***Sudent.cs***

using System;

namespace Section\_10

{

public class Subject

{

public string Sub\_Name { set; get; }

public int Sub\_Code { set; get; }

public string Sub\_Teacher { set; get; }

public Subject(string sub\_Name, int sub\_Code, string sub\_Teacher)

{

Sub\_Name = sub\_Name;

Sub\_Code = sub\_Code;

Sub\_Teacher = sub\_Teacher;

}

}

}

***Teachers.cs***

using System;

namespace Section\_10

{

public class Teachers

{

public string TName { get; set; }

public int TClass { get; set; }

public string TSection { get; set; }

public Teachers(string tname,int tclass,string tsection )

{

TName = tname;

TClass = tclass;

TSection = tsection;

}

}

}

***Subject.cs***

using System;

namespace Section\_10

{

public class Subject

{

public string Sub\_Name { set; get; }

public int Sub\_Code { set; get; }

public string Sub\_Teacher { set; get; }

public Subject(string sub\_Name, int sub\_Code, string sub\_Teacher)

{

Sub\_Name = sub\_Name;

Sub\_Code = sub\_Code;

Sub\_Teacher = sub\_Teacher;

}

}

}

***Operations.cs***

using System;

using System.Collections.Generic;

namespace Section\_10

{

public class Operations

{

static List<Student> students = new List<Student>();

static List<Teachers> teach= new List<Teachers>();

static List<Subject> subjects = new List<Subject>();

public void AddStudent(Student student)

{

students.Add(student);

Console.WriteLine("Student Added sucessfully");

}

public void AddSubject(Subject subject)

{

subjects.Add(subject);

Console.WriteLine("Subject Added Successfully");

}

public void AddTeacher(Teachers teachers)

{

teach.Add(teachers);

}

public void ViewStudents()

{

foreach (Student student in students)

{

Console.WriteLine(student);

}

}

public void ViewTeachers()

{

foreach (Teachers t in teach)

{

Console.WriteLine($"Name : {t.TName} Class : {t.TClass} section : {t.TSection} ");

}

}

public void ViewSubjects()

{

foreach (Subject sub in subjects)

{

Console.WriteLine($"SubjectName : {sub.Sub\_Name} SubjectCode : {sub.Sub\_Code} SubjectTeacher : {sub.Sub\_Teacher}");

}

}

public List<Student> GetStudentsinClass(int target\_Class) {

return students.FindAll(s=>s.Class == target\_Class);

}

public List<Subject> GetSubjectByTeacher(string Teacher\_name)

{

return subjects.FindAll(sub=>sub.Sub\_Teacher == Teacher\_name);

}

}

}

***Program.cs***

using System;

using System.Collections.Generic;

namespace Section\_10

{

public class Program

{

static void Main(string[] args)

{

Operations op=new Operations();

op.AddStudent(new Student ( "Evan", 8, "A" ));

op.AddStudent(new Student("Alice",9,"B"));

op.AddStudent(new Student("Raj", 9, "B"));

op.AddTeacher(new Teachers("Mr.Jhon",9,"B"));

op.AddTeacher(new Teachers("Mr.Doe", 8, "A"));

op.AddTeacher(new Teachers("Ms.Copper", 9, "B"));

op.AddSubject(new Subject("Math", 101, "Mr.Jhon"));

op.AddSubject(new Subject("Social", 102, "Mr.Doe"));

op.AddSubject(new Subject("science", 103, "Ms.Copper"));

char ch;

do

{

Console.WriteLine("Select\n1.Add Details\t2.Display Details\t3.Students in a class\t4.Students taught by a teacher");

int choice=int.Parse(Console.ReadLine());

switch (choice)

{

case 1:

Console.WriteLine("Select\na.Add details of students\tb.Add details of Teachers\tc.Add Details of subjects ");

char details=char.Parse(Console.ReadLine());

switch (details)

{

case 'a':char addstu;

do

{

Console.WriteLine("Enter Student Name");

string name=Console.ReadLine();

Console.WriteLine("Enter Student Class");

int cl=int.Parse(Console.ReadLine());

Console.WriteLine("Enter Student Section");

string sec=Console.ReadLine();

Student st=new Student(name, cl, sec);

op.AddStudent(st);

Console.WriteLine("if you want to add more student details press y");

addstu = char.Parse(Console.ReadLine().ToLower());

} while (addstu == 'y');

break;

case 'b':

char addteach;

do

{

Console.WriteLine("Enter Teacher Name");

string tname = Console.ReadLine();

Console.WriteLine("Enter teacher Class");

int tcl = int.Parse(Console.ReadLine());

Console.WriteLine("Enter teacher Section");

string tsec = Console.ReadLine();

Teachers teach = new Teachers(tname, tcl, tsec);

op.AddTeacher(teach);

Console.WriteLine("if you want to add more Teacher details press y");

addteach = char.Parse(Console.ReadLine().ToLower());

} while (addteach == 'y');

break;

case 'c':

char addsub;

do

{

Console.WriteLine("Enter Subject Name");

string Subname = Console.ReadLine();

Console.WriteLine("Enter Subject Code");

int Subcode = int.Parse(Console.ReadLine());

Console.WriteLine("Enter Subject Teacher");

string SubTeach = Console.ReadLine();

Subject sub = new Subject(Subname, Subcode, SubTeach);

op.AddSubject(sub);

Console.WriteLine("if you want to add more Subject details press y");

addsub = char.Parse(Console.ReadLine().ToLower());

} while (addsub == 'y');

break;

default:

Console.WriteLine("Invalid Section to add details");

break;

}

break;

case 2:

Console.WriteLine("-----Student Details-----");

op.ViewStudents();

Console.WriteLine("-----Teacher Details-----");

op.ViewTeachers();

Console.WriteLine("-----Subject Details-----");

op.ViewSubjects();

break;

case 3:

Console.WriteLine("Enter class to get list of students");

int cls=int.Parse(Console.ReadLine());

Console.WriteLine($"Students in class {cls} : ");

List<Student> stuinclas= op.GetStudentsinClass(cls);

Console.WriteLine("Stu\_name\tstu\_section");

foreach(var stud in stuinclas)

{

Console.WriteLine($"{stud.Name}\t{stud.Section}");

}

break;

case 4:

Console.WriteLine("Enter teacher name to get list of subject ");

string Tname=Console.ReadLine();

Console.WriteLine($"Subjects taught by {Tname} : ");

List<Subject> subbyteach = op.GetSubjectByTeacher(Tname);

Console.WriteLine("Sub\_Name\tSub\_Code");

foreach (var subj in subbyteach)

{

Console.WriteLine($"{subj.Sub\_Name}\t{subj.Sub\_Code}");

}

break;

default:

Console.WriteLine("Invalid selection");

break;

}

Console.WriteLine("If you want to continue press y");

ch = char.Parse(Console.ReadLine().ToLower());

} while (ch == 'y');

}

}

}