SAMPLE DATABASE IMPLEMENTATION REPORT

Contents

PROJECT SUMMARY

The Office of Disability Services facilitates access to programs and activities, coordinates auxiliary aids and services, provides access to adaptive technology, and when necessary, advocates on behalf of students with members of the campus community.

The Mission of the Office of Disability Services is to assist Syracuse University students with disabilities by offering services to promote and ensure that no student with a disability is denied the benefits of, excluded from participation in, or otherwise subjected to discrimination under educational programs and activities.

The students go in to the Disability Services to set up an appointment. Other way they set up an appointment is through email. At the meeting, both the student and the counselor discuss about the books that he/she requires, schedule exam timings and any other accommodations necessary. Accommodations include a reader (a person who reads the question during the exam), and extended duration for taking an exam. If there are any class notes request, that is also taken down. All of the students' records are maintained in a spreadsheet and a paper file, which are used for later reference.

All of the students' records are maintained in a spreadsheet and a paper file. Every time a student has to schedule an appointment with the counselor, he/she has to come in person or call up the front desk. This is the same case for requesting books, exam schedule and class notes. When the student comes in, his/her records have to be searched through the huge pile of files. Also, the records entered in the spreadsheet are often inconsistent and incomplete.

With the current system, it is difficult to make any updates on the students' records. In addition, every time the Office wants to contact the student through email, the email id of that student is searched through the directory of Syracuse University. The same process happens two weeks before the final exams. This is a tedious process and there is a huge possibility of entering wrong information.

The proposed project is for the Office of Disability Services whose primary aim is to provide services for the disabled students here in Syracuse University. There are 3(three) departments under the Office of Disability Services which are:-

- Examination
- Alternate Format
- Note taking

Each of these services caters to different students. Examination deals with coordinating the students and the respective professors with exam questions and schedule. Alternate Format deals with digitizing course books into digital and machine-readable format. Note taking handles all of the students who ask for class notes for their courses.

The proposed system will maintain all of the data in an Access database. It would be an online portal that would allow disabled students to set up appointments with the counselors, request books and notes, and schedule examination.

The proposed project will allow the students to easily sign up at the Office of Disability Services, requests for an appointment, request for books, schedule an examination and request for class notes.

This system will maintain up-to-date information of each student, which will allow effective communication between the students and the counselors.

The Primary users will be the student and the counselors at Office of Disability of Services. The counselors will have full access to the database; of the students and their accommodations. The students will have somewhat restricted access; only view the counselors' schedule and not the other students' records.

ENTITY AND ATTRIBUTE TABLE

1. Student – This entity captures information about the student who signed up for the services and accommodation requested

ENTITY NAME: STUDENT	ATTRIBUTE NAME	FIELD TYPE	NULL/NOT NULL	EXPLANATION
PRIMARY KEY	StudentID	INT	NOT NULL	Unique Identifier for Student Entity ; SUID given by the University
OTHER ATTRIBUTES	StudentFirstName	VARCHAR(20)	NOT NULL	First Name of the student
	StudentLastName	VARCHAR(20)	NOT NULL	Last Name of the student
	StudentContact	NUM	NOT NULL	Contact Number of the student
	StudentEmail	VARCHAR(20)	NOT NULL	Email of the student
	StudentAddress	VARCHAR(20)	NOT NULL	Address of the student
	Accomodation	VARCHAR(20)	NOT NULL	Accomodation required by the student. For example extended duration of exam time, reader during the exam etc

2. Counsellor – This entity captures the information of the counsellors working in Office of Disability Services.

ENTITY NAME: Counsellor	ATTRIBUTE NAME	FIELD TYPE	NULL/NOT NULL	EXPLANATION
PRIMARY KEY	CounsellorID	INT	NOT NULL	Unique Identifier for Counsellor Entity ; SUID given by the University
OTHER ATTRIBUTES	CounsellorFirstName	VARCHAR(20)	NOT NULL	First Name of the Counsellor
	CounsellorLastName	VARCHAR(20)	NOT NULL	Last Name of the Counsellor

	CounsellorContact	NUM	NOT NULL	Contact Number of the Counsellor
	CounsellorEmail	VARCHAR(20)	NOT NULL	Email of the Counsellor
	CounsellorAddress	VARCHAR(20)	NOT NULL	Address of the Counsellor

3. Course – This entity capture the information of the courses in which the students are enrolled.

ENTITY NAME: Course	ATTRIBUTE NAME	FIELD TYPE	NULL/NOT NULL	EXPLANATION
PRIMARY KEY	CourseID	INT	NOT NULL	Unique Identifier for Counsellor Entity ; SUID given by the University
OTHER ATTRIBUTES	CourseName	VARCHAR(20)	NOT NULL	Name of the Course
	CourseSection	VARCHAR(20)	NOT NULL	Section Number of the Course
	CourseLocation	NUM	NOT NULL	Room/Lab No. of the course
FOREIGN KEY	ProfessorID	INT	NOT NULL	Primary key associated with Professor Entity.

4. Professor – This entity captures the information of the professor who teach the course.

ENTITY NAME: Professor	ATTRIBUTE NAME	FIELD TYPE	NULL/NOT NULL	EXPLANATION
PRIMARY KEY	ProfessorID	INT	NOT NULL	Unique Identifier for Professor Entity ; SUID given by the University
OTHER ATTRIBUTES	ProfessorFirstName	VARCHAR(20)	NOT NULL	First Name of the Professor
	ProfessorLastName	VARCHAR(20)	NOT NULL	Last Name of the Professor
	ProfessorContact	NUM	NOT NULL	Contact Number of the Professor
	ProfessorEmail	VARCHAR(20)	NOT NULL	Email of the Professor
	ProfessorAddress	VARCHAR(20)	NOT NULL	Address of the Professor

5. Notetaker – This entity captures the information of the note taker who takes class notes for a particular course as per the student's request. The student can request 0 or more than one note taker.

ENTITY NAME: Notetaker	ATTRIBUTE NAME	FIELD TYPE	NULL/NOT NULL	EXPLANATION
PRIMARY KEY	NotetakerID	INT	NOT NULL	Unique Identifier for Notetaker Entity ;
OTHER ATTRIBUTES	NotetakerFirstName	VARCHAR(20)	NOT NULL	First Name of the Notetaker
	NotetakerLastName	VARCHAR(20)	NOT NULL	Last Name of the Notetaker
	NotetakerContact	NUM	NOT NULL	Contact Number of the Notetaker
	NotetakerEmail	VARCHAR(20)	NOT NULL	Email of the Notetaker
FOREIGN KEY	StudentID	INT	NOT NULL	Primary Key of Student Entity

6. Appointment – This entity captures the appointment details between the student and the counsellor. The student requests as many as appointment with the counsellor. The counsellor setup the appointment according to the student's request.

ENTITY NAME: Appointment	ATTRIBUTE NAME	FIELD TYPE	NULL/NOT NULL	EXPLANATION
PRIMARY KEY, FOREIGN KEY	StudentID	INT	NOT NULL	Unique Identifier for Notetaker Entity; SUID given by the University.
PRIMARY KEY, FOREIGN KEY	CounsellorID	INT	NOT NULL	Unique Identifier for Appointment Entity ; SUID given by the University
OTHER ATTRIBUTES	AppointmentDate	DATE	NOT NULL	Date of the Appointment
OTHER ATTRIBUTES	AppointmentTime	VARCHAR(20)	NOT NULL	Time of the Appointment

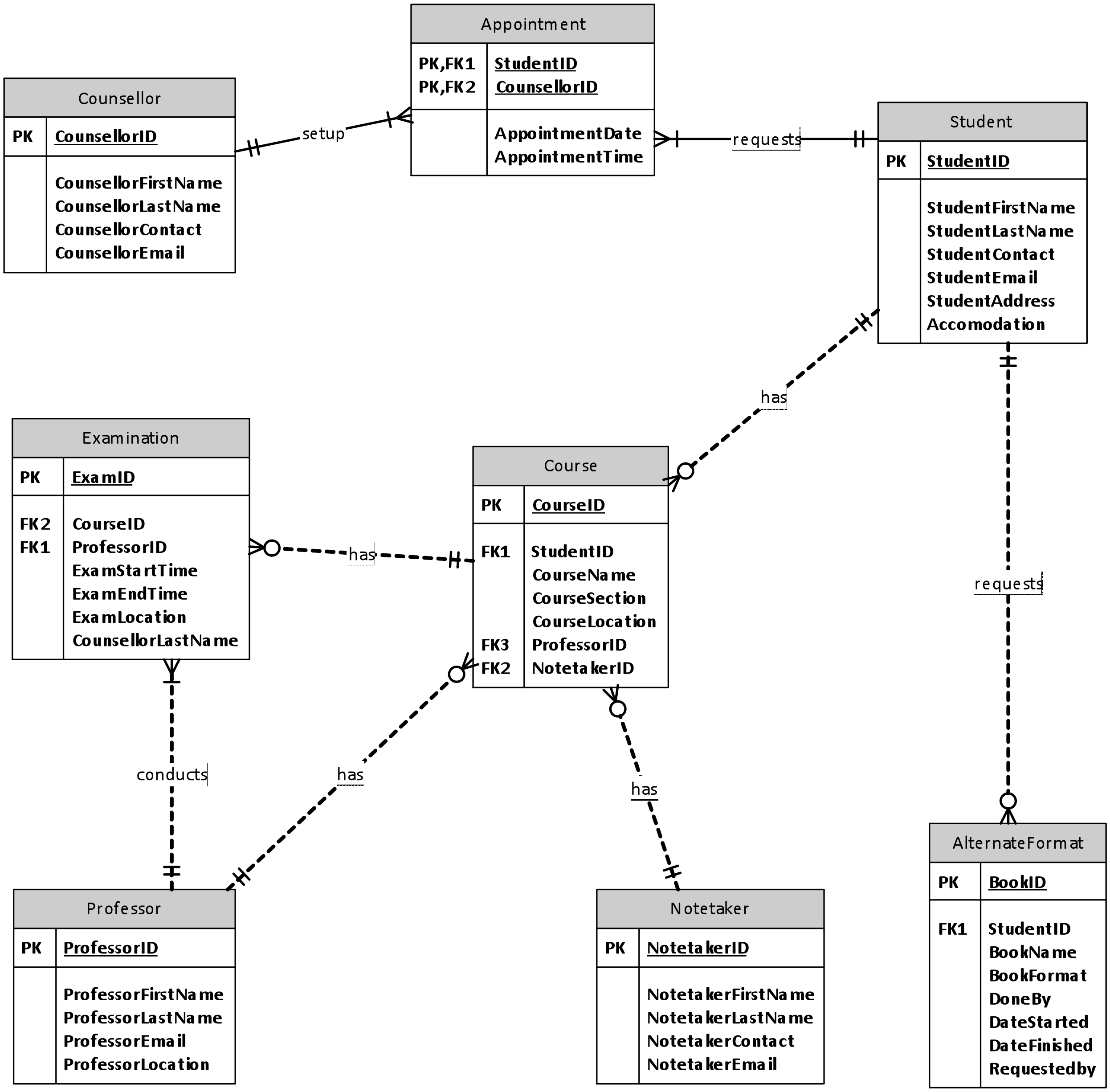
7. Examination – This entity captures the information of the Exam details coordinated between the professor and the Office of Disability Services. At least, one counsellor proctor the exam taken by the students.

ENTITY NAME: Examination	ATTRIBUTE NAME	FIELD TYPE	NULL/NOT NULL	EXPLANATION
PRIMARY KEY	ExamID	INT	NOT NULL	UNIQUE IDENTIFIER for Examination Entity
FOREIGN KEY	StudentID	INT	NOT NULL	Unique Identifier for Student Entity ; SUID given by the University
FOREIGN KEY	CourseID	INT	NOT NULL	Unique Identifier for Course Entity ;
OTHER ATTRIBUTES	ExamDate	DATE	NOT NULL	Name of the Course
OTHER ATTRIBUTES	ExamStartTime	VARCHAR(20)	NOT NULL	Section Number of the Course
OTHER ATTRIBUTES	ExamEndTime	VARCHAR(20)	NOT NULL	Room/Lab No. of the course
FOREIGN KEY	ProfessorID	INT	NOT NULL	Primary key associated with Professor Entity.
FOREIGN KEY	CounsellorID	INT	NOT NULL	Primary key associated with Counsellor Entity.

8. Alternate Format – This entity captures the information of the various books that are requested by the students.

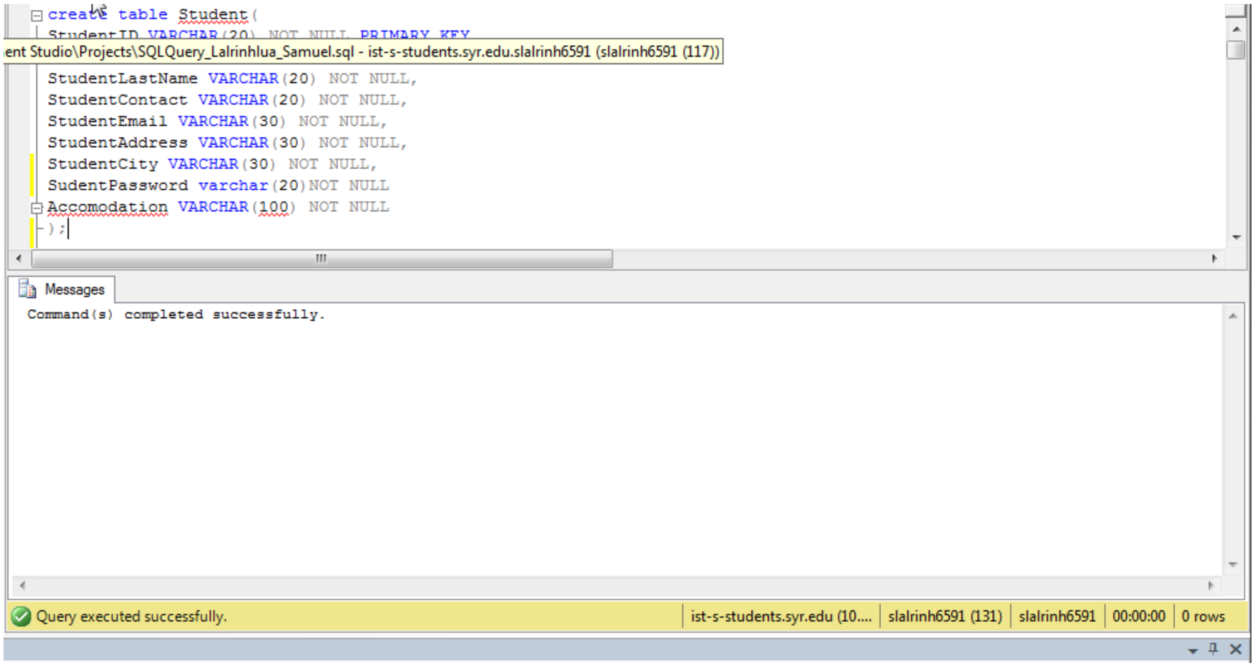
ENTITY NAME: AlternateFormat	ATTRIBUTE NAME	FIELD TYPE	NULL/NOT NULL	EXPLANATION
PRIMARY KEY	BookID	INT	NOT NULL	Unique Identifier for Alternate Format Entity ; ISBN no of the book
OTHER ATTRIBUTES	BookName	VARCHAR(20)	NOT NULL	Name of the book
	BookFormat	VARCHAR(20)	NOT NULL	Format in which the book is required. Example Searchable PDF, Word format etc/
	Doneby	VARCHAR(20)	NOT NULL	Name of the person who processed the book
	DateStarted	DATE	NOT NULL	Date on which the processing of the book is started
	DateFinished	DATE	NOT NULL	Date on which the processing of the book is finished
FOREIGN KEY	StudentID	INT	NOT NULL	Primary Key of Student Entity

ENTITY-RELATIONSHIP DIAGRAM



CREATION OF TABLES

```
create table Student(
StudentID VARCHAR(20) NOT NULL PRIMARY KEY,
StudentFirstName VARCHAR(20) NOT NULL,
StudentLastName VARCHAR(20) NOT NULL,
StudentContact VARCHAR(20) NOT NULL,
StudentEmail VARCHAR(30) NOT NULL,
StudentAddress VARCHAR(30) NOT NULL,
StudentCity VARCHAR(30) NOT NULL,
SudentPassword varchar(20)NOT NULL
Accomodation VARCHAR(100) NOT NULL
);
```



```
create table Student(
StudentID VARCHAR(20) NOT NULL PRIMARY KEY
```

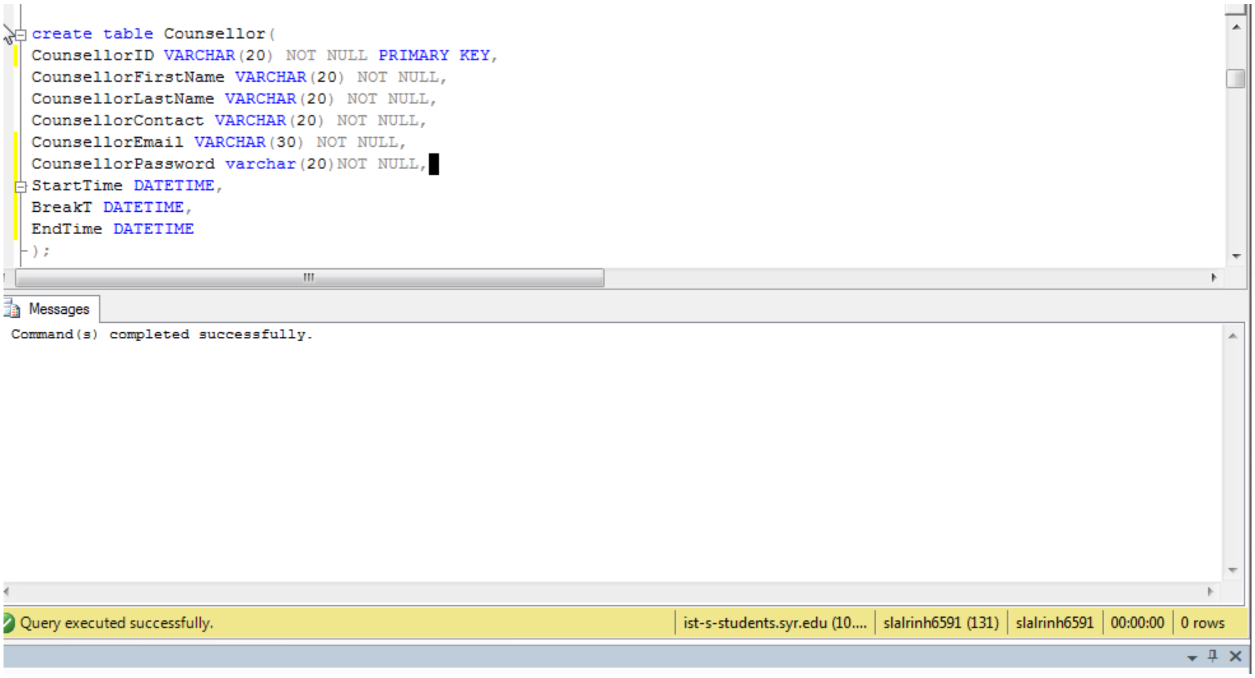
ent Studio\Projects\SQLQuery_Lalrinhlua_Samuel.sql - ist-s-students.syr.edu.slalrinh6591 (slalrinh6591 (117))

```
StudentLastName VARCHAR(20) NOT NULL,
StudentContact VARCHAR(20) NOT NULL,
StudentEmail VARCHAR(30) NOT NULL,
StudentAddress VARCHAR(30) NOT NULL,
StudentCity VARCHAR(30) NOT NULL,
SudentPassword varchar(20)NOT NULL
Accomodation VARCHAR(100) NOT NULL
);
```

```
Command(s) completed successfully.
```

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 0 rows

```
create table Counsellor(
CounsellorID VARCHAR(20) NOT NULL PRIMARY KEY,
CounsellorFirstName VARCHAR(20) NOT NULL,
CounsellorLastName VARCHAR(20) NOT NULL,
CounsellorContact VARCHAR(20) NOT NULL,
CounsellorEmail VARCHAR(30) NOT NULL,
CounsellorPassword varchar(20)NOT NULL,
StartTime DATETIME,
BreakT DATETIME,
EndTime DATETIME
);
```

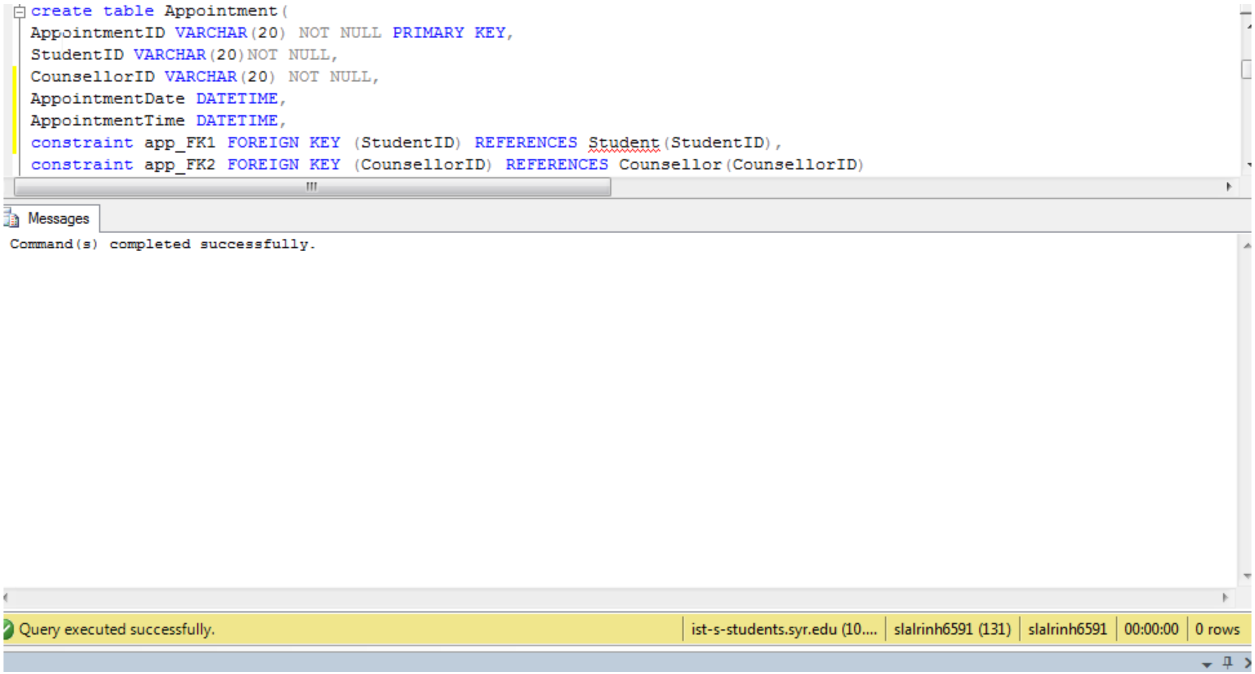


```
create table Counsellor(
CounsellorID VARCHAR(20) NOT NULL PRIMARY KEY,
CounsellorFirstName VARCHAR(20) NOT NULL,
CounsellorLastName VARCHAR(20) NOT NULL,
CounsellorContact VARCHAR(20) NOT NULL,
CounsellorEmail VARCHAR(30) NOT NULL,
CounsellorPassword varchar(20)NOT NULL,
StartTime DATETIME,
BreakT DATETIME,
EndTime DATETIME
);
```

```
Command(s) completed successfully.
```

```
Query executed successfully.   ist-s-students.syr.edu (10....   slalrinh6591 (131)   slalrinh6591   00:00:00   0 rows
```

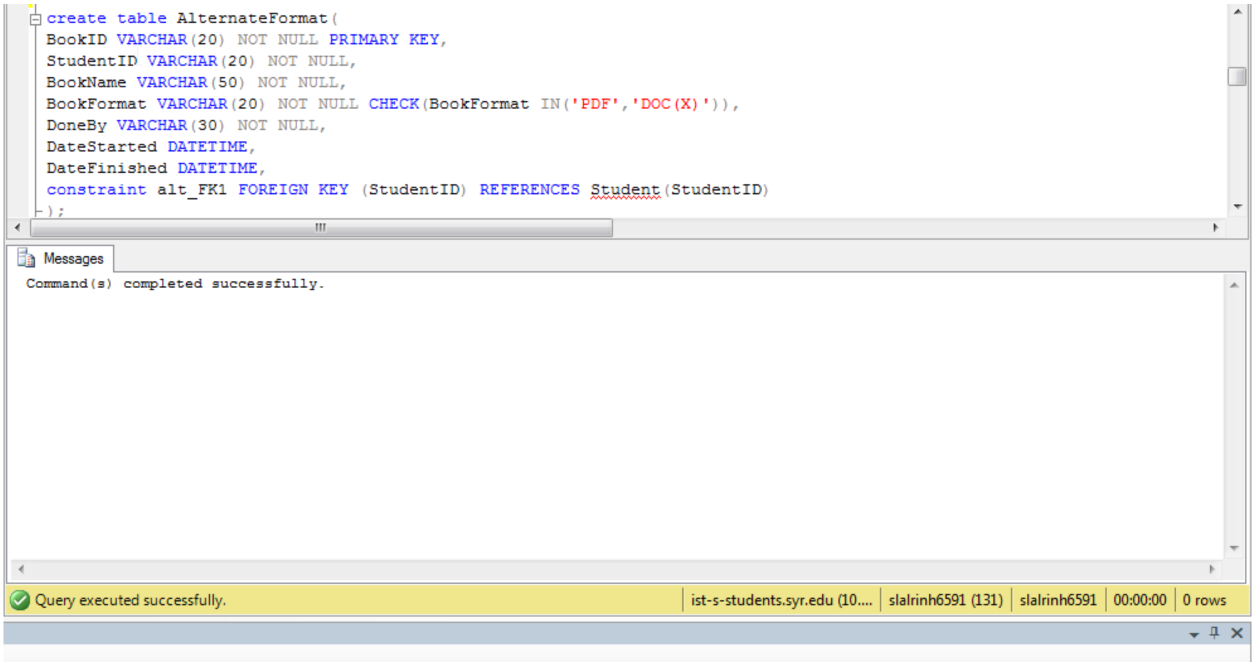
```
create table Appointment(
AppointmentID VARCHAR(20) NOT NULL PRIMARY KEY,
StudentID VARCHAR(20)NOT NULL,
CounsellorID VARCHAR(20) NOT NULL,
AppointmentDate DATETIME,
AppointmentTime DATETIME,
constraint app_FK1 FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
constraint app_FK2 FOREIGN KEY (CounsellorID) REFERENCES
Counsellor(CounsellorID)
);
```



```
create table Appointment(
AppointmentID VARCHAR(20) NOT NULL PRIMARY KEY,
StudentID VARCHAR(20)NOT NULL,
CounsellorID VARCHAR(20) NOT NULL,
AppointmentDate DATETIME,
AppointmentTime DATETIME,
constraint app_FK1 FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
constraint app_FK2 FOREIGN KEY (CounsellorID) REFERENCES Counsellor(CounsellorID)
```

```
Command(s) completed successfully.
```

```
Query executed successfully.   ist-s-students.syr.edu (10....   slalrinh6591 (131)   slalrinh6591   00:00:00   0 rows
create table AlternateFormat(
BookID VARCHAR(20) NOT NULL PRIMARY KEY,
StudentID VARCHAR(20) NOT NULL,
BookName VARCHAR(50) NOT NULL,
BookFormat VARCHAR(20) NOT NULL CHECK(BookFormat IN('PDF','DOC(X)')),
DoneBy VARCHAR(30) NOT NULL,
DateStarted DATETIME,
DateFinished DATETIME,
constraint alt_FK1 FOREIGN KEY (StudentID) REFERENCES Student(StudentID)
);
```

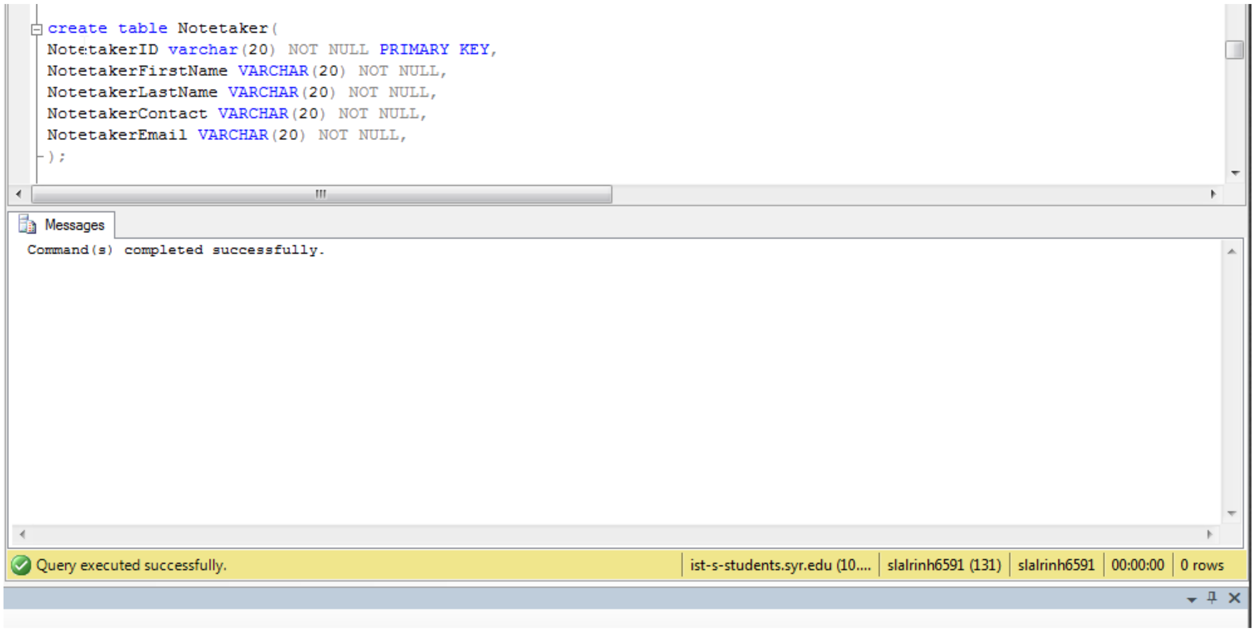


```
create table AlternateFormat(
BookID VARCHAR(20) NOT NULL PRIMARY KEY,
StudentID VARCHAR(20) NOT NULL,
BookName VARCHAR(50) NOT NULL,
BookFormat VARCHAR(20) NOT NULL CHECK(BookFormat IN('PDF','DOC(X)')),
DoneBy VARCHAR(30) NOT NULL,
DateStarted DATETIME,
DateFinished DATETIME,
constraint alt_FK1 FOREIGN KEY (StudentID) REFERENCES Student(StudentID)
);
```

```
Command(s) completed successfully.
```

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 0 rows

```
create table Notetaker(
NotetakerID varchar(20) NOT NULL PRIMARY KEY,
NotetakerFirstName VARCHAR(20) NOT NULL,
NotetakerLastName VARCHAR(20) NOT NULL,
NotetakerContact VARCHAR(20) NOT NULL,
NotetakerEmail VARCHAR(20) NOT NULL,
);
```

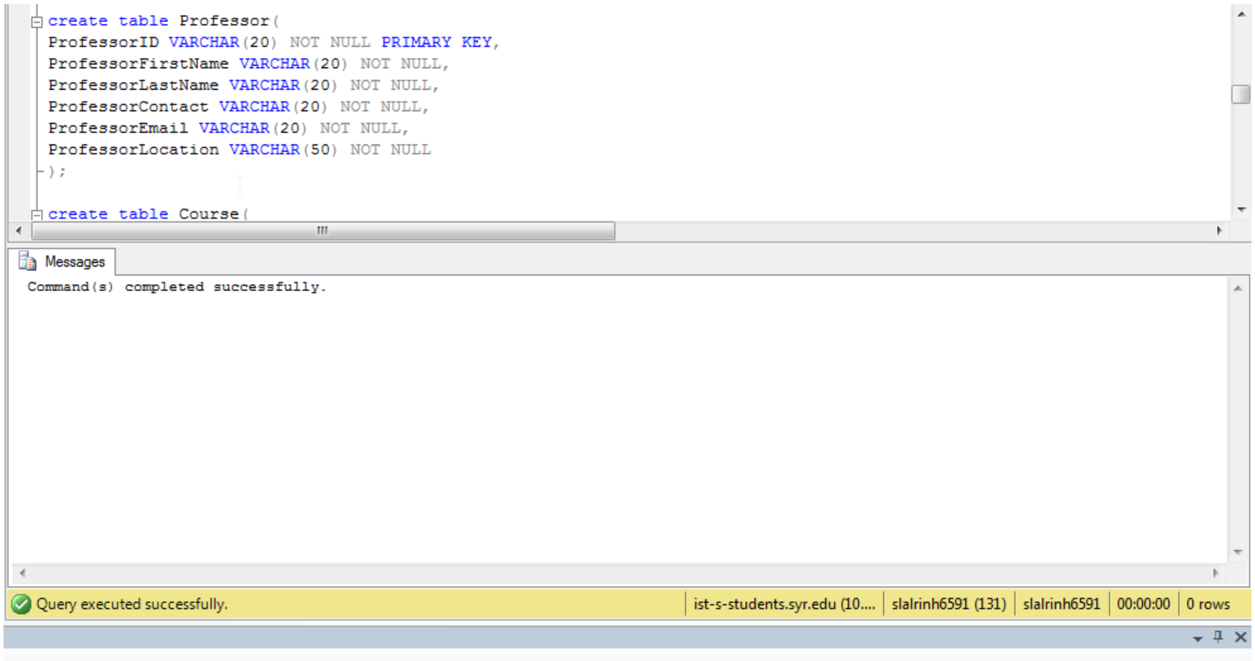


```
create table Notetaker(
NotetakerID varchar(20) NOT NULL PRIMARY KEY,
NotetakerFirstName VARCHAR(20) NOT NULL,
NotetakerLastName VARCHAR(20) NOT NULL,
NotetakerContact VARCHAR(20) NOT NULL,
NotetakerEmail VARCHAR(20) NOT NULL,
);
```

```
Command(s) completed successfully.
```

```
Query executed successfully.    ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 0 rows
```

```
create table Professor(
ProfessorID VARCHAR(20) NOT NULL PRIMARY KEY,
ProfessorFirstName VARCHAR(20) NOT NULL,
ProfessorLastName VARCHAR(20) NOT NULL,
ProfessorContact VARCHAR(20) NOT NULL,
ProfessorEmail VARCHAR(20) NOT NULL,
ProfessorLocation VARCHAR(50) NOT NULL
);
```

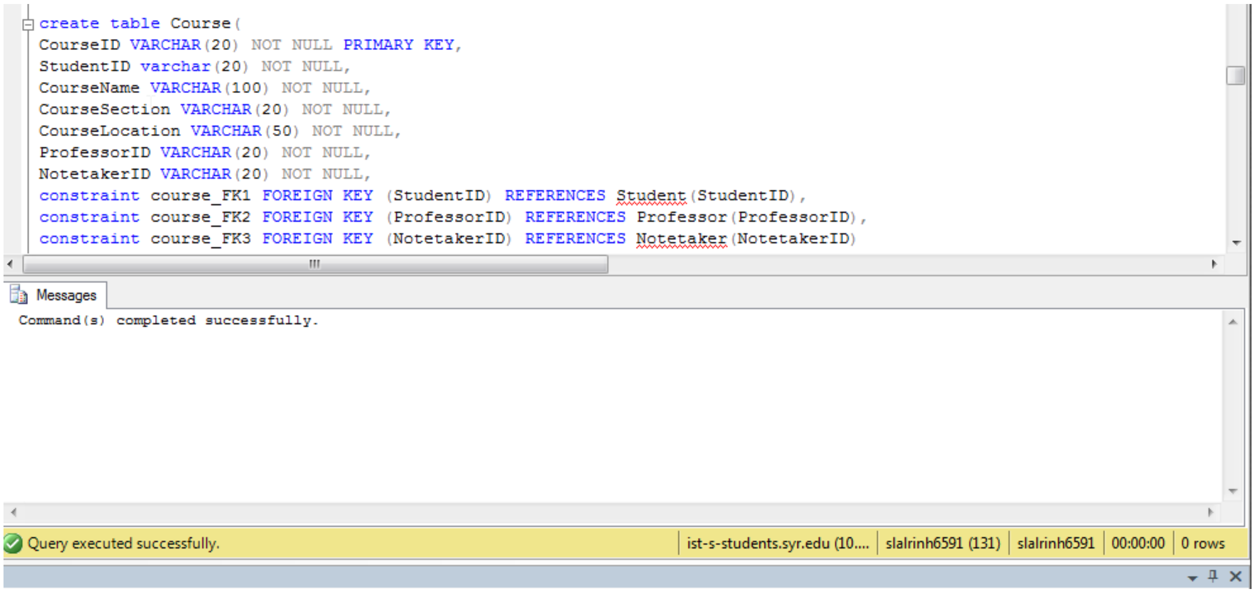


```
create table Professor(
ProfessorID VARCHAR(20) NOT NULL PRIMARY KEY,
ProfessorFirstName VARCHAR(20) NOT NULL,
ProfessorLastName VARCHAR(20) NOT NULL,
ProfessorContact VARCHAR(20) NOT NULL,
ProfessorEmail VARCHAR(20) NOT NULL,
ProfessorLocation VARCHAR(50) NOT NULL
);

create table Course(
```

```
Command(s) completed successfully.
```

```
Query executed successfully.    ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 0 rows
```

```
create table Course(
CourseID VARCHAR(20) NOT NULL PRIMARY KEY,
StudentID varchar(20) NOT NULL,
CourseName VARCHAR(100) NOT NULL,
CourseSection VARCHAR(20) NOT NULL,
CourseLocation VARCHAR(50) NOT NULL,
ProfessorID VARCHAR(20) NOT NULL,
NotetakerID VARCHAR(20) NOT NULL,
constraint course_FK1 FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
constraint course_FK2 FOREIGN KEY (ProfessorID) REFERENCES
Professor(ProfessorID),
constraint course_FK3 FOREIGN KEY (NotetakerID) REFERENCES
Notetaker(NotetakerID)
);
```

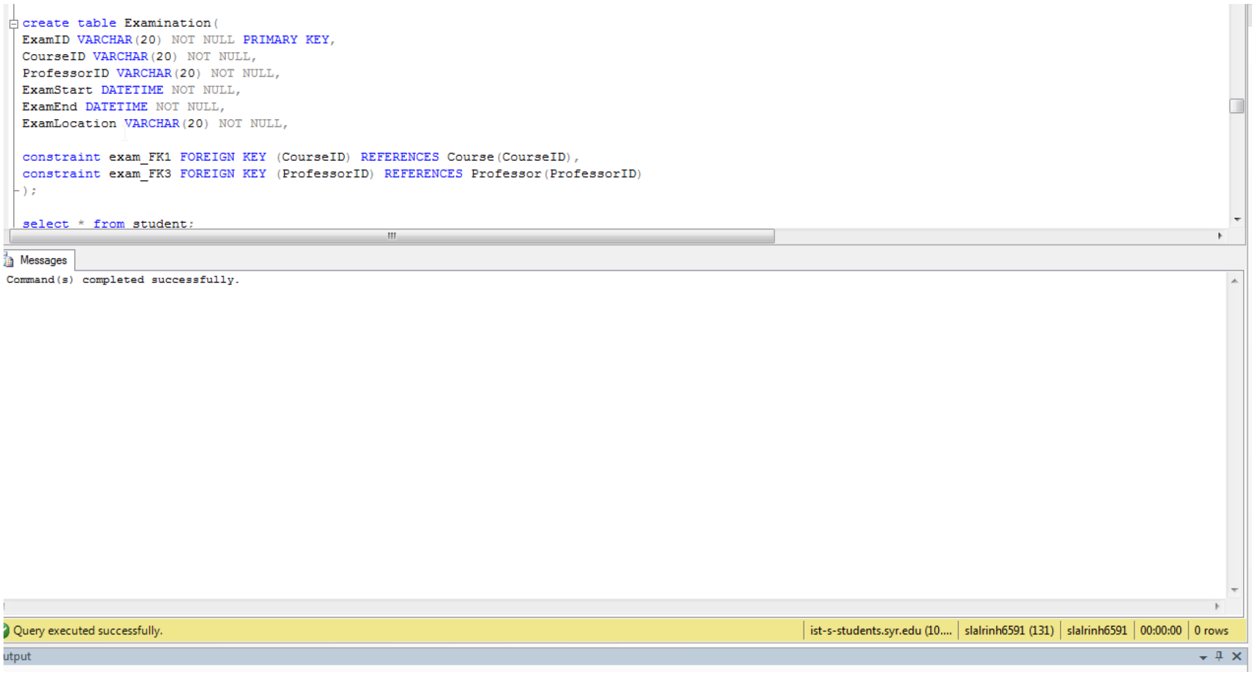


```
create table Course(
CourseID VARCHAR(20) NOT NULL PRIMARY KEY,
StudentID varchar(20) NOT NULL,
CourseName VARCHAR(100) NOT NULL,
CourseSection VARCHAR(20) NOT NULL,
CourseLocation VARCHAR(50) NOT NULL,
ProfessorID VARCHAR(20) NOT NULL,
NotetakerID VARCHAR(20) NOT NULL,
constraint course_FK1 FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
constraint course_FK2 FOREIGN KEY (ProfessorID) REFERENCES Professor(ProfessorID),
constraint course_FK3 FOREIGN KEY (NotetakerID) REFERENCES Notetaker(NotetakerID)
```

```
Command(s) completed successfully.
```

Query executed successfully.

```
create table Examination(
ExamID VARCHAR(20) NOT NULL PRIMARY KEY,
CourseID VARCHAR(20) NOT NULL,
ProfessorID VARCHAR(20) NOT NULL,
ExamStart DATETIME NOT NULL,
ExamEnd DATETIME NOT NULL,
ExamLocation VARCHAR(20) NOT NULL,

constraint exam_FK1 FOREIGN KEY (CourseID) REFERENCES Course(CourseID),
constraint exam_FK3 FOREIGN KEY (ProfessorID) REFERENCES
Professor(ProfessorID)
);
```

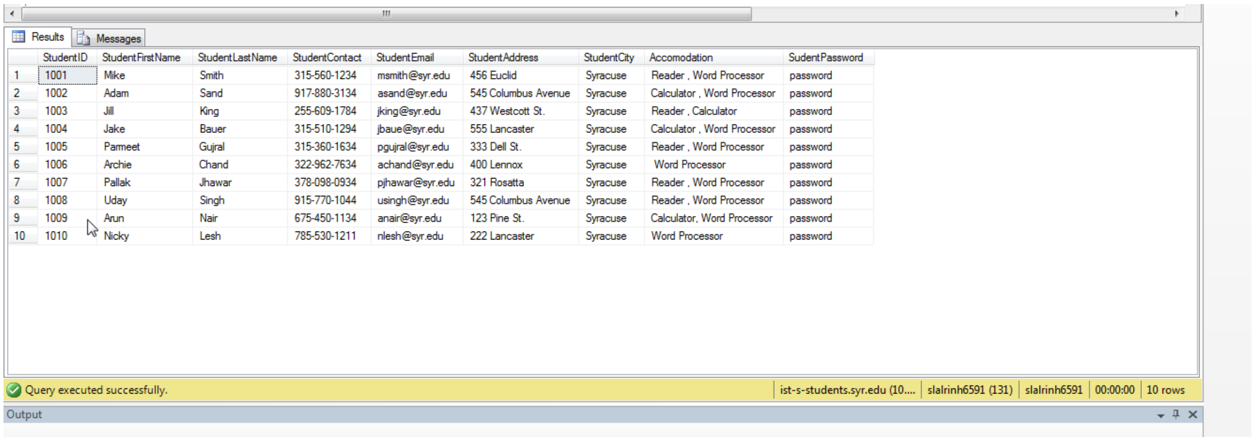


```sql
create table Examination(
ExamID VARCHAR(20) NOT NULL PRIMARY KEY,
CourseID VARCHAR(20) NOT NULL,
ProfessorID VARCHAR(20) NOT NULL,
ExamStart DATETIME NOT NULL,
ExamEnd DATETIME NOT NULL,
ExamLocation VARCHAR(20) NOT NULL,

constraint exam_FK1 FOREIGN KEY (CourseID) REFERENCES Course(CourseID),
constraint exam_FK3 FOREIGN KEY (ProfessorID) REFERENCES Professor(ProfessorID)
);

select * from student;
```

```
Command(s) completed successfully.
```

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 0 rows

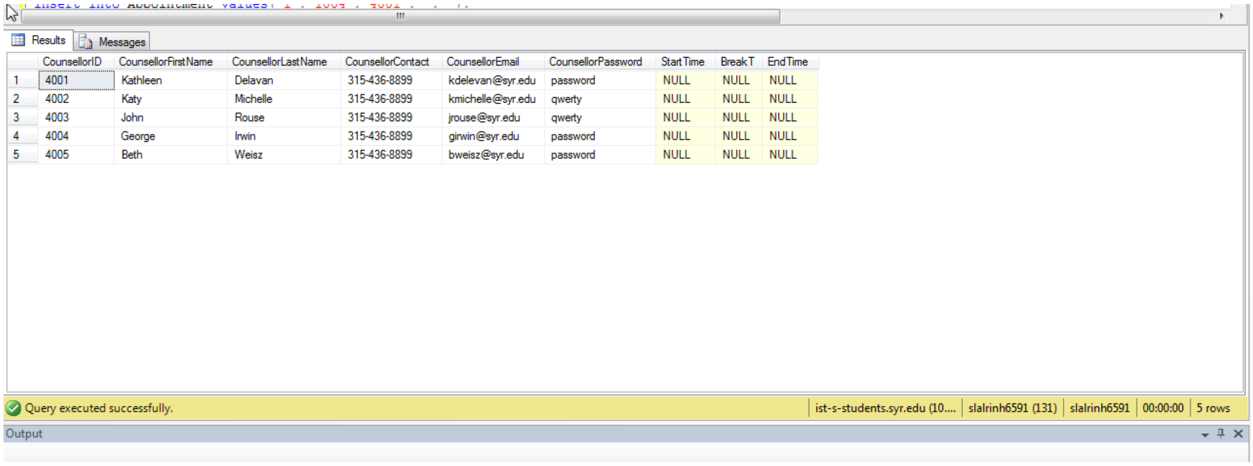
```sql
-- populating the student table with data

insert into Student values ('1001', 'Mike', 'Smith', '315-560-1234',
'msmith@syr.edu', '456 Euclid','Syracuse', 'Reader , Word
Processor','password');
insert into Student values ('1002', 'Adam', 'Sand', '917-880-3134',
'asand@syr.edu', '545 Columbus Avenue','Syracuse', 'Calculator , Word
Processor','password');
insert into Student values ('1003', 'Jill', 'King', '255-609-1784',
'jking@syr.edu', '437 Westcott St.', 'Syracuse','Reader ,
Calculator','password');
insert into Student values ('1004', 'Jake', 'Bauer', '315-510-1294',
'jbaue@syr.edu', '555 Lancaster', 'Syracuse','Calculator , Word
Processor','password');
insert into Student values ('1005', 'Parmeet', 'Gujral', '315-360-1634',
'pgujral@syr.edu', '333 Dell St.','Syracuse', 'Reader , Word
Processor','password');
insert into Student values ('1006', 'Archie', 'Chand', '322-962-7634',
'achand@syr.edu', '400 Lennox', 'Syracuse',' Word Processor','password');
insert into Student values ('1007', 'Pallak', 'Jhawar', '378-098-0934',
'pjhawar@syr.edu', '321 Rosatta', 'Syracuse','Reader , Word
Processor','password');
insert into Student values ('1008', 'Uday', 'Singh', '915-770-1044',
'usingh@syr.edu', '545 Columbus Avenue','Syracuse', 'Reader , Word
Processor','password');
insert into Student values ('1009', 'Arun', 'Nair', '675-450-1134',
'anair@syr.edu', '123 Pine St.','Syracuse', 'Calculator, Word
Processor','password');
insert into Student values ('1010', 'Nicky', 'Lesh', '785-530-1211',
'nlesh@syr.edu', '222 Lancaster','Syracuse', 'Word Processor','password');


select * from student;
```



	StudentID	StudentFirstName	StudentLastName	StudentContact	StudentEmail	StudentAddress	StudentCity	Accomodation	SudentPassword
1	1001	Mike	Smith	315-560-1234	msmith@syr.edu	456 Euclid	Syracuse	Reader , Word Processor	password
2	1002	Adam	Sand	917-880-3134	asand@syr.edu	545 Columbus Avenue	Syracuse	Calculator , Word Processor	password
3	1003	Jill	King	255-609-1784	jking@syr.edu	437 Westcott St.	Syracuse	Reader , Calculator	password
4	1004	Jake	Bauer	315-510-1294	jbaue@syr.edu	555 Lancaster	Syracuse	Calculator , Word Processor	password
5	1005	Parmeet	Gujral	315-360-1634	pgujral@syr.edu	333 Dell St.	Syracuse	Reader , Word Processor	password
6	1006	Archie	Chand	322-962-7634	achand@syr.edu	400 Lennox	Syracuse	Word Processor	password
7	1007	Pallak	Jhawar	378-098-0934	pjhawar@syr.edu	321 Rosatta	Syracuse	Reader , Word Processor	password
8	1008	Uday	Singh	915-770-1044	usingh@syr.edu	545 Columbus Avenue	Syracuse	Reader , Word Processor	password
9	1009	Arun	Nair	675-450-1134	anair@syr.edu	123 Pine St.	Syracuse	Calculator, Word Processor	password
10	1010	Nicky	Lesh	785-530-1211	nlesh@syr.edu	222 Lancaster	Syracuse	Word Processor	password

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 10 rows

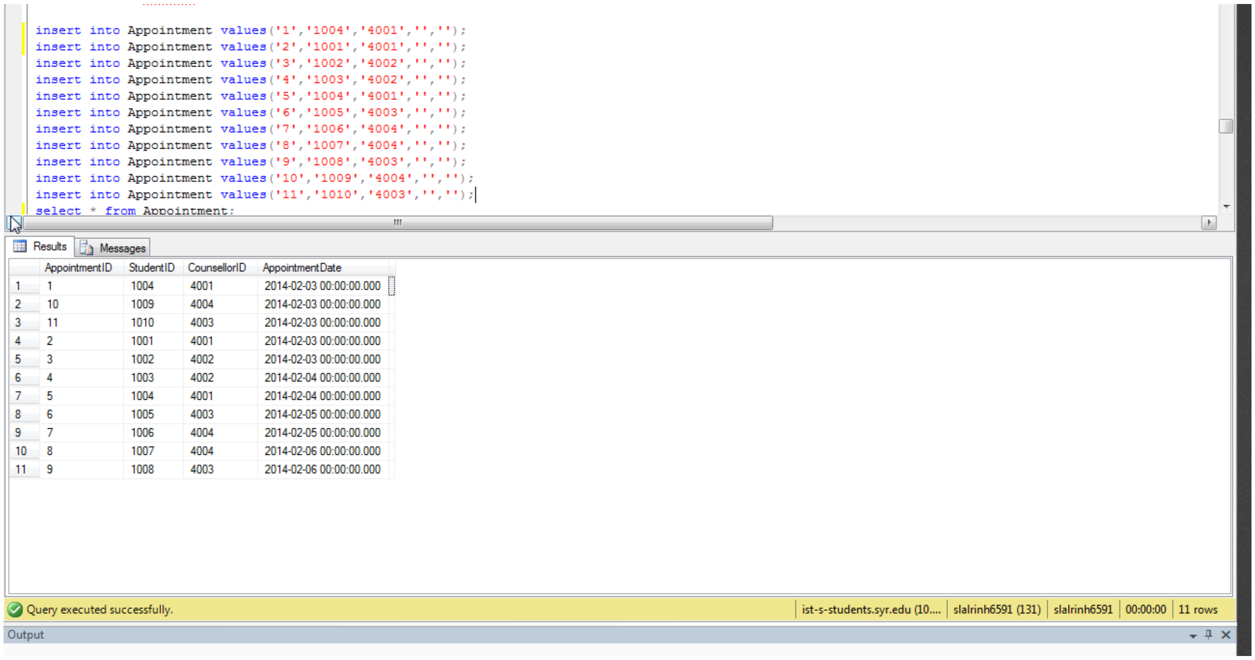
```
-- populating the Counsellor table with data

insert into Counsellor (CounsellorID, CounsellorFirstName,
CounsellorLastName, CounsellorContact, CounsellorEmail, CounsellorPassword)
values ('4001', 'Kathleen','Delavan','315-436-8899', 'kdelevan@syr.edu',
'password');
insert into Counsellor (CounsellorID, CounsellorFirstName,
CounsellorLastName, CounsellorContact, CounsellorEmail, CounsellorPassword)
values ('4002', 'Katy','Michelle','315-436-8899', 'kmichelle@syr.edu',
'qwerty');
insert into Counsellor (CounsellorID, CounsellorFirstName,
CounsellorLastName, CounsellorContact, CounsellorEmail, CounsellorPassword)
values ('4003', 'John','Rouse','315-436-8899', 'jrouse@syr.edu', 'qwerty');
insert into Counsellor (CounsellorID, CounsellorFirstName,
CounsellorLastName, CounsellorContact, CounsellorEmail, CounsellorPassword)
values ('4004', 'George','Irwin','315-436-8899',
'girwin@syr.edu','password');
insert into Counsellor (CounsellorID, CounsellorFirstName,
CounsellorLastName, CounsellorContact, CounsellorEmail, CounsellorPassword)
values ('4005', 'Beth','Weisz','315-436-8899', 'bweisz@syr.edu','password');
```



	CounsellorID	CounsellorFirstName	CounsellorLastName	CounsellorContact	CounsellorEmail	CounsellorPassword	StartTime	BreakT	EndTime
1	4001	Kathleen	Delavan	315-436-8899	kdelevan@syr.edu	password	NULL	NULL	NULL
2	4002	Katy	Michelle	315-436-8899	kmichelle@syr.edu	qwerty	NULL	NULL	NULL
3	4003	John	Rouse	315-436-8899	jrouse@syr.edu	qwerty	NULL	NULL	NULL
4	4004	George	Irwin	315-436-8899	girwin@syr.edu	password	NULL	NULL	NULL
5	4005	Beth	Weisz	315-436-8899	bweisz@syr.edu	password	NULL	NULL	NULL

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 5 rows

```sql
-- populating the Appointment table with data

insert into Appointment values('1','1004','4001','','');
insert into Appointment values('2','1001','4001','','');
insert into Appointment values('3','1002','4002','','');
insert into Appointment values('4','1003','4002','','');
insert into Appointment values('5','1004','4001','','');
insert into Appointment values('6','1005','4003','','');
insert into Appointment values('7','1006','4004','','');
insert into Appointment values('8','1007','4004','','');
insert into Appointment values('9','1008','4003','','');
insert into Appointment values('10','1009','4004','','');
insert into Appointment values('11','1010','4003','','');
```

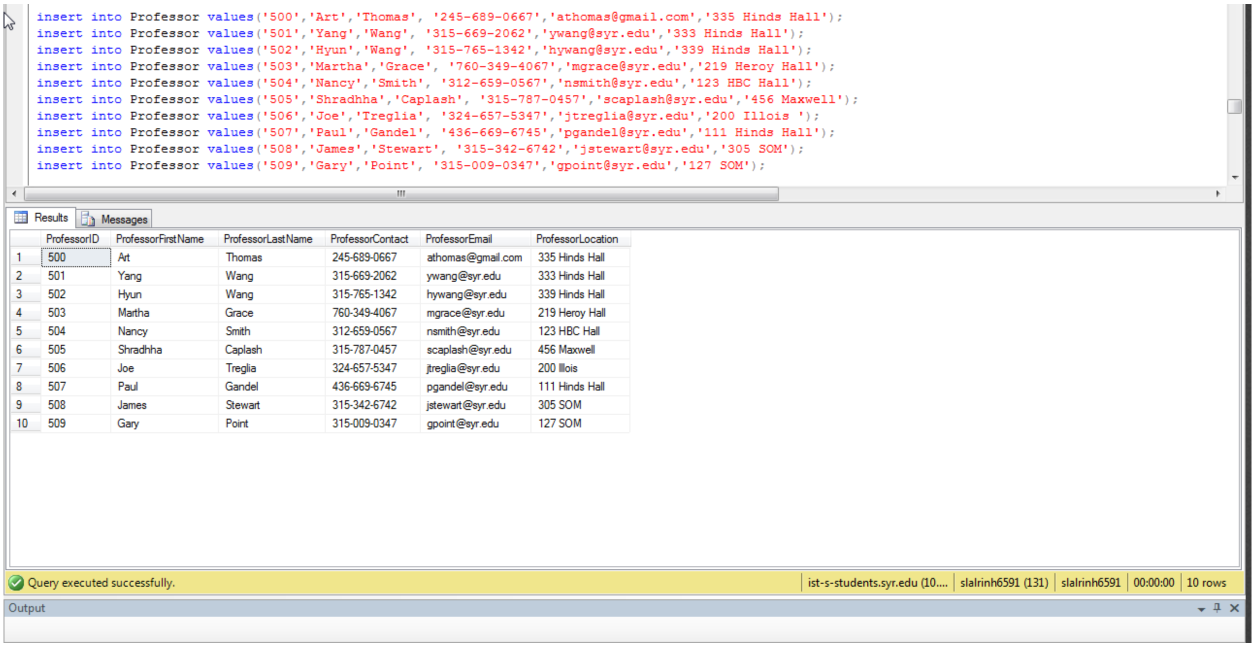


```sql
insert into Appointment values('1','1004','4001','','');
insert into Appointment values('2','1001','4001','','');
insert into Appointment values('3','1002','4002','','');
insert into Appointment values('4','1003','4002','','');
insert into Appointment values('5','1004','4001','','');
insert into Appointment values('6','1005','4003','','');
insert into Appointment values('7','1006','4004','','');
insert into Appointment values('8','1007','4004','','');
insert into Appointment values('9','1008','4003','','');
insert into Appointment values('10','1009','4004','','');
insert into Appointment values('11','1010','4003','','');
select * from Appointment;
```

	AppointmentID	StudentID	CounsellorID	AppointmentDate
1	1	1004	4001	2014-02-03 00:00:00.000
2	10	1009	4004	2014-02-03 00:00:00.000
3	11	1010	4003	2014-02-03 00:00:00.000
4	2	1001	4001	2014-02-03 00:00:00.000
5	3	1002	4002	2014-02-03 00:00:00.000
6	4	1003	4002	2014-02-04 00:00:00.000
7	5	1004	4001	2014-02-04 00:00:00.000
8	6	1005	4003	2014-02-05 00:00:00.000
9	7	1006	4004	2014-02-05 00:00:00.000
10	8	1007	4004	2014-02-06 00:00:00.000
11	9	1008	4003	2014-02-06 00:00:00.000

```
Query executed successfully.    ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 11 rows
```

```sql
-- populating the Professor table with data

insert into Professor values('500','Art','Thomas', '245-689-0667','athomas@gmail.com','335 Hinds Hall');
insert into Professor values('501','Yang','Wang', '315-669-2062','ywang@syr.edu','333 Hinds Hall');
insert into Professor values('502','Hyun','Wang', '315-765-1342','hywang@syr.edu','339 Hinds Hall');
insert into Professor values('503','Martha','Grace', '760-349-4067','mgrace@syr.edu','219 Heroy Hall');
insert into Professor values('504','Nancy','Smith', '312-659-0567','nsmith@syr.edu','123 HBC Hall');
insert into Professor values('505','Shradhha','Caplash', '315-787-0457','scaplash@syr.edu','456 Maxwell');
insert into Professor values('506','Joe','Treglia', '324-657-5347','jtreglia@syr.edu','200 Illois ');
insert into Professor values('507','Paul','Gandel', '436-669-6745','pgandel@syr.edu','111 Hinds Hall');
insert into Professor values('508','James','Stewart', '315-342-
6742','jstewart@syr.edu','305 SOM');
insert into Professor values('509','Gary','Point', '315-009-
0347','gpoint@syr.edu','127 SOM');

select * from Professor;
```

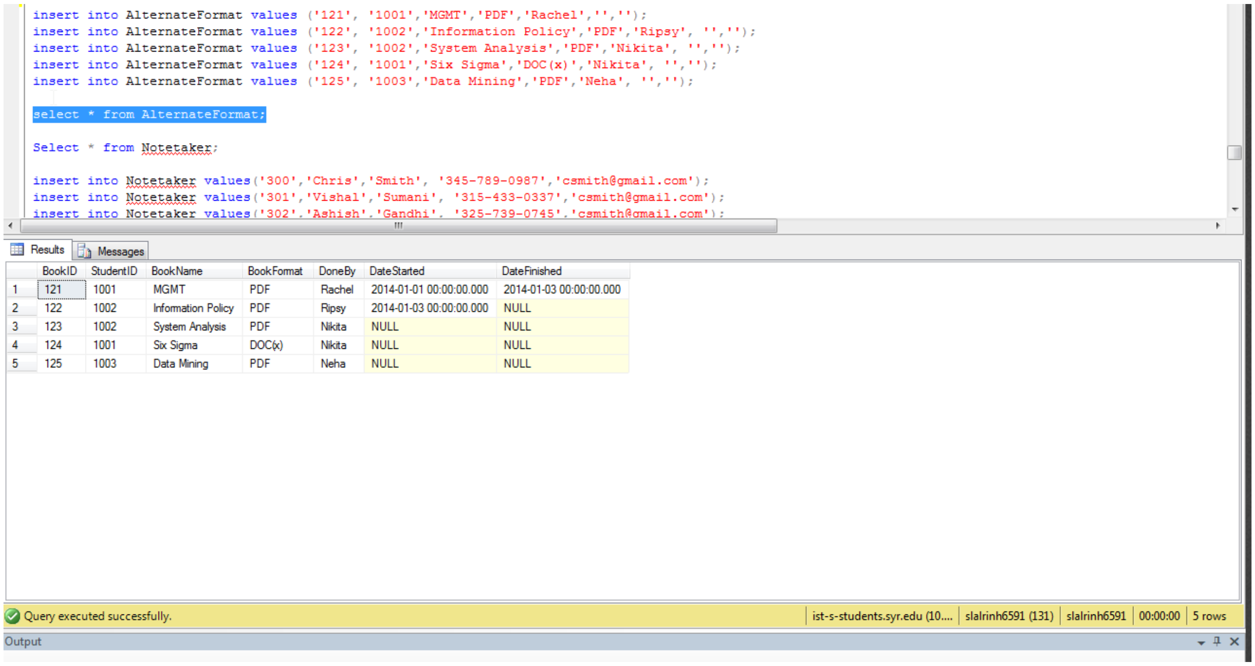


```
insert into Professor values('500','Art','Thomas', '245-689-0667','athomas@gmail.com','335 Hinds Hall');
insert into Professor values('501','Yang','Wang', '315-669-2062','ywang@syr.edu','333 Hinds Hall');
insert into Professor values('502','Hyun','Wang', '315-765-1342','hywang@syr.edu','339 Hinds Hall');
insert into Professor values('503','Martha','Grace', '760-349-4067','mgrace@syr.edu','219 Heroy Hall');
insert into Professor values('504','Nancy','Smith', '312-659-0567','nsmith@syr.edu','123 HBC Hall');
insert into Professor values('505','Shradhha','Caplash', '315-787-0457','scaplash@syr.edu','456 Maxwell');
insert into Professor values('506','Joe','Treglia', '324-657-5347','jtreglia@syr.edu','200 Illois ');
insert into Professor values('507','Paul','Gandel', '436-669-6745','pgandel@syr.edu','111 Hinds Hall');
insert into Professor values('508','James','Stewart', '315-342-6742','jstewart@syr.edu','305 SOM');
insert into Professor values('509','Gary','Point', '315-009-0347','gpoint@syr.edu','127 SOM');
```

	ProfessorID	ProfessorFirstName	ProfessorLastName	ProfessorContact	ProfessorEmail	ProfessorLocation
1	500	Art	Thomas	245-689-0667	athomas@gmail.com	335 Hinds Hall
2	501	Yang	Wang	315-669-2062	ywang@syr.edu	333 Hinds Hall
3	502	Hyun	Wang	315-765-1342	hywang@syr.edu	339 Hinds Hall
4	503	Martha	Grace	760-349-4067	mgrace@syr.edu	219 Heroy Hall
5	504	Nancy	Smith	312-659-0567	nsmith@syr.edu	123 HBC Hall
6	505	Shradhha	Caplash	315-787-0457	scaplash@syr.edu	456 Maxwell
7	506	Joe	Treglia	324-657-5347	jtreglia@syr.edu	200 Illois
8	507	Paul	Gandel	436-669-6745	pgandel@syr.edu	111 Hinds Hall
9	508	James	Stewart	315-342-6742	jstewart@syr.edu	305 SOM
10	509	Gary	Point	315-009-0347	gpoint@syr.edu	127 SOM

```
Query executed successfully.   ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 10 rows
```

```
-- populating the Alternate Format table with data

insert into AlternateFormat values ('121',
'1001','MGMT','PDF','Rachel','','');
insert into AlternateFormat values ('122', '1002','Information
Policy','PDF','Ripsy', '','');
insert into AlternateFormat values ('123', '1002','System
Analysis','PDF','Nikita', '','');
insert into AlternateFormat values ('124', '1001','Six
Sigma','DOC(x)','Nikita', '','');
insert into AlternateFormat values ('125', '1003','Data Mining','PDF','Neha',
'','');

select * from AlternateFormat;
```

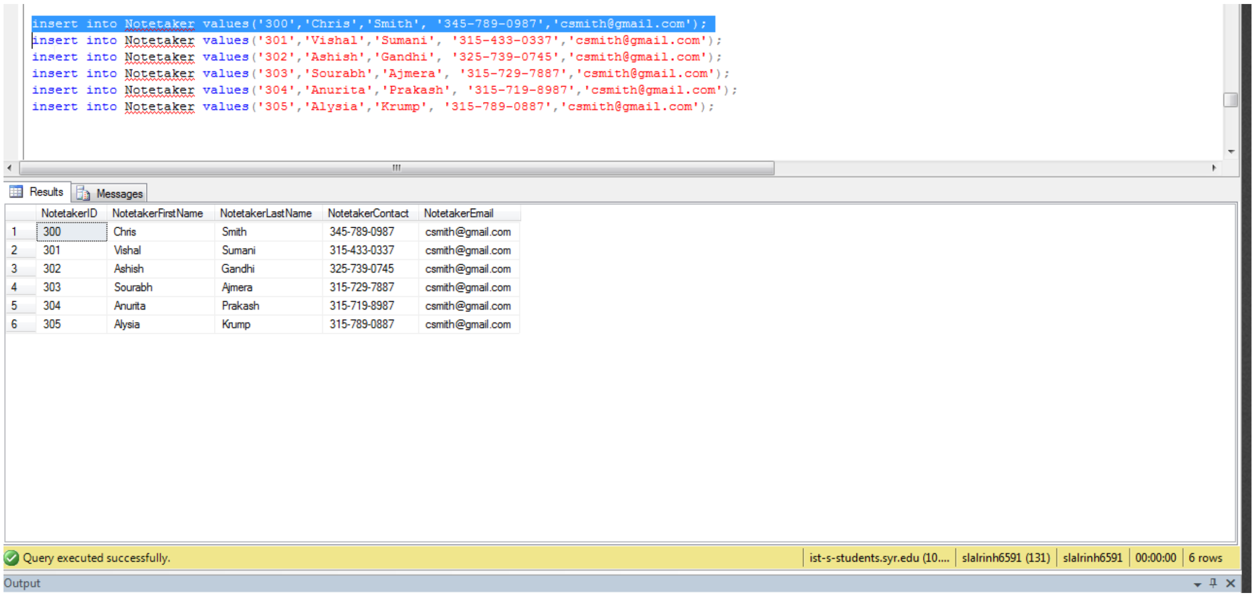


```sql
insert into AlternateFormat values ('121', '1001','MGMT','PDF','Rachel','','');
insert into AlternateFormat values ('122', '1002','Information Policy','PDF','Ripsy', '','');
insert into AlternateFormat values ('123', '1002','System Analysis','PDF','Nikita', '','');
insert into AlternateFormat values ('124', '1001','Six Sigma','DOC(x)','Nikita', '','');
insert into AlternateFormat values ('125', '1003','Data Mining','PDF','Neha', '','');

select * from AlternateFormat;

Select * from Notetaker;

insert into Notetaker values('300','Chris','Smith', '345-789-0987','csmith@gmail.com');
insert into Notetaker values('301','Vishal','Sumani', '315-433-0337','csmith@gmail.com');
insert into Notetaker values('302','Ashish','Gandhi', '325-739-0745','csmith@gmail.com');
```

	BookID	StudentID	BookName	BookFormat	DoneBy	DateStarted	DateFinished
1	121	1001	MGMT	PDF	Rachel	2014-01-01 00:00:00.000	2014-01-03 00:00:00.000
2	122	1002	Information Policy	PDF	Ripsy	2014-01-03 00:00:00.000	NULL
3	123	1002	System Analysis	PDF	Nikita	NULL	NULL
4	124	1001	Six Sigma	DOC(x)	Nikita	NULL	NULL
5	125	1003	Data Mining	PDF	Neha	NULL	NULL

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 5 rows

```sql
-- populating the Notetaker table with data
insert into Notetaker values('300','Chris','Smith', '345-789-0987','csmith@gmail.com');
insert into Notetaker values('301','Vishal','Sumani', '315-433-0337','csmith@gmail.com');
insert into Notetaker values('302','Ashish','Gandhi', '325-739-0745','csmith@gmail.com');
insert into Notetaker values('303','Sourabh','Ajmera', '315-729-7887','csmith@gmail.com');
insert into Notetaker values('304','Anurita','Prakash', '315-719-8987','csmith@gmail.com');
insert into Notetaker values('305','Alysia','Krump', '315-789-0887','csmith@gmail.com');
```

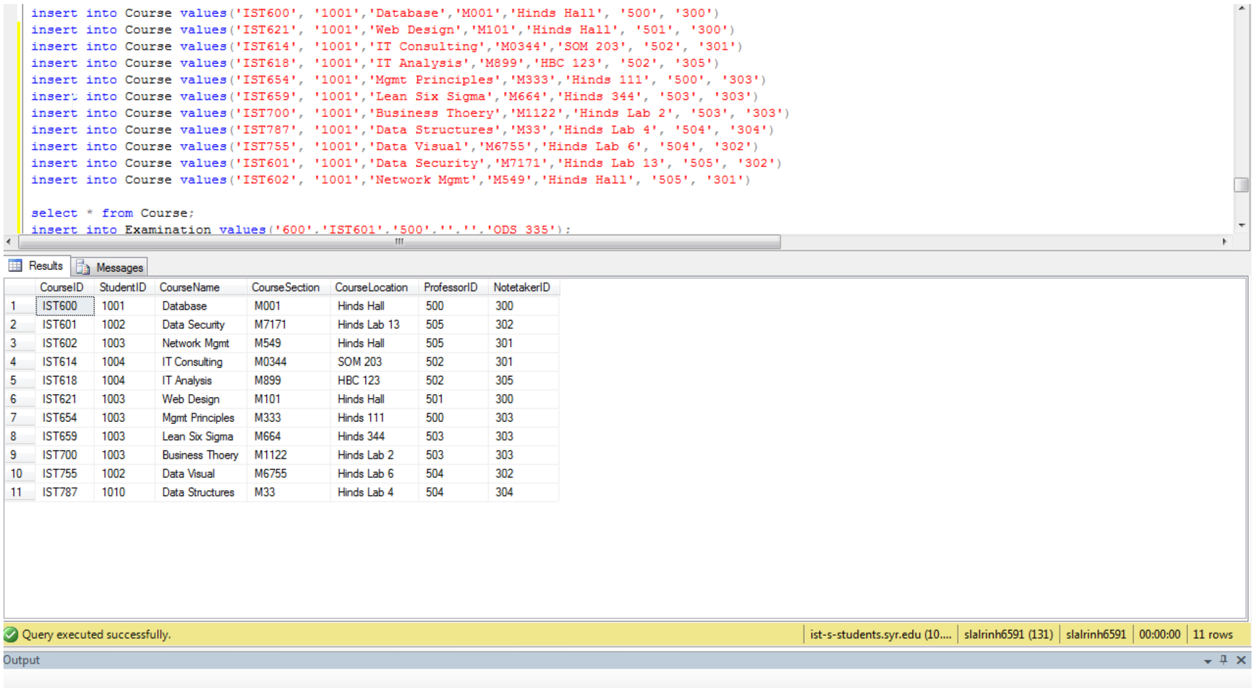


```sql
insert into Notetaker values('300','Chris','Smith', '345-789-0987','csmith@gmail.com');
insert into Notetaker values('301','Vishal','Sumani', '315-433-0337','csmith@gmail.com');
insert into Notetaker values('302','Ashish','Gandhi', '325-739-0745','csmith@gmail.com');
insert into Notetaker values('303','Sourabh','Ajmera', '315-729-7887','csmith@gmail.com');
insert into Notetaker values('304','Anurita','Prakash', '315-719-8987','csmith@gmail.com');
insert into Notetaker values('305','Alysia','Krump', '315-789-0887','csmith@gmail.com');
```

	NotetakerID	NotetakerFirstName	NotetakerLastName	NotetakerContact	NotetakerEmail
1	300	Chris	Smith	345-789-0987	csmith@gmail.com
2	301	Vishal	Sumani	315-433-0337	csmith@gmail.com
3	302	Ashish	Gandhi	325-739-0745	csmith@gmail.com
4	303	Sourabh	Ajmera	315-729-7887	csmith@gmail.com
5	304	Anurita	Prakash	315-719-8987	csmith@gmail.com
6	305	Alysia	Krump	315-789-0887	csmith@gmail.com

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 6 rows

```sql
-- populating the Course table with data

insert into Course values('IST600', '1001','Database','M001','Hinds Hall',
'500', '300')
insert into Course values('IST621', '1001','Web Design','M101','Hinds Hall',
'501', '300')
insert into Course values('IST614', '1001','IT Consulting','M0344','SOM 203',
'502', '301')
insert into Course values('IST618', '1001','IT Analysis','M899','HBC 123',
'502', '305')
insert into Course values('IST654', '1001','Mgmt Principles','M333','Hinds
111', '500', '303')
insert into Course values('IST659', '1001','Lean Six Sigma','M664','Hinds
344', '503', '303')
insert into Course values('IST700', '1001','Business Thoery','M1122','Hinds
Lab 2', '503', '303')
insert into Course values('IST787', '1001','Data Structures','M33','Hinds Lab
4', '504', '304')
insert into Course values('IST755', '1001','Data Visual','M6755','Hinds Lab
6', '504', '302')
insert into Course values('IST601', '1001','Data Security','M7171','Hinds Lab
13', '505', '302')
insert into Course values('IST602', '1001','Network Mgmt','M549','Hinds
Hall', '505', '301')
```

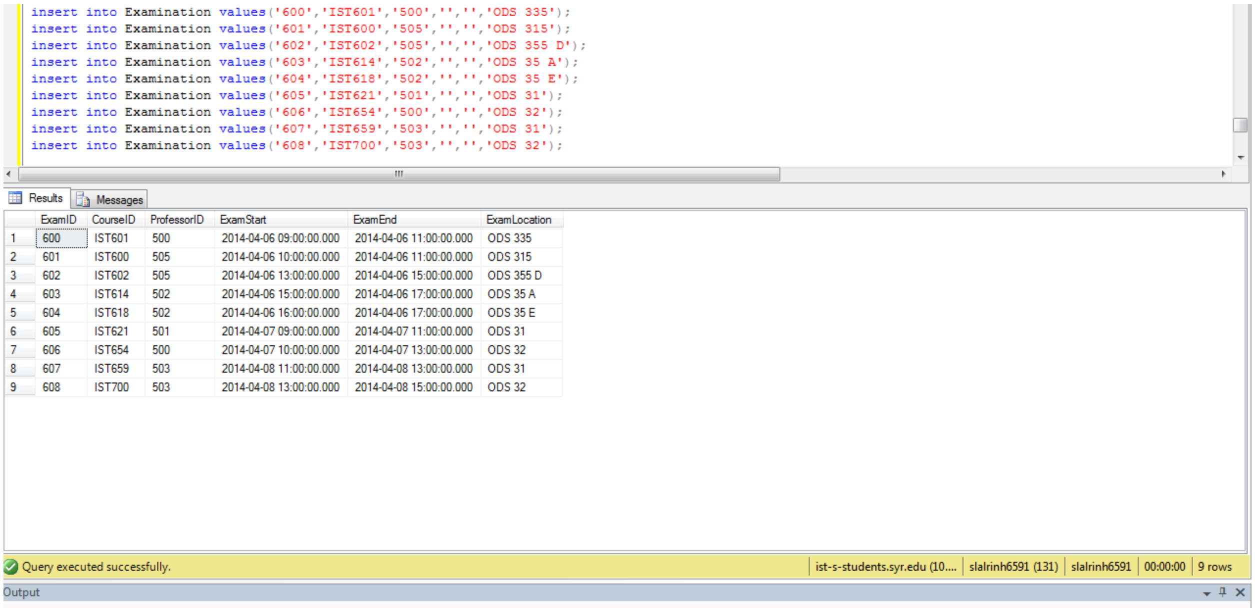


```
insert into Course values('IST600', '1001','Database','M001','Hinds Hall', '500', '300')
insert into Course values('IST621', '1001','Web Design','M101','Hinds Hall', '501', '300')
insert into Course values('IST614', '1001','IT Consulting','M0344','SOM 203', '502', '301')
insert into Course values('IST618', '1001','IT Analysis','M899','HBC 123', '502', '305')
insert into Course values('IST654', '1001','Mgmt Principles','M333','Hinds 111', '500', '303')
insert into Course values('IST659', '1001','Lean Six Sigma','M664','Hinds 344', '503', '303')
insert into Course values('IST700', '1001','Business Thoery','M1122','Hinds Lab 2', '503', '303')
insert into Course values('IST787', '1001','Data Structures','M33','Hinds Lab 4', '504', '304')
insert into Course values('IST755', '1001','Data Visual','M6755','Hinds Lab 6', '504', '302')
insert into Course values('IST601', '1001','Data Security','M7171','Hinds Lab 13', '505', '302')
insert into Course values('IST602', '1001','Network Mgmt','M549','Hinds Hall', '505', '301')

select * from Course;
insert into Examination values('600','IST601','500','','','ODS 335');
```

	CourseID	StudentID	CourseName	CourseSection	CourseLocation	ProfessorID	NotetakerID
1	IST600	1001	Database	M001	Hinds Hall	500	300
2	IST601	1002	Data Security	M7171	Hinds Lab 13	505	302
3	IST602	1003	Network Mgmt	M549	Hinds Hall	505	301
4	IST614	1004	IT Consulting	M0344	SOM 203	502	301
5	IST618	1004	IT Analysis	M899	HBC 123	502	305
6	IST621	1003	Web Design	M101	Hinds Hall	501	300
7	IST654	1003	Mgmt Principles	M333	Hinds 111	500	303
8	IST659	1003	Lean Six Sigma	M664	Hinds 344	503	303
9	IST700	1003	Business Thoery	M1122	Hinds Lab 2	503	303
10	IST755	1002	Data Visual	M6755	Hinds Lab 6	504	302
11	IST787	1010	Data Structures	M33	Hinds Lab 4	504	304

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 11 rows

```
-- populating theExamination table with data
insert into Examination values('600','IST601','500','','','ODS 335');
insert into Examination values('601','IST600','505','','','ODS 315');
insert into Examination values('602','IST602','505','','','ODS 355 D');
insert into Examination values('603','IST614','502','','','ODS 35 A');
insert into Examination values('604','IST618','502','','','ODS 35 E');
insert into Examination values('605','IST621','501','','','ODS 31');
insert into Examination values('606','IST654','500','','','ODS 32');
insert into Examination values('607','IST659','503','','','ODS 31');
insert into Examination values('608','IST700','503','','','ODS 32');
```



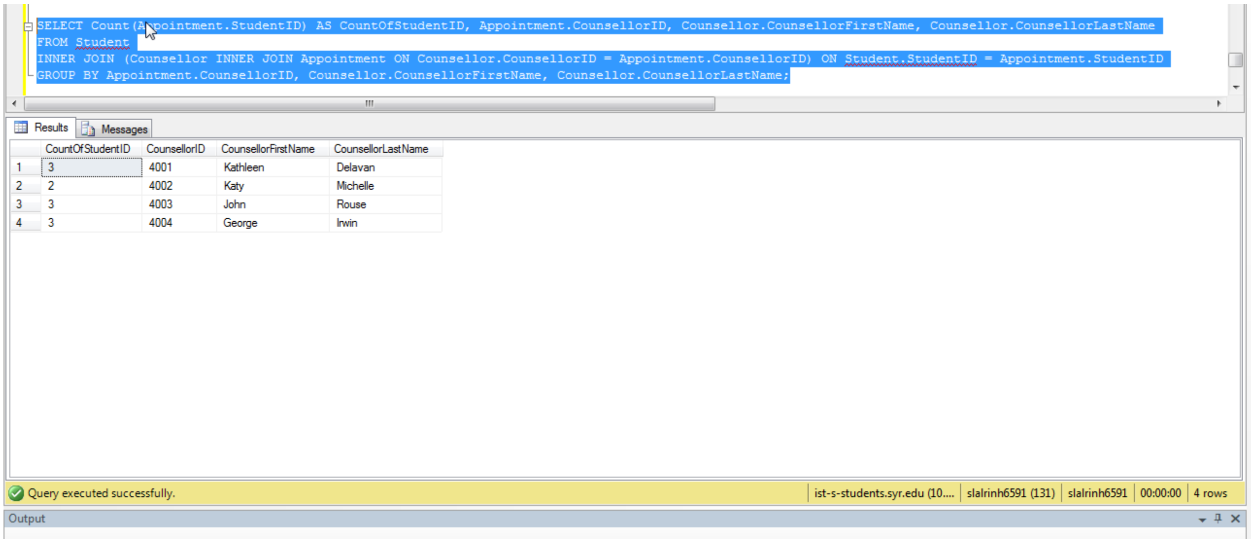
```
insert into Examination values('600','IST601','500','','','ODS 335');
insert into Examination values('601','IST600','505','','','ODS 315');
insert into Examination values('602','IST602','505','','','ODS 355 D');
insert into Examination values('603','IST614','502','','','ODS 35 A');
insert into Examination values('604','IST618','502','','','ODS 35 E');
insert into Examination values('605','IST621','501','','','ODS 31');
insert into Examination values('606','IST654','500','','','ODS 32');
insert into Examination values('607','IST659','503','','','ODS 31');
insert into Examination values('608','IST700','503','','','ODS 32');
```

	ExamID	CourseID	ProfessorID	ExamStart	ExamEnd	ExamLocation
1	600	IST601	500	2014-04-06 09:00:00.000	2014-04-06 11:00:00.000	ODS 335
2	601	IST600	505	2014-04-06 10:00:00.000	2014-04-06 11:00:00.000	ODS 315
3	602	IST602	505	2014-04-06 13:00:00.000	2014-04-06 15:00:00.000	ODS 355 D
4	603	IST614	502	2014-04-06 15:00:00.000	2014-04-06 17:00:00.000	ODS 35 A
5	604	IST618	502	2014-04-06 16:00:00.000	2014-04-06 17:00:00.000	ODS 35 E
6	605	IST621	501	2014-04-07 09:00:00.000	2014-04-07 11:00:00.000	ODS 31
7	606	IST654	500	2014-04-07 10:00:00.000	2014-04-07 13:00:00.000	ODS 32
8	607	IST659	503	2014-04-08 11:00:00.000	2014-04-08 13:00:00.000	ODS 31
9	608	IST700	503	2014-04-08 13:00:00.000	2014-04-08 15:00:00.000	ODS 32

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 9 rows

MAJOR DATA QUESTIONS

1. No. of Students under each Counsellor.

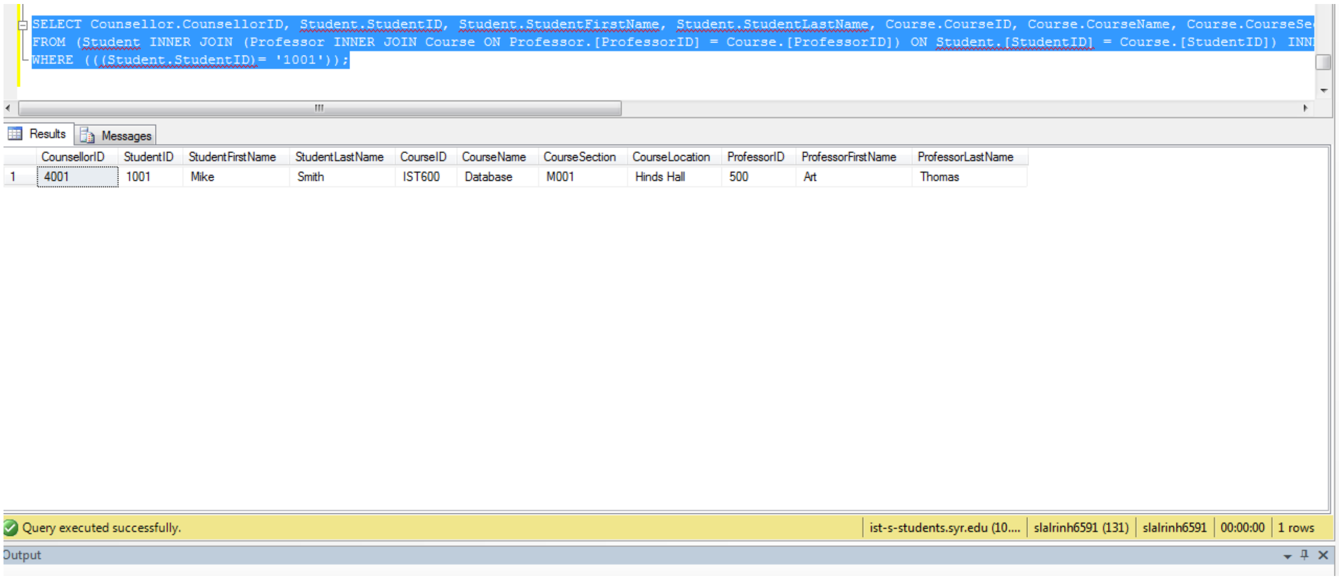


```
SELECT Count(Appointment.StudentID) AS CountOfStudentID, Appointment.CounsellorID, Counsellor.CounsellorFirstName, Counsellor.CounsellorLastName
FROM Student
INNER JOIN (Counsellor INNER JOIN Appointment ON Counsellor.CounsellorID = Appointment.CounsellorID) ON Student.StudentID = Appointment.StudentID
GROUP BY Appointment.CounsellorID, Counsellor.CounsellorFirstName, Counsellor.CounsellorLastName;
```

	CountOfStudentID	CounsellorID	CounsellorFirstName	CounsellorLastName
1	3	4001	Kathleen	Delavan
2	2	4002	Katy	Michelle
3	3	4003	John	Rouse
4	3	4004	George	Irwin

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 4 rows

2. Details of the StudentID = ‘1001’ , including the Course taken and the Professor who teaches the course

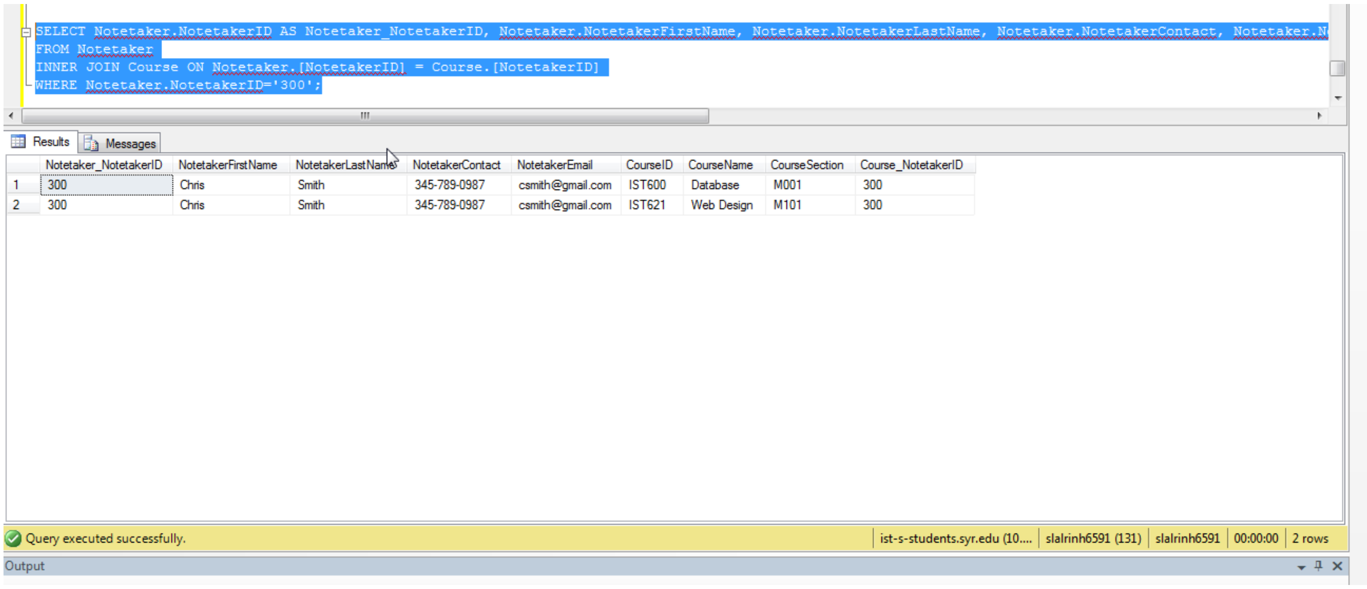


```
SELECT Counsellor.CounsellorID, Student.StudentID, Student.StudentFirstName, Student.StudentLastName, Course.CourseID, Course.CourseName, Course.CourseSe
FROM (Student INNER JOIN (Professor INNER JOIN Course ON Professor.[ProfessorID] = Course.[ProfessorID]) ON Student.[StudentID] = Course.[StudentID]) INN
WHERE (((Student.StudentID)= '1001'));
```

	CounsellorID	StudentID	StudentFirstName	StudentLastName	CourseID	CourseName	CourseSection	CourseLocation	ProfessorID	ProfessorFirstName	ProfessorLastName
1	4001	1001	Mike	Smith	IST600	Database	M001	Hinds Hall	500	Art	Thomas

Query executed successfully. | ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 1 rows

3. How many classes do a particular Notetaker take notes for?

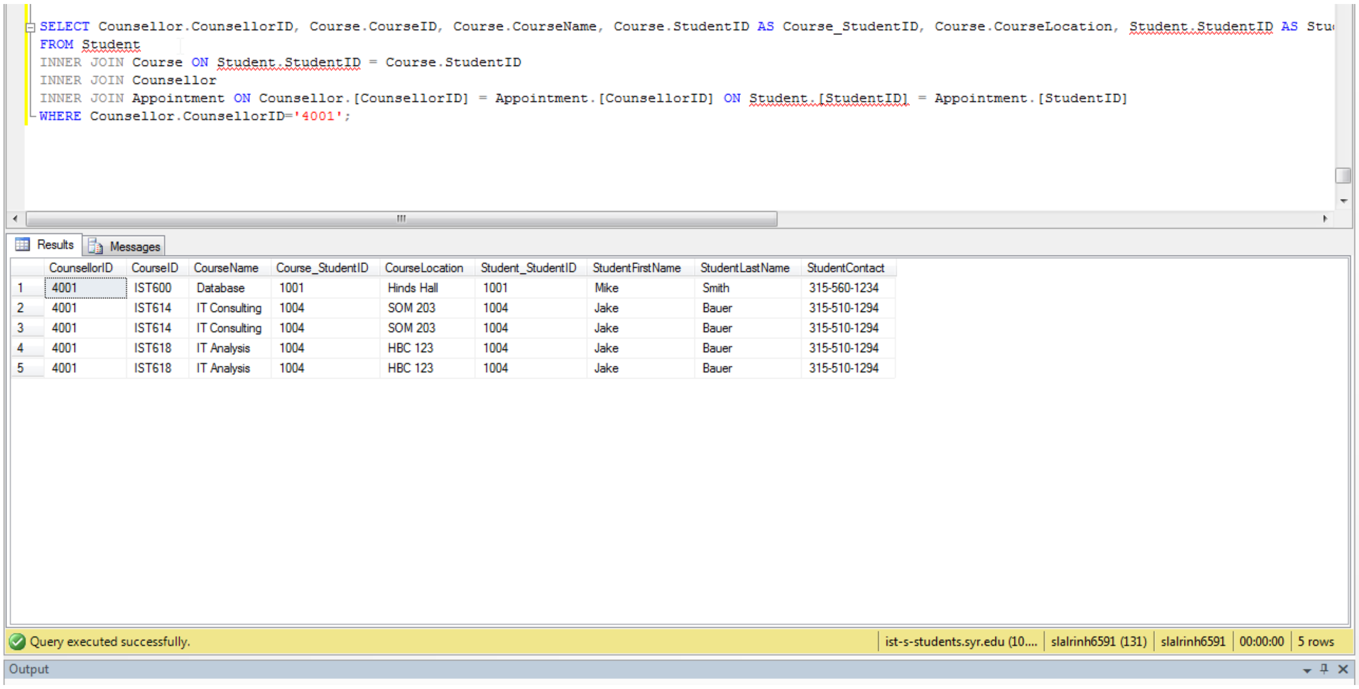


```
SELECT Notetaker.NotetakerID AS Notetaker_NotetakerID, Notetaker.NotetakerFirstName, Notetaker.NotetakerLastName, Notetaker.NotetakerContact, Notetaker.N
FROM Notetaker
INNER JOIN Course ON Notetaker.[NotetakerID] = Course.[NotetakerID]
WHERE Notetaker.NotetakerID='300';
```

	Notetaker_NotetakerID	NotetakerFirstName	NotetakerLastName	NotetakerContact	NotetakerEmail	CourseID	CourseName	CourseSection	Course_NotetakerID
1	300	Chris	Smith	345-789-0987	csmith@gmail.com	IST600	Database	M001	300
2	300	Chris	Smith	345-789-0987	csmith@gmail.com	IST621	Web Design	M101	300

```
Query executed successfully.    ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 2 rows
```

4. What are the various classes in which a counsellor's students are enrolled for?

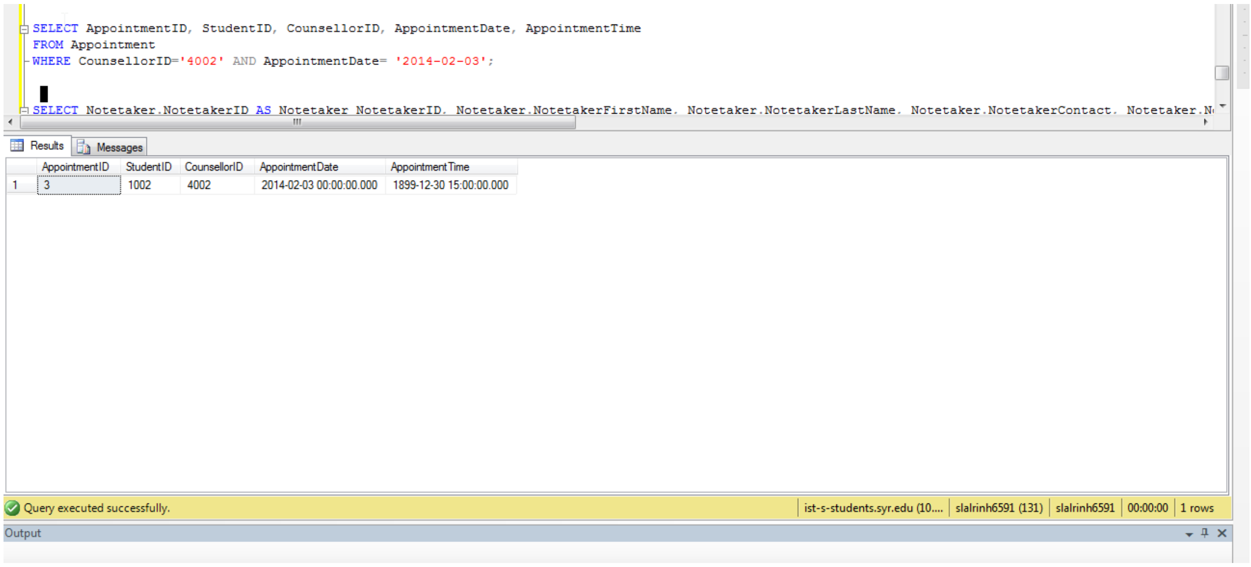


```
SELECT Counsellor.CounsellorID, Course.CourseID, Course.CourseName, Course.StudentID AS Course_StudentID, Course.CourseLocation, Student.StudentID AS Stu
FROM Student
INNER JOIN Course ON Student.StudentID = Course.StudentID
INNER JOIN Counsellor
INNER JOIN Appointment ON Counsellor.[CounsellorID] = Appointment.[CounsellorID] ON Student.[StudentID] = Appointment.[StudentID]
WHERE Counsellor.CounsellorID='4001';
```

	CounsellorID	CourseID	CourseName	Course_StudentID	CourseLocation	Student_StudentID	StudentFirstName	StudentLastName	StudentContact
1	4001	IST600	Database	1001	Hinds Hall	1001	Mike	Smith	315-560-1234
2	4001	IST614	IT Consulting	1004	SOM 203	1004	Jake	Bauer	315-510-1294
3	4001	IST614	IT Consulting	1004	SOM 203	1004	Jake	Bauer	315-510-1294
4	4001	IST618	IT Analysis	1004	HBC 123	1004	Jake	Bauer	315-510-1294
5	4001	IST618	IT Analysis	1004	HBC 123	1004	Jake	Bauer	315-510-1294

```
Query executed successfully.    ist-s-students.syr.edu (10.... | slalrinh6591 (131) | slalrinh6591 | 00:00:00 | 5 rows
```

5. How many appointments does a counsellor have for a particular day?



```
SELECT AppointmentID, StudentID, CounsellorID, AppointmentDate, AppointmentTime
FROM Appointment
WHERE CounsellorID='4002' AND AppointmentDate= '2014-02-03';


SELECT Notetaker.NotetakerID AS Notetaker NotetakerID, Notetaker.NotetakerFirstName, Notetaker.NotetakerLastName, Notetaker.NotetakerContact, Notetaker.N
```

	AppointmentID	StudentID	CounsellorID	AppointmentDate	AppointmentTime
1	3	1002	4002	2014-02-03 00:00:00.000	1899-12-30 15:00:00.000

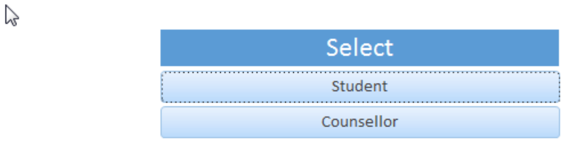
```
Query executed successfully.    ist-s-students.syr.edu (10....  slalrinh6591 (131)  slalrinh6591  00:00:00  1 rows
```

INTERFACES

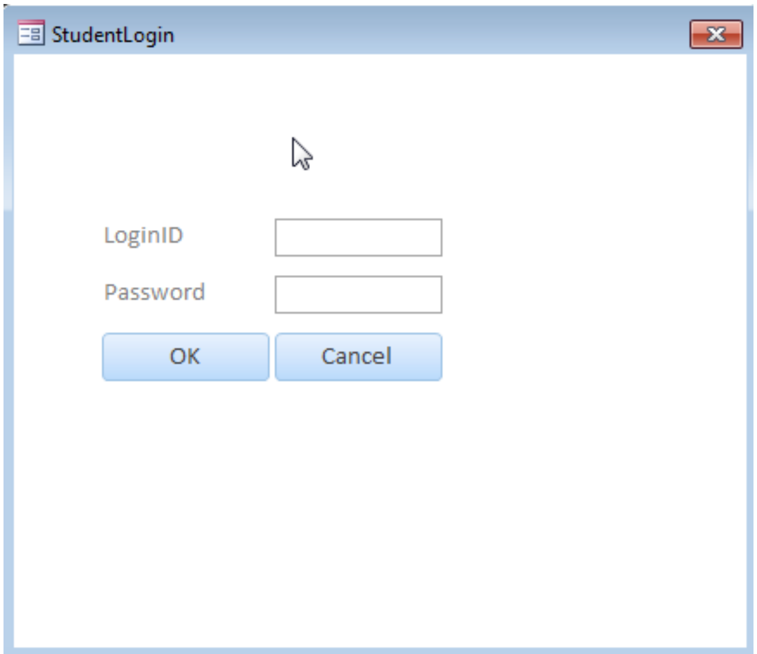
There are two types of log in

1. Student

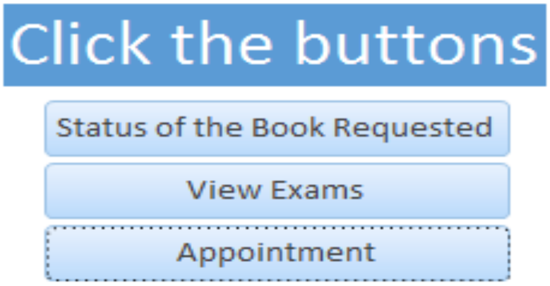
2. Counsellor



1. Student Login

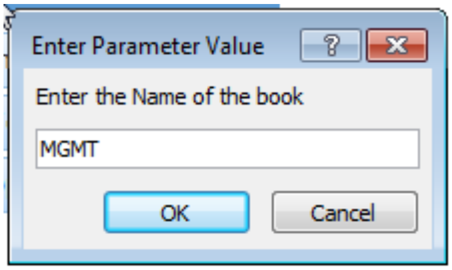


After Successful Login, a student will be directed to a page, where there are options to go to three pages.



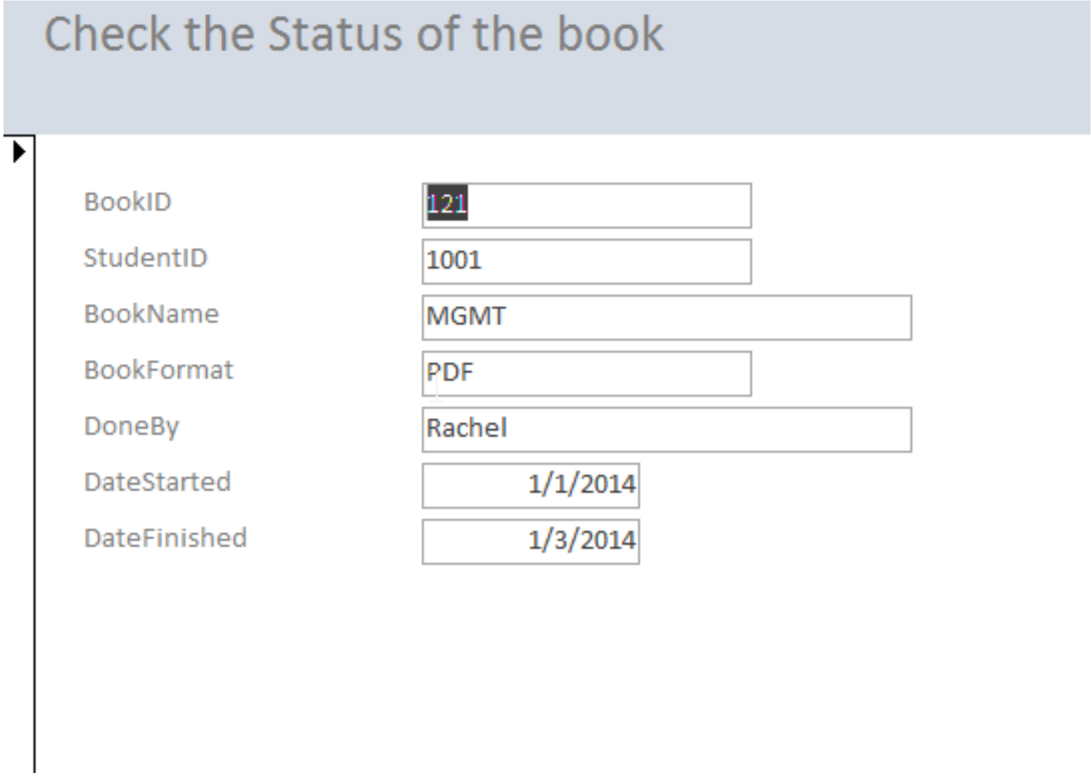
- Status of the Book Requested.

This will prompt the student to enter the name of the book he requested.



Enter the Name of the book

MGMT

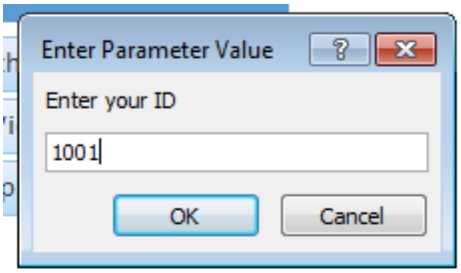


Check the Status of the book

BookID	121
StudentID	1001
BookName	MGMT
BookFormat	PDF
DoneBy	Rachel
DateStarted	1/1/2014
DateFinished	1/3/2014

If the DateFinished has a value, then it means the book is already processed.

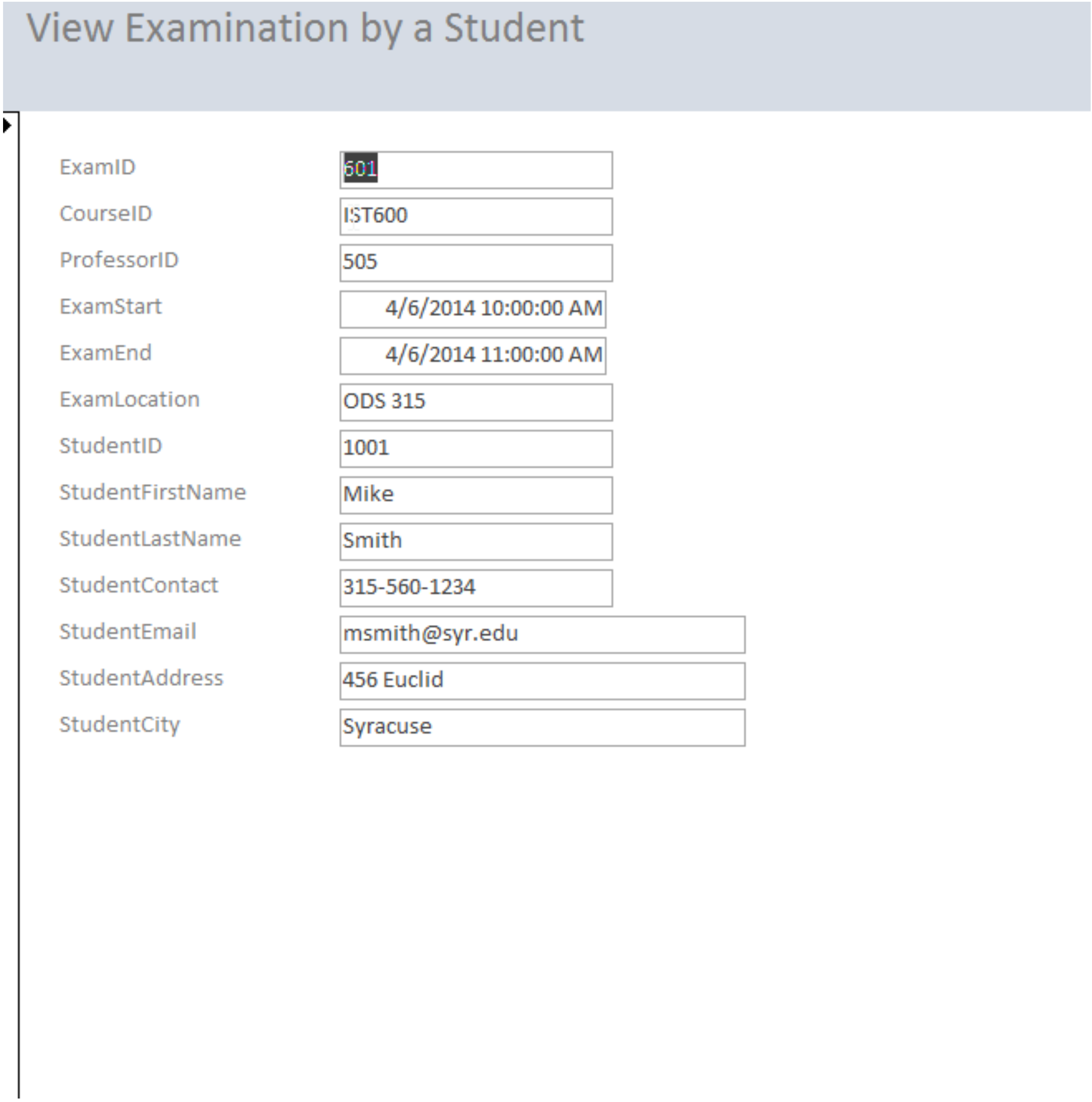
- View Exams
  - This will prompt the student to enter his StudentID



Enter your ID

1001

- After entering his/her ID, the student can view the scheduled exams.

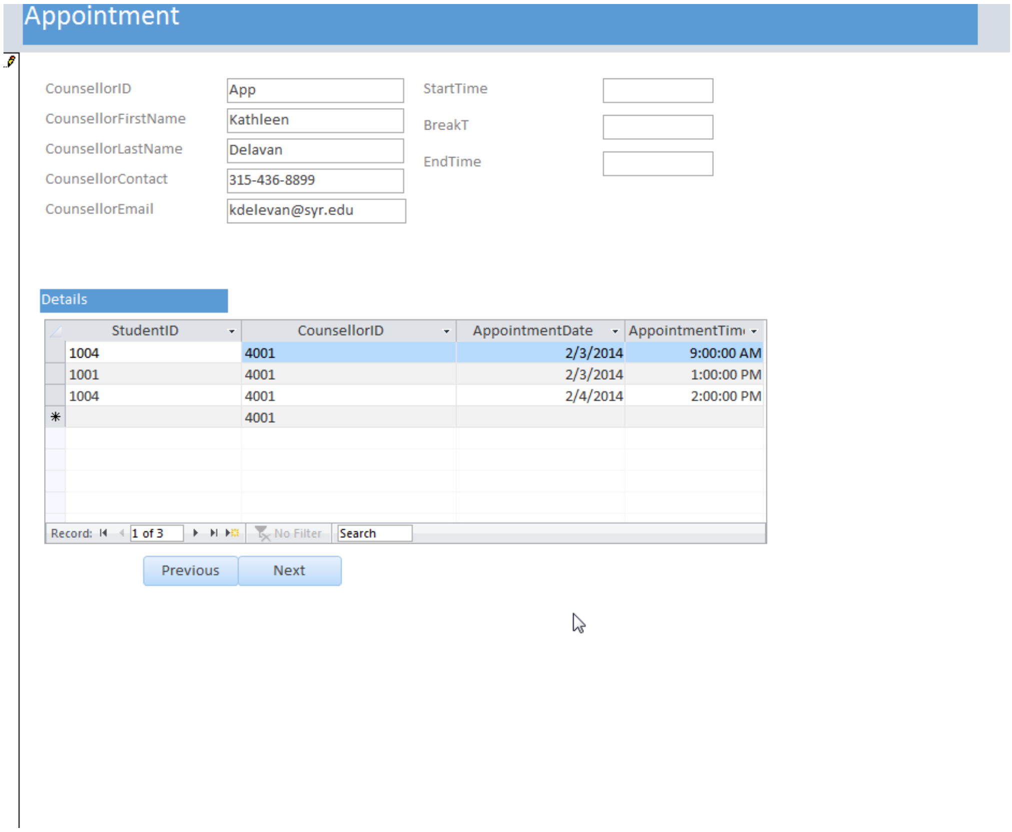


View Examination by a Student

ExamID	601
CourseID	IST600
ProfessorID	505
ExamStart	4/6/2014 10:00:00 AM
ExamEnd	4/6/2014 11:00:00 AM
ExamLocation	ODS 315
StudentID	1001
StudentFirstName	Mike
StudentLastName	Smith
StudentContact	315-560-1234
StudentEmail	msmith@syr.edu
StudentAddress	456 Euclid
StudentCity	Syracuse

- Appointment

  This page will allow the student to view all the appointment details and sign up for an appointment with a counsellor



Appointment

CounsellorID	App
CounsellorFirstName	Kathleen
CounsellorLastName	Delavan
CounsellorContact	315-436-8899
CounsellorEmail	kdelevan@syr.edu
StartTime	
BreakT	
EndTime	

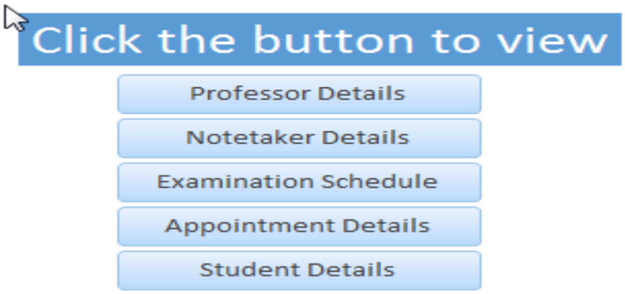
Details

StudentID	CounsellorID	AppointmentDate	AppointmentTim
1004	4001	2/3/2014	9:00:00 AM
1001	4001	2/3/2014	1:00:00 PM
1004	4001	2/4/2014	2:00:00 PM
*	4001		

Record: 1 of 3

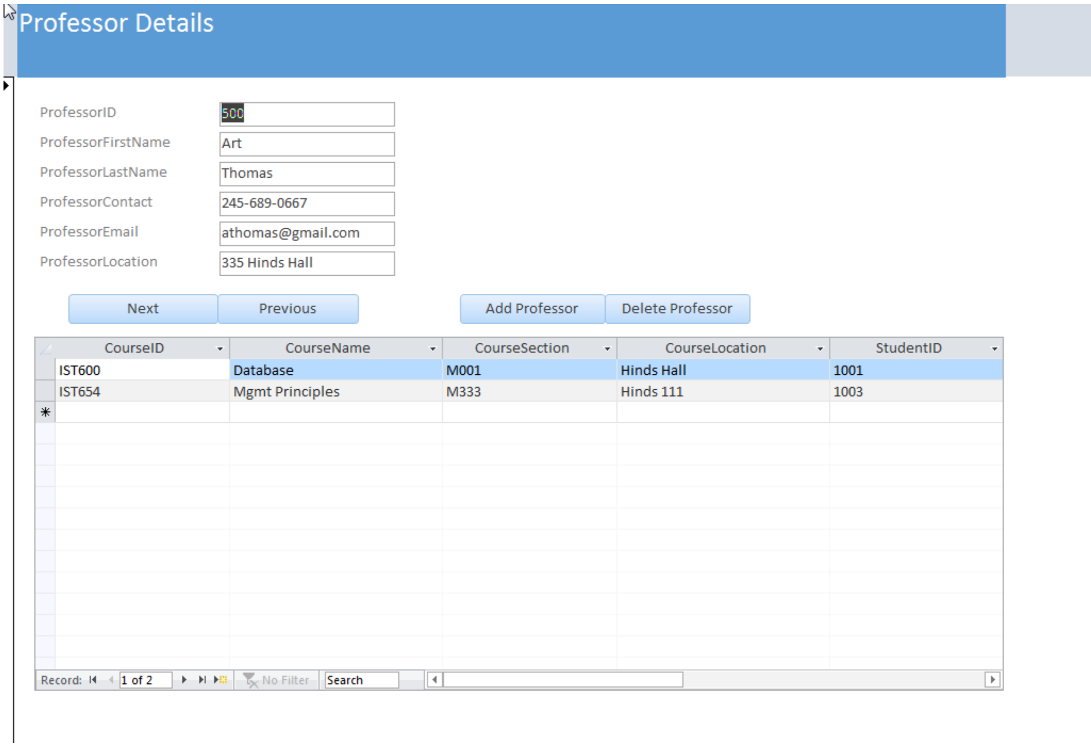
2. Counsellor Login

After successful Login, this navigation page will show up.



Click the button to view

- Professor Details
    - This will list out all the professor under whom the students have taken classes

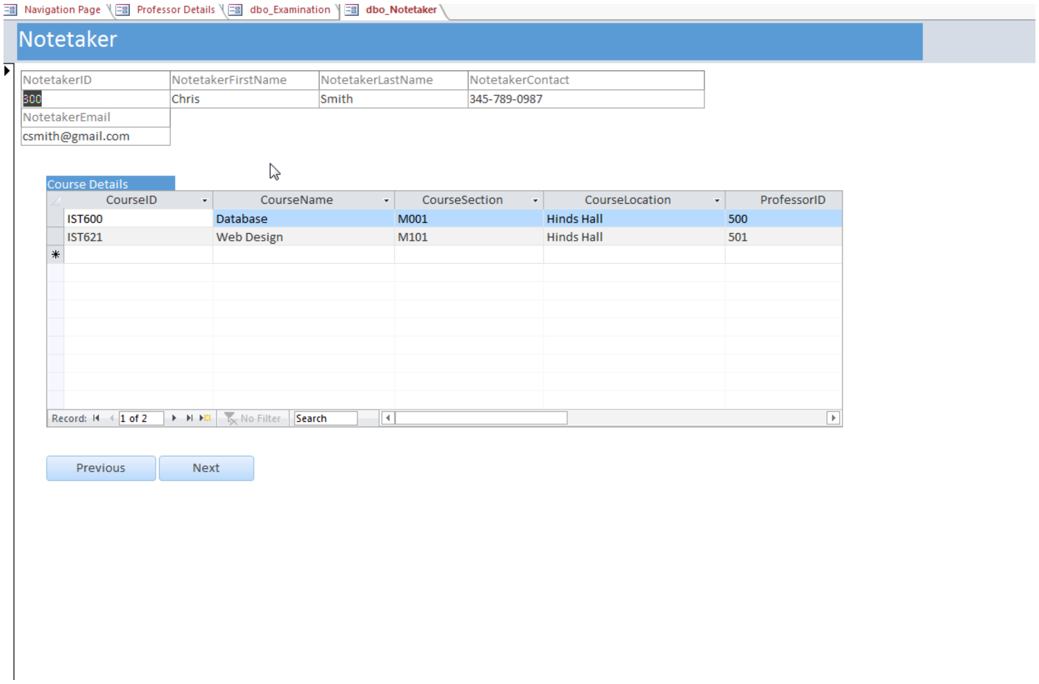


Professor Details

ProfessorID	500
ProfessorFirstName	Art
ProfessorLastName	Thomas
ProfessorContact	245-689-0667
ProfessorEmail	athomas@gmail.com
ProfessorLocation	335 Hinds Hall

CourseID	CourseName	CourseSection	CourseLocation	StudentID
IST600	Database	M001	Hinds Hall	1001
IST654	Mgmt Principles	M333	Hinds 111	1003

- Notetaker Details
    - This will show all the notetaker that are currently working for Disability Services



Notetaker

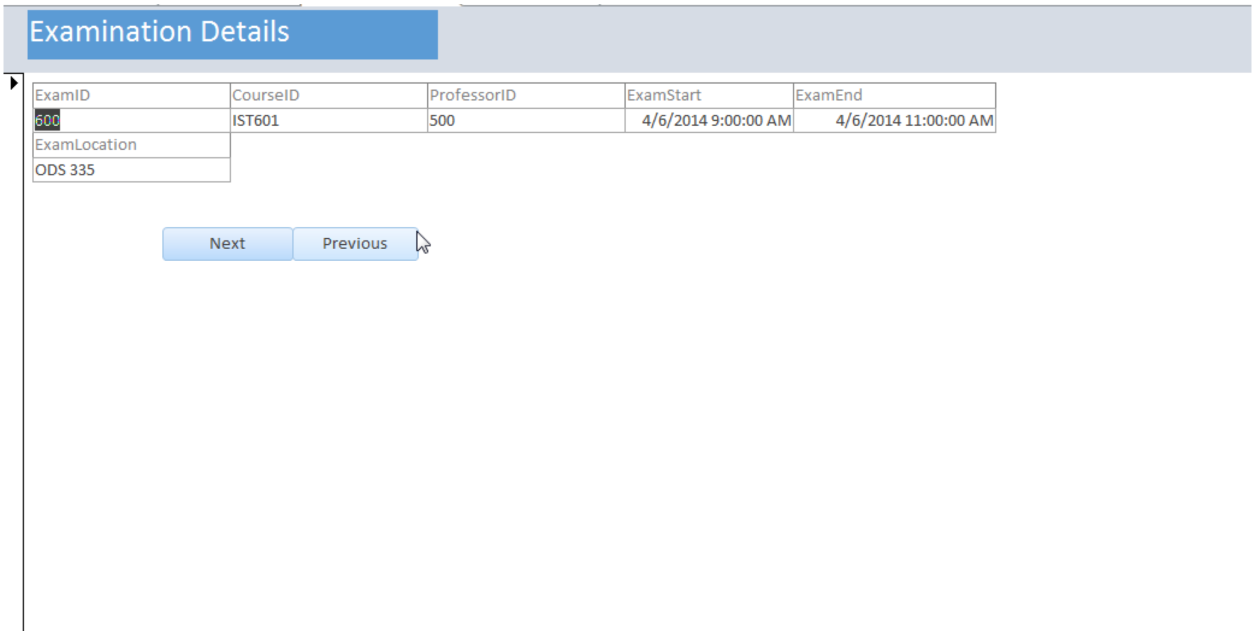
NotetakerID	NotetakerFirstName	NotetakerLastName	NotetakerContact
300	Chris	Smith	345-789-0987

NotetakerEmail
csmith@gmail.com

Course Details

CourseID	CourseName	CourseSection	CourseLocation	ProfessorID
IST600	Database	M001	Hinds Hall	500
IST621	Web Design	M101	Hinds Hall	501

- Examination Schedule
    - This will list out all the Examination Schedule

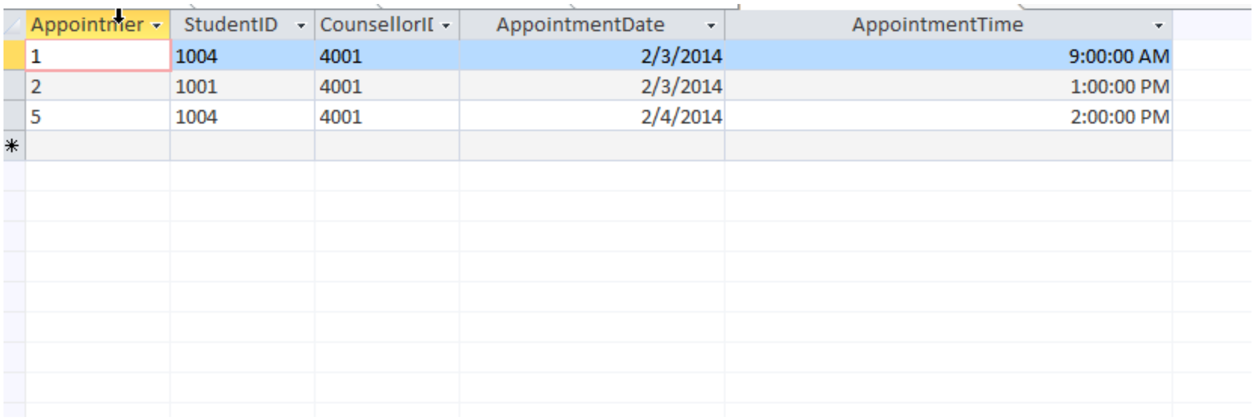


Examination Details

ExamID	CourseID	ProfessorID	ExamStart	ExamEnd
600	IST601	500	4/6/2014 9:00:00 AM	4/6/2014 11:00:00 AM

ExamLocation
ODS 335

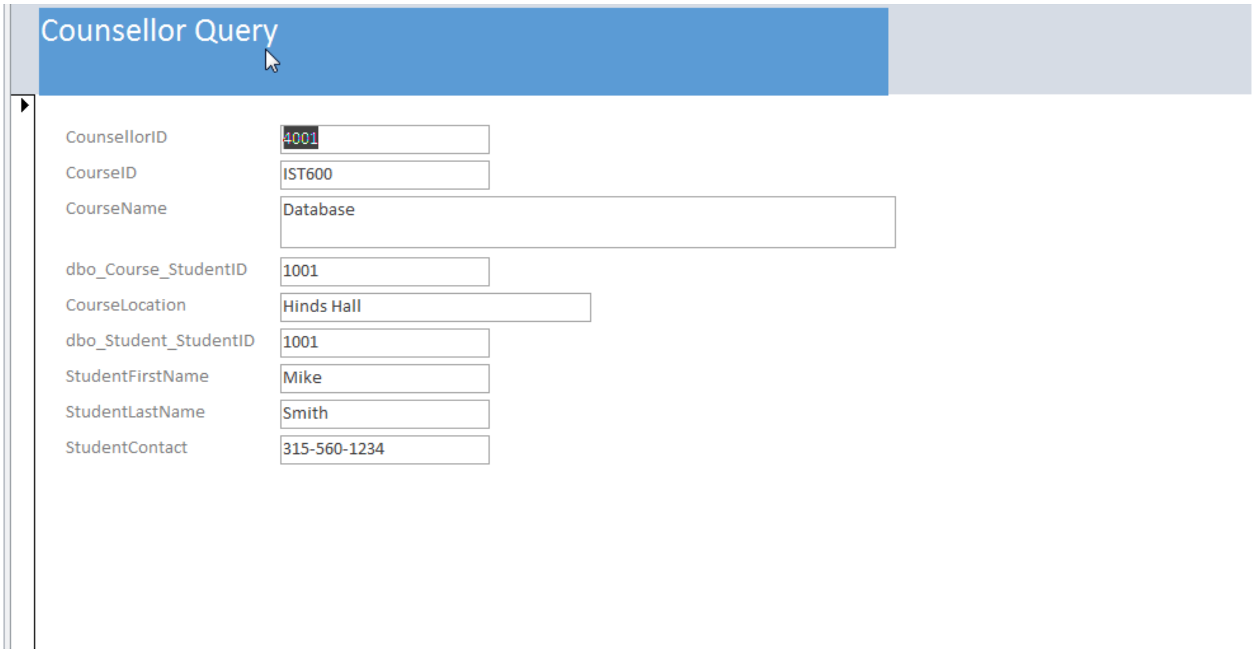
- Appointment Details
    - This will prompt the counsellor to enter the ID and show the scheduled appointment that he/she has



Appointmer	StudentID	CounsellorII	AppointmentDate	AppointmentTime
1	1004	4001	2/3/2014	9:00:00 AM
2	1001	4001	2/3/2014	1:00:00 PM
5	1004	4001	2/4/2014	2:00:00 PM

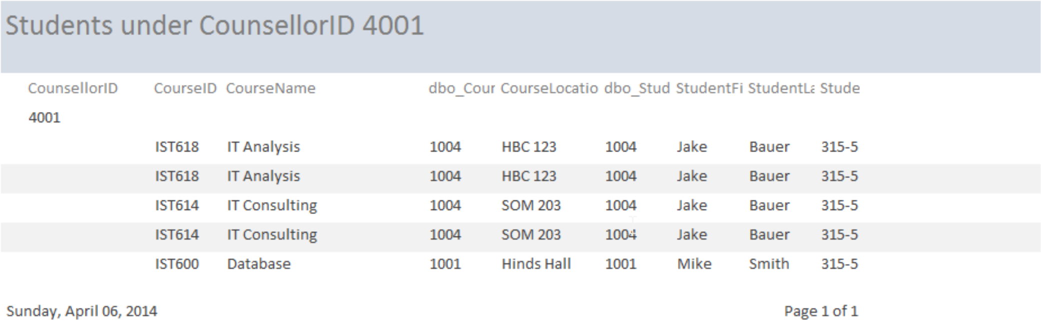
- Student Details
    - This will list out all the students under a counsellor’s guidance

REPORT



Counsellor Query

CounsellorID	4001
CourseID	IST600
CourseName	Database
dbo_Course_StudentID	1001
CourseLocation	Hinds Hall
dbo_Student_StudentID	1001
StudentFirstName	Mike
StudentLastName	Smith
StudentContact	315-560-1234

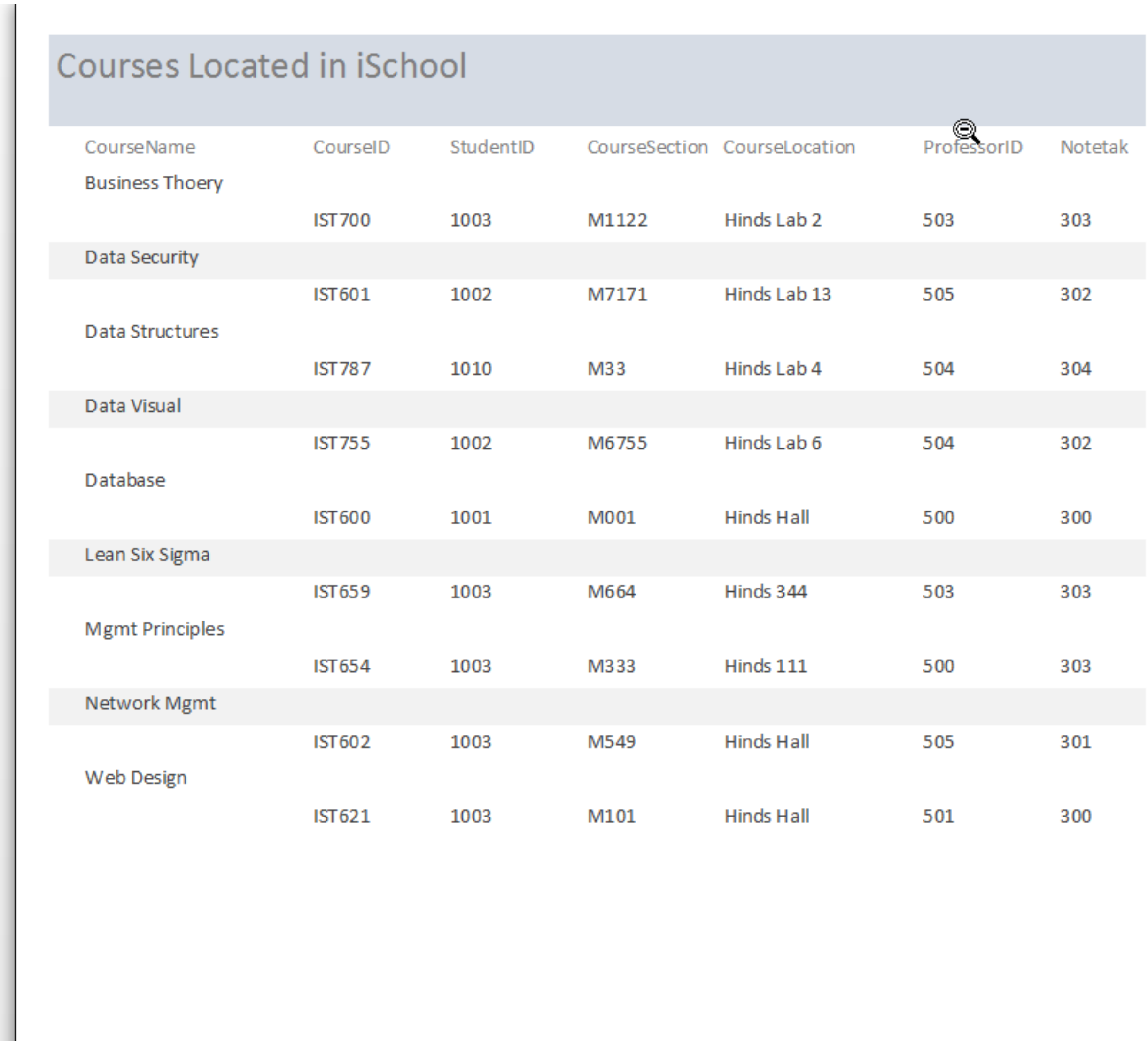


Students under CounsellorID 4001

CounsellorID	CourseID	CourseName	dbo_Cour	CourseLocatio	dbo_Stud	StudentFi	StudentLa	Stude
4001								
	IST618	IT Analysis	1004	HBC 123	1004	Jake	Bauer	315-5
	IST618	IT Analysis	1004	HBC 123	1004	Jake	Bauer	315-5
	IST614	IT Consulting	1004	SOM 203	1004	Jake	Bauer	315-5
	IST614	IT Consulting	1004	SOM 203	1004	Jake	Bauer	315-5
	IST600	Database	1001	Hinds Hall	1001	Mike	Smith	315-5

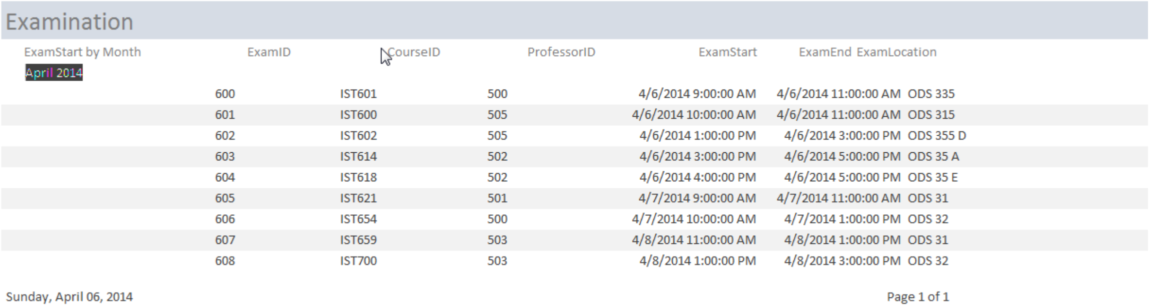
Sunday, April 06, 2014

Page 1 of 1



Courses Located in iSchool

CourseName	CourseID	StudentID	CourseSection	CourseLocation	ProfessorID	Notetak
Business Thoery						
	IST700	1003	M1122	Hinds Lab 2	503	303
Data Security						
	IST601	1002	M7171	Hinds Lab 13	505	302
Data Structures						
	IST787	1010	M33	Hinds Lab 4	504	304
Data Visual						
	IST755	1002	M6755	Hinds Lab 6	504	302
Database						
	IST600	1001	M001	Hinds Hall	500	300
Lean Six Sigma						
	IST659	1003	M664	Hinds 344	503	303
Mgmt Principles						
	IST654	1003	M333	Hinds 111	500	303
Network Mgmt						
	IST602	1003	M549	Hinds Hall	505	301
Web Design						
	IST621	1003	M101	Hinds Hall	501	300



Examination

ExamStart by Month	ExamID	CourseID	ProfessorID	ExamStart	ExamEnd	ExamLocation
April 2014						
	600	IST601	500	4/6/2014 9:00:00 AM	4/6/2014 11:00:00 AM	ODS 335
	601	IST600	505	4/6/2014 10:00:00 AM	4/6/2014 11:00:00 AM	ODS 315
	602	IST602	505	4/6/2014 1:00:00 PM	4/6/2014 3:00:00 PM	ODS 355 D
	603	IST614	502	4/6/2014 3:00:00 PM	4/6/2014 5:00:00 PM	ODS 35 A
	604	IST618	502	4/6/2014 4:00:00 PM	4/6/2014 5:00:00 PM	ODS 35 E
	605	IST621	501	4/7/2014 9:00:00 AM	4/7/2014 11:00:00 AM	ODS 31
	606	IST654	500	4/7/2014 10:00:00 AM	4/7/2014 1:00:00 PM	ODS 32
	607	IST659	503	4/8/2014 11:00:00 AM	4/8/2014 1:00:00 PM	ODS 31
	608	IST700	503	4/8/2014 1:00:00 PM	4/8/2014 3:00:00 PM	ODS 32

Sunday, April 06, 2014

Page 1 of 1