes	NULL

查询已成功执行。 LAPTOP-BSO7LT4F\SQLEXPRESS ... LAPTOP-BSO7LT4F\User (67) whb 00:00:00 4 行

Picture20: The information is succeed adding into SQL Table Homework.Student

Fourth Page of Student:

If the student clicks the second button [Summary of the assignment], the page will be look like this.



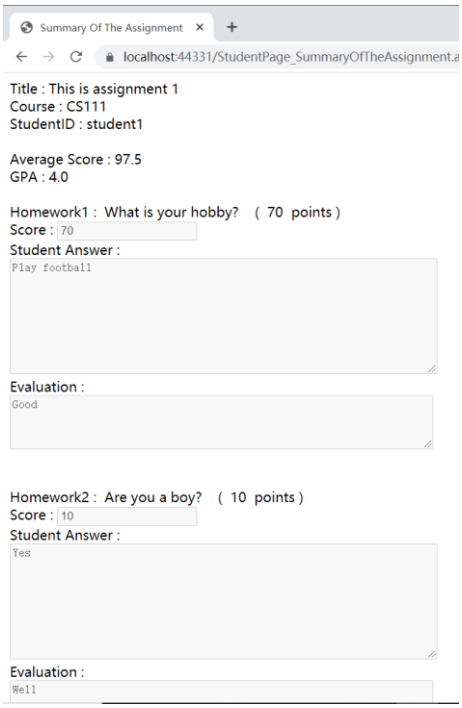
Course Homework Grade

localhost:44331/StudentPage_Check.aspx?name=student1

ID	HomeworkID	Course	Title	TotalScore	
1	1	CS108	HW1	80	Details
2	2	CS105	HW1	75	Details
4	7	CS111	This is assignment 1	97.5	Details

Picture21: The student can look details about them homework

Then student can choose a homework and look details about the score



Summary Of The Assignment

localhost:44331/StudentPage_SummaryOfTheAssignment.a

Title : This is assignment 1
Course : CS111
StudentID : student1

Average Score : 97.5
GPA : 4.0

Homework1 : What is your hobby? (70 points)
Score : 70
Student Answer :
Play football

Evaluation :
Good

Homework2 : Are you a boy? (10 points)
Score : 10
Student Answer :
Yes

Evaluation :
Well

Picture22: The details about homework