Konya Food & Agriculture University Dormitory Management System

Group 7

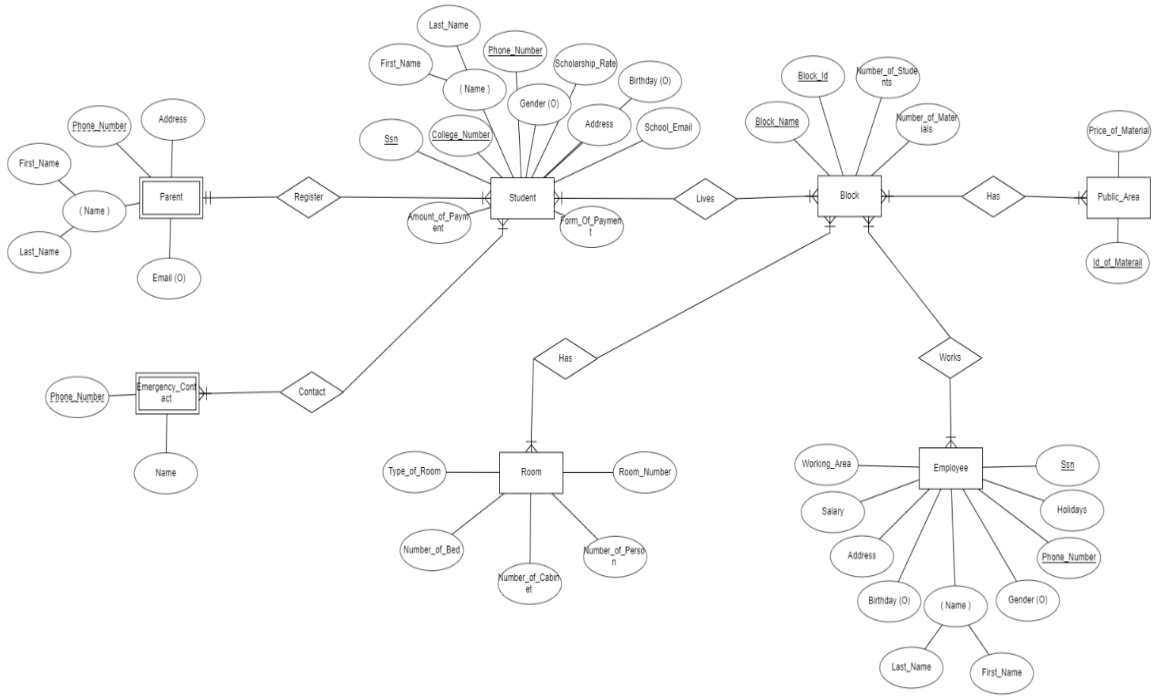
Members of Group:

Fatih ATALAY 192010020008 Ertuğrul KÖKSAL 192010020006

In our project we design a dormitory management system for Konya Food & Agriculture University. We aimed to doing basic but effective system for a dormitory manager. We did our entities suitable in KFAU dormitory.

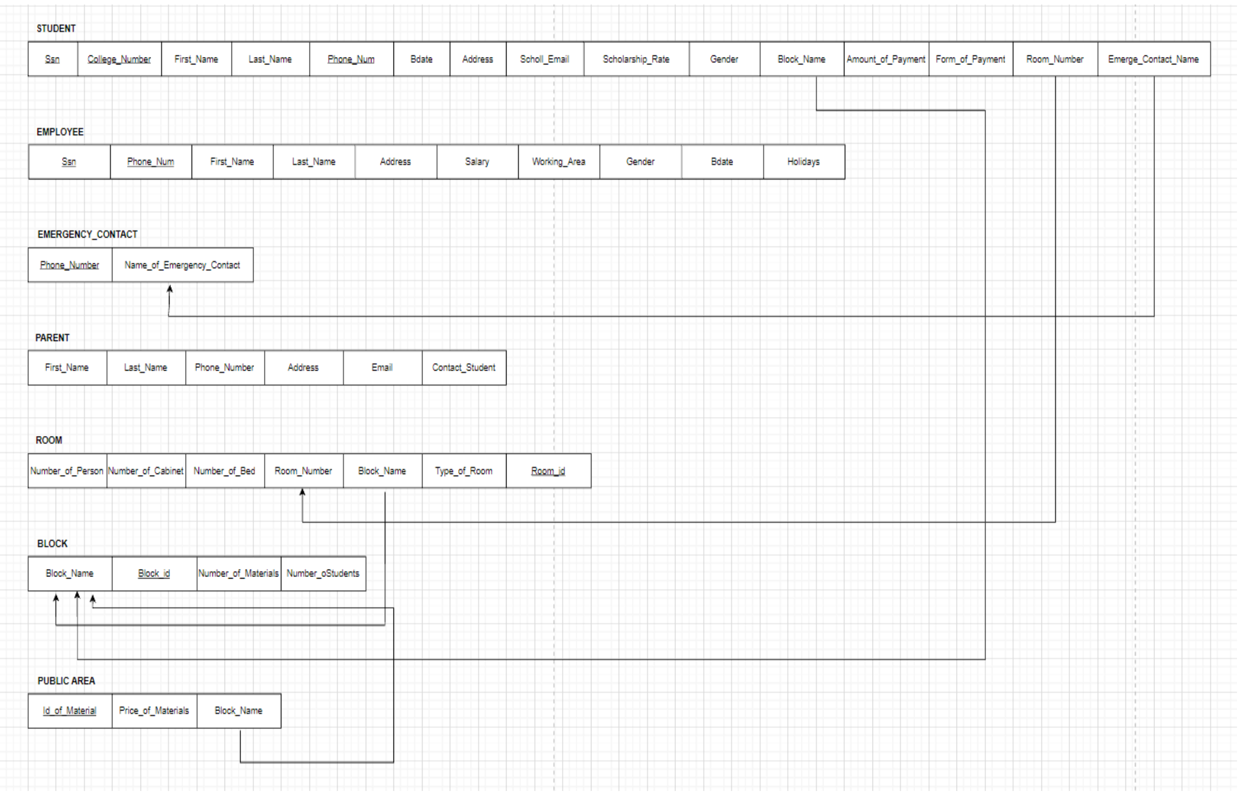
Firstly, we designed an ERD diagram. In this diagram, our assets are as follows; there are students hosted in the dormitory, public areas and rooms in the blocks, and blocks that are the living spaces of the students, the employees which are working in the dormitory blocks, parents and emergency contact who are connected with the students.

ERD Diagram of our Project:



Secondly, we designed a relational schema. In this step we make some relational connections such as “Block_Number”, “Room_Number” and” Emergence_Contact_Name”.

The result of our Relational schema is like that:



STUDENT

Ssn	College_Number	First_Name	Last_Name	Phone_Num	Bdate	Address	Scholl_Email	Scholarship_Rate	Gender	Block_Name	Amount_of_Payment	Form_of_Payment	Room_Number	Emerge_Contact_Name

EMPLOYEE

Ssn	Phone_Num	First_Name	Last_Name	Address	Salary	Working_Area	Gender	Bdate	Holidays

EMERGENCY_CONTACT

Phone_Number	Name_of_Emergency_Contact

PARENT

First_Name	Last_Name	Phone_Number	Address	Email	Contact_Student

ROOM

Number_of_Person	Number_of_Cabinet	Number_of_Bed	Room_Number	Block_Name	Type_of_Room	Room_id

BLOCK

Block_Name	Block_id	Number_of_Materials	Number_oStudents

PUBLIC AREA

Id_of_Material	Price_of_Materials	Block_Name

Then we converted our database diagram into SQL code. Firstly, we added elements into the tables with create function. Also, we added ids into tables because of counting number situation. And there are our codes:

```
CREATE TABLE Block(
blockId int identity(1,1) NOT NULL PRIMARY KEY,
blockName varchar(10) NOT NULL UNIQUE,
numberOfMaterial int NOT NULL,
numberOfStudents int NOT NULL,
)
```

--In Block Table we have our block’s number, number of materials and students inside the block.

```
CREATE TABLE Room(
roomId int identity(1,1) NOT NULL PRIMARY KEY,
blockName varchar(10) NOT NULL UNIQUE,
roomNumber int NOT NULL,
typeofRoom varchar(15) NOT NULL,
numOfPerson int ,
numOfCabinet int NOT NULL,
numOfBed int NOT NULL,
FOREIGN KEY(blockName) REFERENCES Block(blockName),
)
```

--In Room Table we have an identity id, block’s number, number of room, room type, and number of person, cabinet and bed inside the room.

```
CREATE TABLE Student(
studentId int identity(1,1) NOT NULL PRIMARY KEY,
blockName varchar(10),
roomId int,
firstName nvarchar(50) NOT NULL,
lastName nvarchar(50) NOT NULL,
ssn varchar(11) NOT NULL UNIQUE,
collegeNumber varchar(20) NOT NULL UNIQUE,
phoneNumber varchar(20) NOT NULL UNIQUE,
emergencyId int NOT NULL,
parentId int NOT NULL,
birthDate DateTime,
address varchar(200),
schollEmail varchar(70),
scholarshipRate varchar(5),
gender char(1),
amountOfPayment varchar(10),
formOfPayment varchar(10),
FOREIGN KEY(parentId) REFERENCES Parent(parentId),
FOREIGN KEY(emergencyId) REFERENCES EmergencyContact(emergencyId),
FOREIGN KEY(blockId) REFERENCES Block(blockId),
FOREIGN KEY(roomId) REFERENCES Room(roomId),
)
```

--In Student table we kept identity id for counting, student’s block number, student’s room number, first and last name of student, ssn number of students, college number, phone number, name of emergency contact birthday, address, school email, scholarship rate gender, amount and form of payment of our students. We took block number from block table, and room id from room table.

```
CREATE TABLE Employee(
employeeId int identity(1,1) NOT NULL PRIMARY KEY,
ssn varchar(11) NOT NULL UNIQUE,
firstName nvarchar(50) NOT NULL,
lastName nvarchar(50) NOT NULL,
phoneNumber  nvarchar(20)) NOT NULL UNIQUE,
workingArea varchar (20),
salary float NOT NULL,
birthDate Datetime,
address varchar(200),
holidays int,
gender char(1),
)
```

--In Employee Table we kept identity id and staff information just like ssn, first and last name, phone number, working area, salary, birthday, address, holidays and gender of employees.

```
CREATE TABLE EmergencyContact(
emergencyId int identity(1,1) NOT NULL PRIMARY KEY,
nameofEmergencyContact varchar(100) NOT NULL UNIQUE,
phoneNumber varchar(20) NOT NULL UNIQUE,
)
```

--In EmergencyContact Table we kept our student’s emergency contact’s information. Such as name and phone number of contact. Also, we kept identity id.

```
CREATE TABLE Parent(
parentId int identity(1,1) NOT NULL PRIMARY KEY,
firstName varchar(50) NOT NULL,
lastName varchar(50) NOT NULL,
phoneNumber varchar(15) NOT NULL UNIQUE,
address varchar(200),
email varchar(25),
contactStudentSsn varchar(20) NOT NULL,
)
```

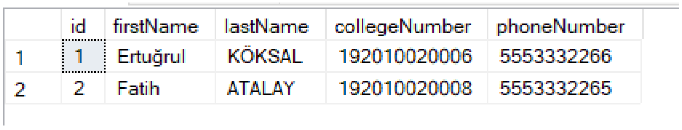
--In Parent Table we kept first and last name of student’s parent, phone number address, email and registered student name kept also in this table.

```
CREATE TABLE PublicArea(
idOfMaterial int identity(1,1) NOT NULL PRIMARY KEY,
priceOfMaterials int,
blockName varchar(10),
FOREIGN KEY(blockName) REFERENCES Block(blockName)
)
```

--In Public Area Table we kept block’s common area information like id and price of materials in the blocks. Also, we took block number from block table for matching.

```
CREATE VIEW [femaleStudents] AS
SELECT studentId, firstName, lastName, collegeNumber, phoneNumber
FROM Student
WHERE gender= 'K';
--In View[femaleStudents] we showed female student's first and last name, college number --
and phone number of the female students also showed in this View.
```

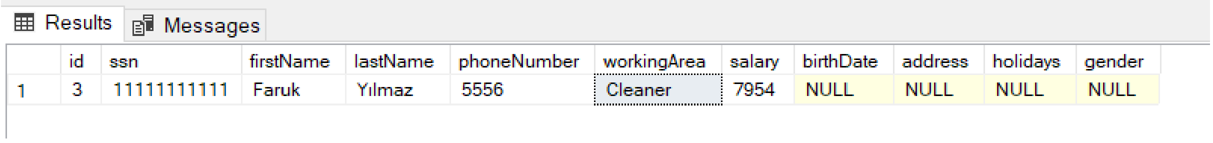
```
CREATE VIEW [maleStudents] AS
SELECT studentId, firstName, lastName, collegeNumber, phoneNumber
FROM Student
WHERE gender= 'E;
--In View[maleStudents] we showed male student's first and last name. College number and --
phone number of the male students also showed in this View.
```



	id	firstName	lastName	collegeNumber	phoneNumber
1	1	Ertuğrul	KÖKSAL	192010020006	5553332266
2	2	Fatih	ATALAY	192010020008	5553332265

Figure of View [maleStudents]

```
CREATE PROCEDURE empProcedure @wArea varchar(50), @lName varchar(50)
AS
SELECT * FROM Employee WHERE workingArea = @wArea AND lastName = @lName
-- We showed the employees in the empProcedure Procedure with their working areas and
surnames.
```

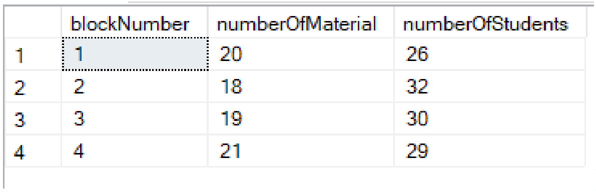


	id	ssn	firstName	lastName	phoneNumber	workingArea	salary	birthDate	address	holidays	gender
1	3	11111111111	Faruk	Yılmaz	5556	Cleaner	7954	NULL	NULL	NULL	NULL

Figure of empProcedure

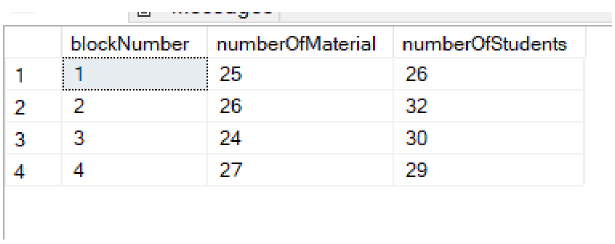
```
CREATE trigger updateNumOfMaterails
on Room
after insert
as
begin
        declare @blockNumber varchar(10)
        declare @numofbed int
        declare @numofcabinet int
        select  @numofbed= numOfBed, @numofcabinet= numOfCabinet from inserted

        update Block
        set numberOfMaterial = numberOfMaterial + @numofbed +@numofcabinet
        where blockName=@blockNumber
end;
```

--In updateNumOfMaterails trigger we made an addition which is about the when we entered new number of bed and number of cabinet in Room Table, it adds the number of them and send it to numberOfMaterial which inside the Block Table.



	blockNumber	numberOfMaterial	numberOfStudents
1	1	20	26
2	2	18	32
3	3	19	30
4	4	21	29

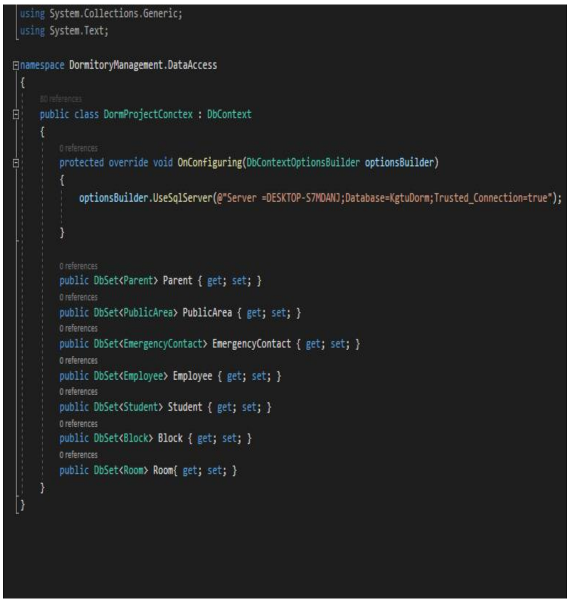
Before adding in Room Table



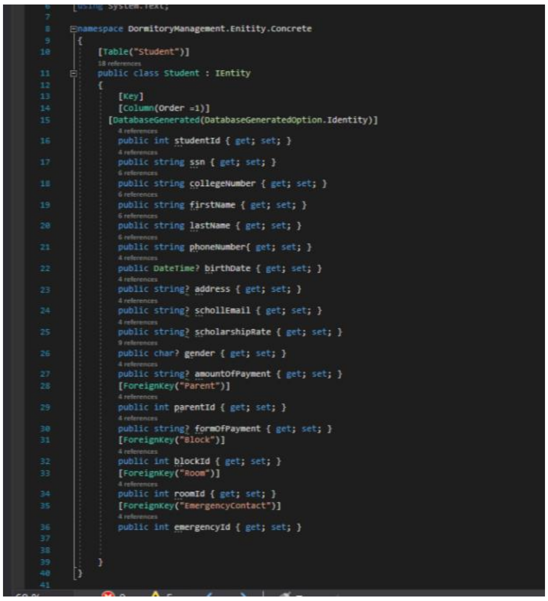
	blockNumber	numberOfMaterial	numberOfStudents
1	1	25	26
2	2	26	32
3	3	24	30
4	4	27	29

After adding in Room Table

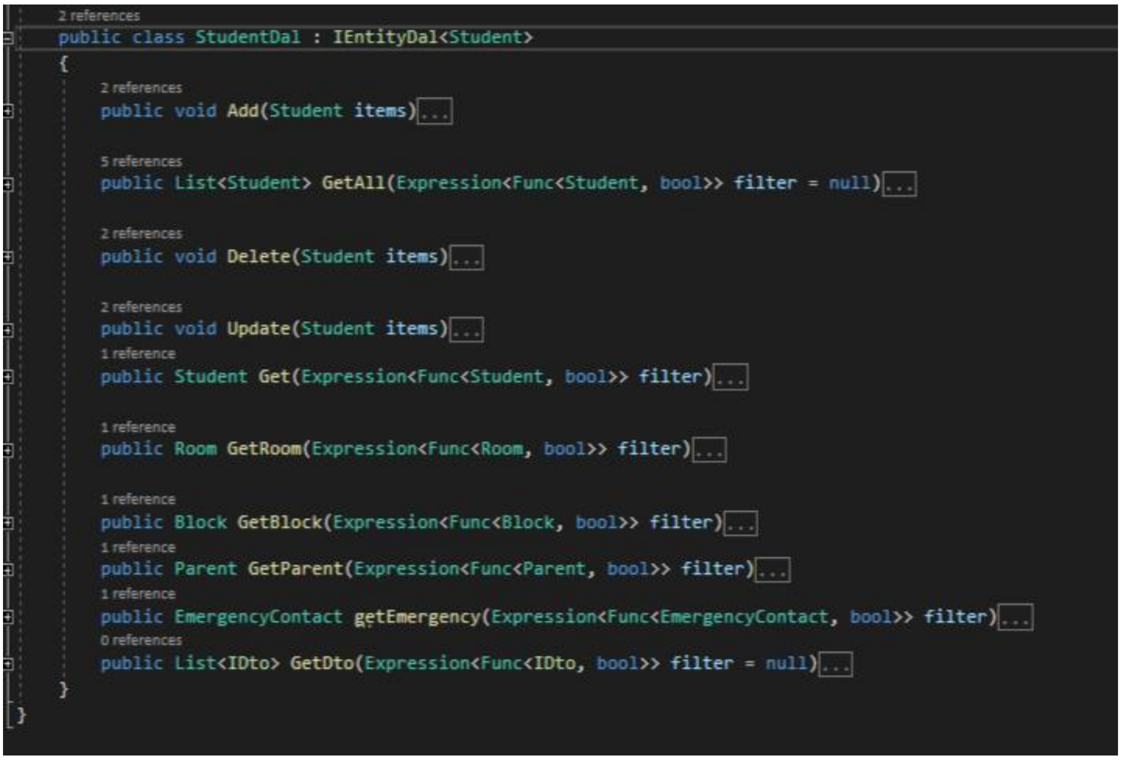
Finally, we implemented our project with C#. There are some snapshots from our application:



```
using System.Collections.Generic;
using System.Text;

namespace DormitoryManagement.DataAccess
{
    80 references
    public class DormProjectConctex : DbContext
    {
        0 references
        protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
        {
            optionsBuilder.UseSqlServer(@"Server =DESKTOP-S7MDANJ;Database=KgtuDorm;Trusted_Connection=true");

        }

        0 references
        public DbSet<Parent> Parent { get; set; }
        0 references
        public DbSet<PublicArea> PublicArea { get; set; }
        0 references
        public DbSet<EmergencyContact> EmergencyContact { get; set; }
        0 references
        public DbSet<Employee> Employee { get; set; }
        0 references
        public DbSet<Student> Student { get; set; }
        0 references
        public DbSet<Block> Block { get; set; }
        0 references
        public DbSet<Room> Room{ get; set; }
    }
}
```

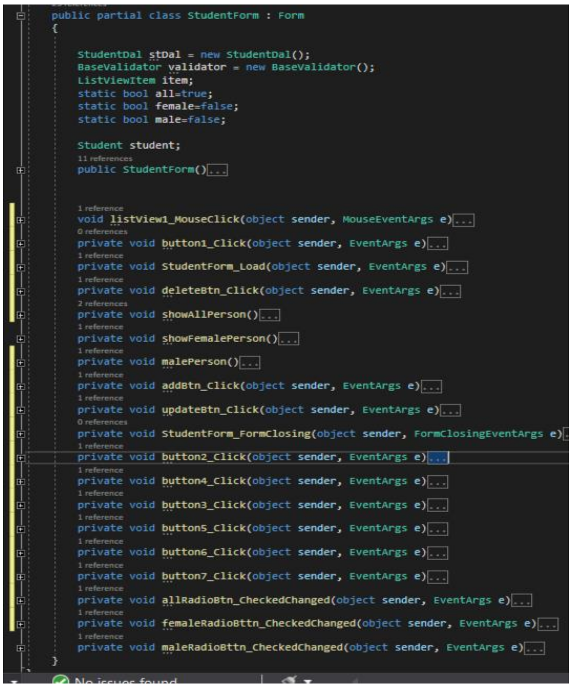


```
using System.Text;

namespace DormitoryManagement.Enitity.Concrete
{
    [Table("Student")]
    18 references
    public class Student : IEntity
    {
        [Key]
        [Column(Order =1)]
      [DatabaseGenerated(DatabaseGeneratedOption.Identity)]
        4 references
        public int studentId { get; set; }
        4 references
        public string ssn { get; set; }
        6 references
        public string collegeNumber { get; set; }
        6 references
        public string firstName { get; set; }
        6 references
        public string lastName { get; set; }
        6 references
        public string phoneNumber{ get; set; }
        4 references
        public DateTime? birthDate { get; set; }
        4 references
        public string? address { get; set; }
        4 references
        public string? schollEmail { get; set; }
        4 references
        public string? scholarshipRate { get; set; }
        9 references
        public char? gender { get; set; }
        4 references
        public string? amountOfPayment { get; set; }
        [ForeignKey("Parent")]
        4 references
        public int parentId { get; set; }
        4 references
        public string? formOfPayment { get; set; }
        [ForeignKey("Block")]
        4 references
        public int blockId { get; set; }
        [ForeignKey("Room")]
        4 references
        public int roomId { get; set; }
        [ForeignKey("EmergencyContact")]
        4 references
        public int emergencyId { get; set; }


    }
}
```

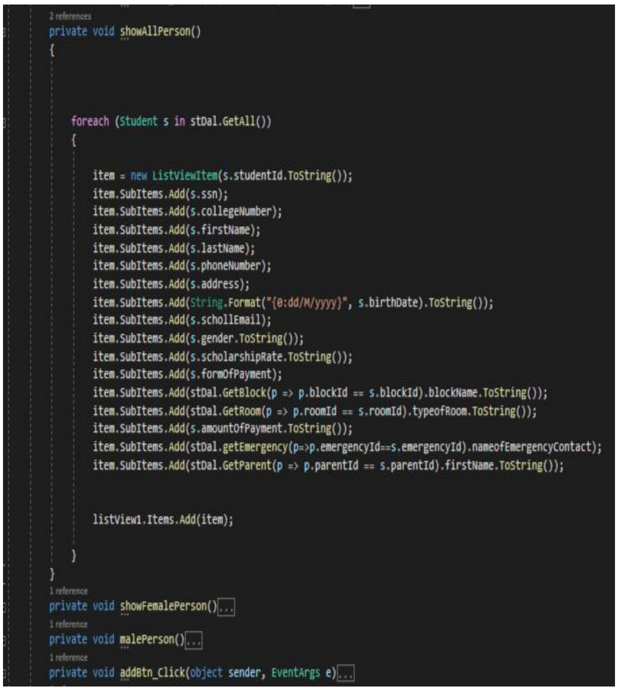


```
    2 references
    public class StudentDal : IEntityDal<Student>
    {
        2 references
        public void Add(Student items)...

        5 references
        public List<Student> GetAll(Expression<Func<Student, bool>> filter = null)...

        2 references
        public void Delete(Student items)...

        2 references
        public void Update(Student items)...
        1 reference
        public Student Get(Expression<Func<Student, bool>> filter)...

        1 reference
        public Room GetRoom(Expression<Func<Room, bool>> filter)...

        1 reference
        public Block GetBlock(Expression<Func<Block, bool>> filter)...
        1 reference
        public Parent GetParent(Expression<Func<Parent, bool>> filter)...
        1 reference
        public EmergencyContact getEmergency(Expression<Func<EmergencyContact, bool>> filter)...
        0 references
        public List<IDto> GetDto(Expression<Func<IDto, bool>> filter = null)...
    }
}
```

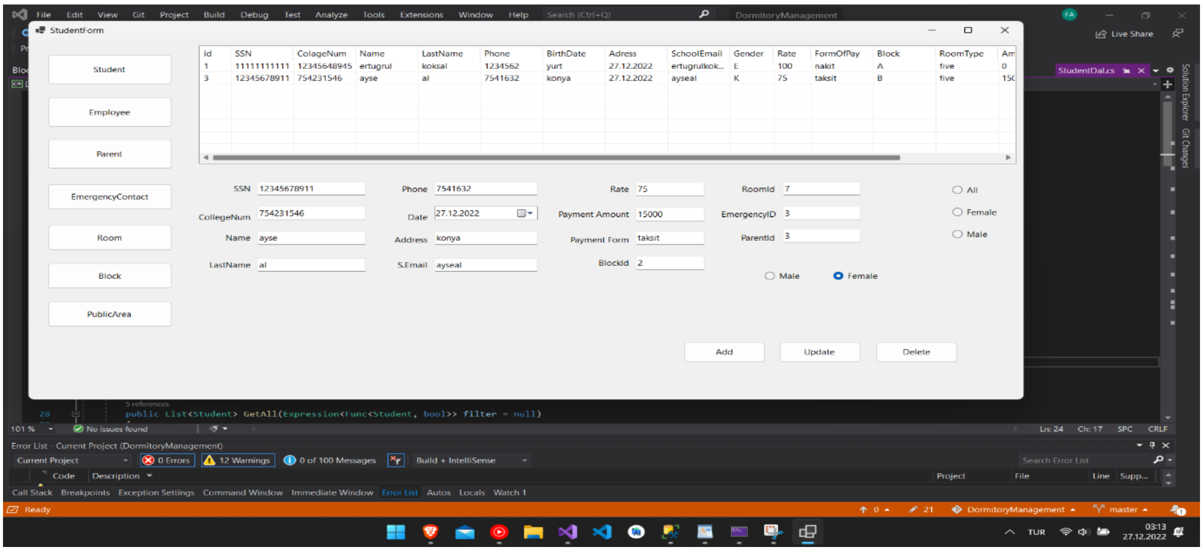
Konya Food & Agriculture University Dormitory Management System



```
public partial class StudentForm : Form
{

    StudentDal stDal = new StudentDal();
    BaseValidator validator = new BaseValidator();
    ListViewItem item;
    static bool all=true;
    static bool female=false;
    static bool male=false;

    Student student;
    11 references
    public StudentForm()[...]


    1 reference
    void listView1_MouseClick(object sender, MouseEventArgs e)[...]
    0 references
    private void button1_Click(object sender, EventArgs e)[...]
    1 reference
    private void StudentForm_Load(object sender, EventArgs e)[...]
    1 reference
    private void deleteBtn_Click(object sender, EventArgs e)[...]
    2 references
    private void showAllPerson()[...]
    1 reference
    private void showFemalePerson()[...]
    1 reference
    private void malePerson()[...]
    1 reference
    private void addBtn_Click(object sender, EventArgs e)[...]
    1 reference
    private void updateBtn_Click(object sender, EventArgs e)[...]
    0 references
    private void StudentForm_FormClosing(object sender, FormClosingEventArgs e)[
    1 reference
    private void button2_Click(object sender, EventArgs e)[...]
    1 reference
    private void button4_Click(object sender, EventArgs e)[...]
    1 reference
    private void button3_Click(object sender, EventArgs e)[...]
    1 reference
    private void button5_Click(object sender, EventArgs e)[...]
    1 reference
    private void button6_Click(object sender, EventArgs e)[...]
    1 reference
    private void button7_Click(object sender, EventArgs e)[...]
    1 reference
    private void allRadioBtn_CheckedChanged(object sender, EventArgs e)[...]
    1 reference
    private void femaleRadioBttn_CheckedChanged(object sender, EventArgs e)[...]
    1 reference
    private void maleRadioBttn_CheckedChanged(object sender, EventArgs e)[...]
}
```



```
2 references
private void showAllPerson()
{



    foreach (Student s in stDal.GetAll())
    {

        item = new ListViewItem(s.studentId.ToString());
        item.SubItems.Add(s.ssn);
        item.SubItems.Add(s.collegeNumber);
        item.SubItems.Add(s.firstName);
        item.SubItems.Add(s.lastName);
        item.SubItems.Add(s.phoneNumber);
        item.SubItems.Add(s.address);
        item.SubItems.Add(String.Format("{0:dd/M/yyyy}", s.birthDate).ToString());
        item.SubItems.Add(s.schollEmail);
        item.SubItems.Add(s.gender.ToString());
        item.SubItems.Add(s.scholarshipRate.ToString());
        item.SubItems.Add(s.formOfPayment);
        item.SubItems.Add(stDal.GetBlock(p => p.blockId == s.blockId).blockName.ToString());
        item.SubItems.Add(stDal.GetRoom(p => p.roomId == s.roomId).typeofRoom.ToString());
        item.SubItems.Add(s.amountOfPayment.ToString());
        item.SubItems.Add(stDal.getEmergency(p=>p.emergencyId==s.emergencyId).nameofEmergencyContact);
        item.SubItems.Add(stDal.GetParent(p => p.parentId == s.parentId).firstName.ToString());


        listView1.Items.Add(item);

    }
}
1 reference
private void showFemalePerson()[...]
1 reference
private void malePerson()[...]
1 reference
private void addBtn_Click(object sender, EventArgs e)[...]
```



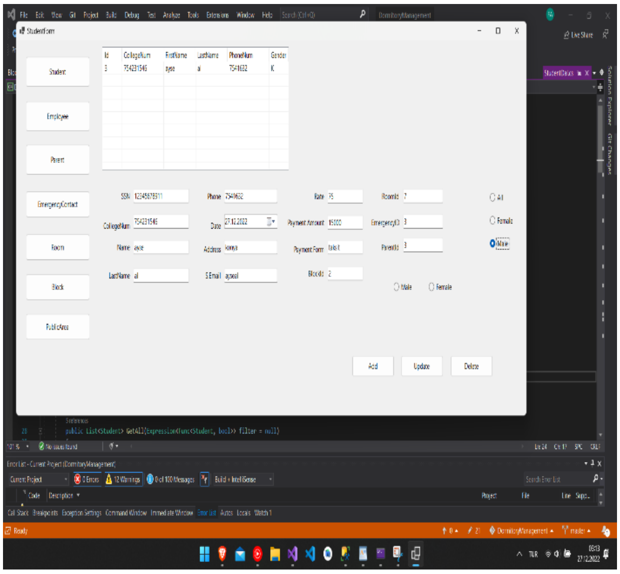
id	SSN	ColageNum	Name	LastName	Phone	BirthDate	Adress	SchoolEmail	Gender	Rate	FormOfPay	Block	RoomType	Am
1	11111111111	12345648945	ertugrul	koksal	1234562	yurt	27.12.2022	ertugrulkok...	E	100	nakit	A	five	0
3	12345678911	754231546	ayse	al	7541632	konya	27.12.2022	ayseal	K	75	taksit	B	five	15(

SSN 12345678911
CollegeNum 754231546
Name ayse
LastName al
Phone 7541632
Date 27.12.2022
Address konya
S.Email ayseal
Rate 75
Payment Amount 15000
Payment Form taksit
BlockId 2
RoomId 7
EmergencyID 3
ParentId 3
○ All
○ Female
○ Male
○ Male ● Female

```
5 references
public List<Student> GetAll(Expression<Func<Student, bool>> filter = null)
```

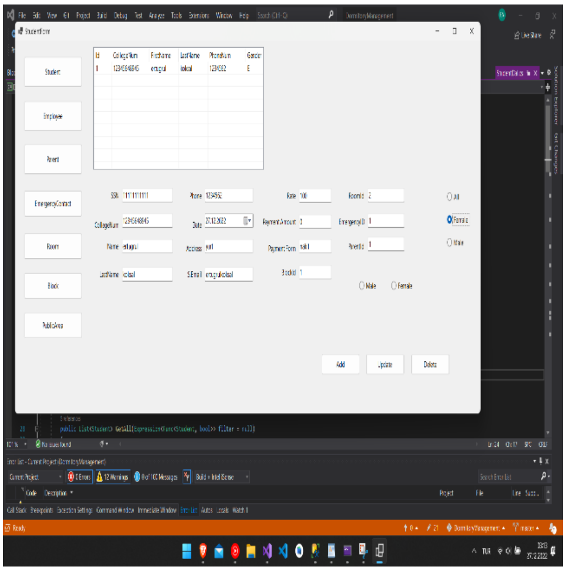
101 % No issues found Ln: 24 Ch: 17 SPC CRLF

Error List - Current Project (DormitoryManagement)
Current Project 0 Errors 12 Warnings 0 of 100 Messages Build + IntelliSense
Code Description Project File Line Supp...
Call Stack Breakpoints Exception Settings Command Window Immediate Window Error List Autos Locals Watch 1
Ready 0 21 DormitoryManagement master
TUR 03:13 27.12.2022



Id	CollegeNum	FirstName	LastName	PhoneNum	Gender
3	754231546	ayse	al	7541632	K

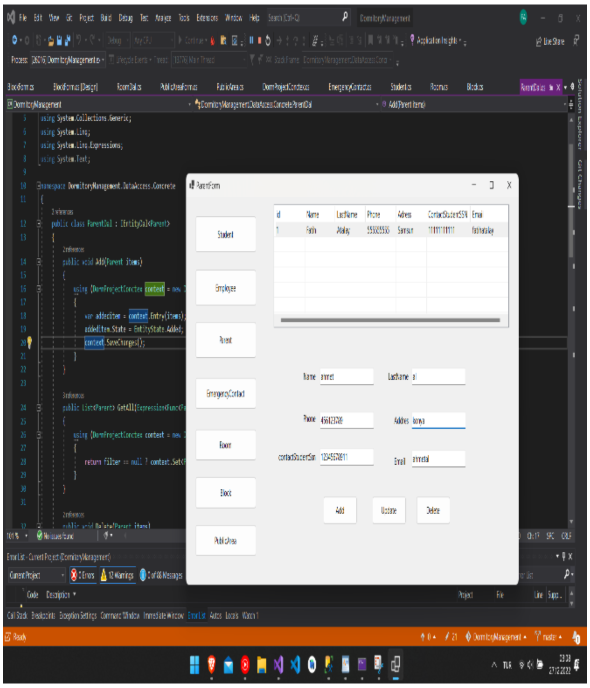
SSN 12345678911
CollegeNum 754231546
Name ayse
LastName al
Phone 7541632
Date 27.12.2022
Address konya
S.Email ayseal
Rate 75
Payment Amount 15000
Payment Form taksit
BlockId 2
RoomId 7
EmergencyID 3
ParentId 3
○ All
○ Female
● Male
○ Male ○ Female



Id	CollegeNum	FirstName	LastName	PhoneNum	Gender
1	12345648945	ertugrul	koksal	1234562	E

SSN 11111111111
CollegeNum 12345648945
Name ertugrul
LastName koksal
Phone 1234562
Date 27.12.2022
Address yurt
S.Email ertugrulkoksal
Rate 100
Payment Amount 0
Payment Form nakit
BlockId 1
RoomId 2
EmergencyID 1
ParentId 1
○ All
● Female
○ Male
○ Male ○ Female

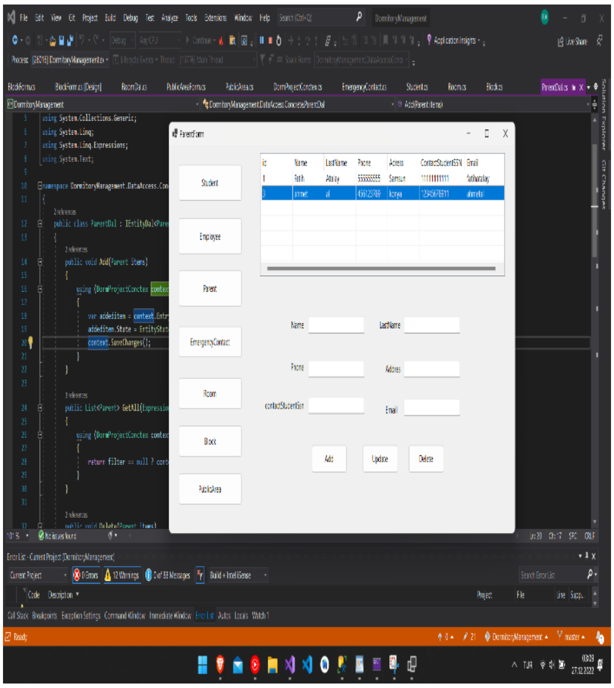
Konya Food & Agriculture University Dormitory Management System



ParentForm

id	Name	LastName	Phone	Adress	ContactStudentSSN	Email
1	Fatih	Atalay	555555555	Samsun	11111111111	fatihatalay

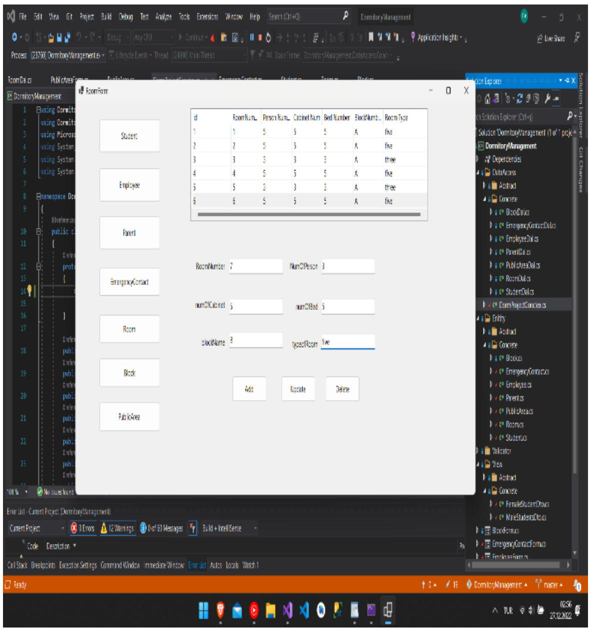
Name: ahmet LastName: al
Phone: 456123789 Addres: konya
contactStudentSsn: 12345678911 Email: ahmetal



ParentForm

id	Name	LastName	Phone	Adress	ContactStudentSSN	Email
1	Fatih	Atalay	555555555	Samsun	11111111111	fatihatalay
3	ahmet	al	456123789	konya	12345678911	ahmetal

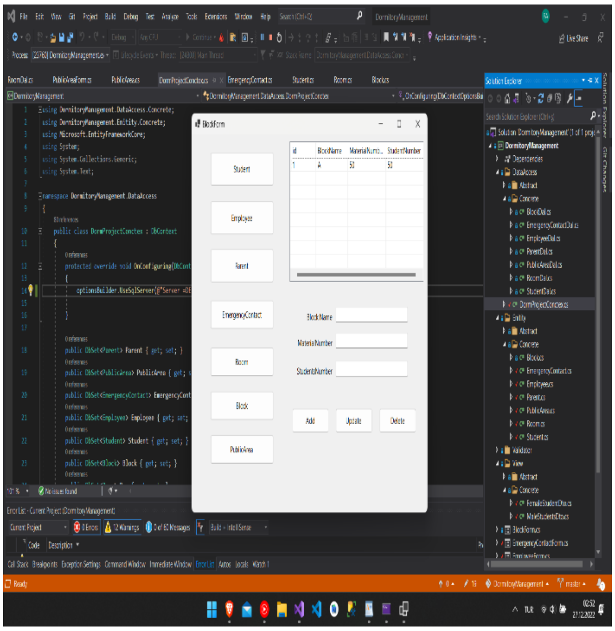
Name LastName
Phone Addres
contactStudentSsn Email



RoomForm

id	RoomNum...	Person Num...	Cabinet Num	Bed Number	BlockNumb...	Room Type
1	1	5	5	5	A	five
2	2	5	5	5	A	five
3	3	3	3	3	A	three
4	4	5	5	5	A	five
5	5	3	3	3	A	three
6	6	5	5	5	A	five

RoomNumber: 7 NumOfPerson: 3
numOfCabinet: 5 numOfBed: 5
blockName: 3 typeofRoom: five

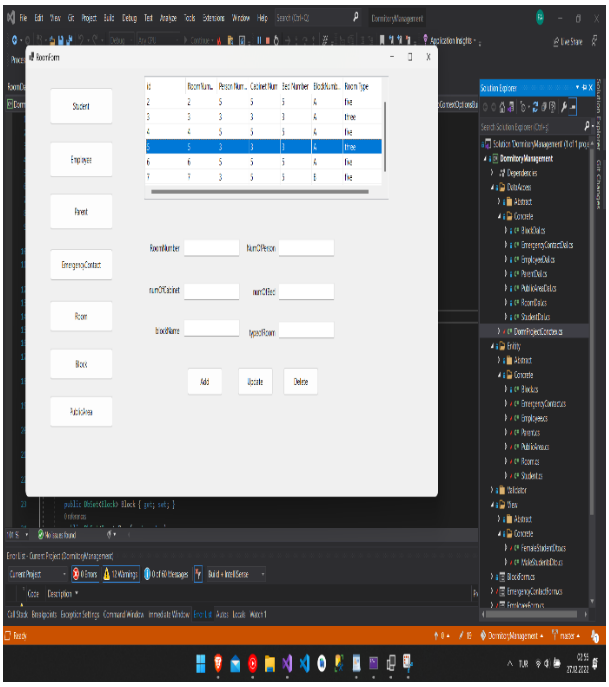


BlockForm

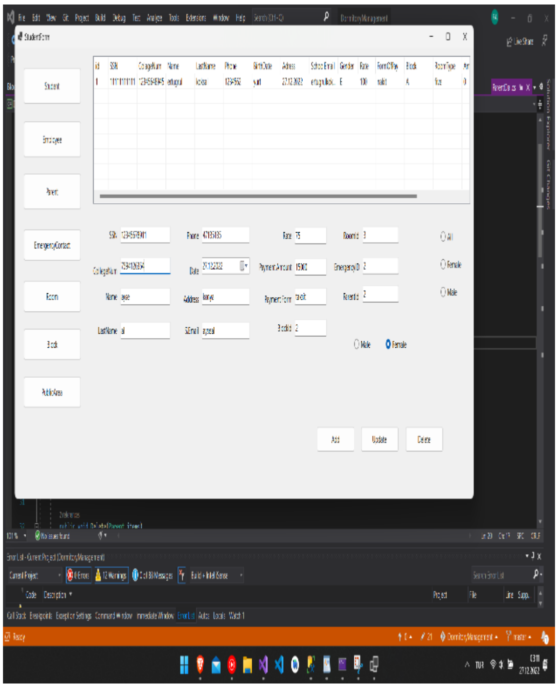
id	BlockName	MaterialNumb...	StudentNumber
1	A	50	50

Block Name
Materia Number
StudentsNumber

Konya Food & Agriculture University Dormitory Management System



id	RoomNum...	Person Num...	Cabinet Num	Bed Number	BlockNumb...	Room Type
2	2	5	5	5	A	five
3	3	3	3	3	A	three
4	4	5	5	5	A	five
5	5	3	3	3	A	three
6	6	5	5	5	A	five
7	7	3	5	5	B	five



id	SSN	ColageNum	Name	LastName	Phone	BirthDate	Adress	School Email	Gender	Rate	FormOfPay	Block	RoomType	Am
1	1111111111	12345648945	ertugrul	kolsal	1234562	yurt	27.12.2022	ertugrulkok...	E	100	nakit	A	five	0

SSN 12345678901
Phone 47135135
Rate 75
RoomId 3
CollegeNum 7594126354
Date 27.12.2022
PaymentAmount 15000
EmergencyID 2
Name ayse
Address konya
Payment Form kredit
ParentId 2
LastName al
SEmail ayseal
BlockId 2
○ Male ● Female