

College Database Project Report

SQL Code For Creation of Database:

```
CREATE TABLE Course (  
    cid INT NOT NULL,  
    cname CHAR (50) NULL,  
    sid INT NULL,  
    iid INT NULL,  
    PRIMARY KEY (cid)  
);
```

```
CREATE TABLE Instructor(  
    iid INT NOT NULL,  
    iname CHAR (50) NULL,  
    age INT NULL,  
    PRIMARY KEY (iid)  
);
```

```
CREATE TABLE Student  
(  
    sid integer,  
    sname CHAR(50) NULL,  
    grade INT NULL,  
    PRIMARY KEY(sid)  
  
)
```

```
CREATE TABLE Department  
(  
    did integer,  
    dname char(50),  
    iid integer,  
    cid integer  
    PRIMARY KEY(did)  
  
)
```

```
CREATE TABLE Offer
```

```
(
    cid integer REFERENCES Course,
    did integer REFERENCES Department,
    Primary key(cid, did)
)
```

CREATE TABLE Enroll

```
(
    cid integer REFERENCES Course,
    sid integer REFERENCES Student,
    Primary key(cid, sid)
)
```

CREATE TABLE Teaches

```
(
    cid integer REFERENCES Course,
    iid integer REFERENCES Instructor,
    Primary key(cid, iid)
)
```

CREATE TABLE Works_In

```
(
    did integer REFERENCES Department,
    iid integer REFERENCES Instructor,
    Primary key(did, iid)
)
```

CREATE TABLE Heads

```
(
    did integer REFERENCES Department,
    iid integer REFERENCES Instructor,
    Primary key(did, iid)
)
```

```
insert into Student (sid, sname, grade) values (210, 'Ashley', 14);
insert into Student (sid, sname, grade) values (148, 'Anthony', 13);
insert into Student (sid, sname, grade) values (915, 'Clarence', 15);
insert into Student (sid, sname, grade) values (428, 'Jennifer', 15);
insert into Student (sid, sname, grade) values (388, 'Jerry', 14);
insert into Student (sid, sname, grade) values (340, 'Bill', 14);
```

```
insert into Instructor (iid, iname, age) values (1000, 'Jason', 35);
insert into Instructor (iid, iname, age) values (2000, 'Matt', 90);
```

insert into Instructor (iid, iname, age) values (3000, 'Bob', 47);
insert into Instructor (iid, iname, age) values (4000, 'Ethan', 62);
insert into Instructor (iid, iname, age) values (5000, 'Liam', 29);
insert into Instructor (iid, iname, age) values (6000, 'Joe', 42);
insert into Instructor (iid, iname, age) values (7000, 'Alfred', 50);

insert into Course (cid, cname, sid, iid) values (22, 'Math 123', 148, 1000);
insert into Course (cid, cname, sid, iid) values (25, 'Physics 133', 210, 2000);
insert into Course (cid, cname, sid, iid) values (35, 'Biology 101', 388, 3000);
insert into Course (cid, cname, sid, iid) values (46, 'History 111', 428, 4000);
insert into Course (cid, cname, sid, iid) values (56, 'Chemistry 405', 915, 5000);
insert into Course (cid, cname, sid, iid) values (78, 'Database 109', 148, 6000);
insert into Course (cid, cname, sid, iid) values (92, 'English 110', 915, 7000);
insert into Course (cid, cname, sid, iid) values (12, 'Sociology 120', 915, 7000);
insert into Course (cid, cname, sid, iid) values (32, 'Economics 456', 428, 4000);
insert into Course (cid, cname, sid, iid) values (41, 'Astronomy 208', 340, 1000);

insert into Department (did, dname, iid, cid) values (1, 'Computer Science', 6000, 78);
insert into Department (did, dname, iid, cid) values (2, 'Chemistry', 5000, 56);
insert into Department (did, dname, iid, cid) values (3, 'English', 7000, 92);
insert into Department (did, dname, iid, cid) values (4, 'Economics', 4000, 32);
insert into Department (did, dname, iid, cid) values (5, 'History', 4000, 46);
insert into Department (did, dname, iid, cid) values (6, 'Math', 1000, 22);
insert into Department (did, dname, iid, cid) values (7, 'Physics', 2000, 25);
insert into Department (did, dname, iid, cid) values (8, 'Sociology', 7000, 12);
insert into Department (did, dname, iid, cid) values (9, 'Astronomy', 1000, 41);
insert into Department (did, dname, iid, cid) values (10, 'Biology', 3000, 35);

insert into Enroll (cid, sid) values (22, 148);
insert into Enroll (cid, sid) values (25, 210);
insert into Enroll (cid, sid) values (35, 388);
insert into Enroll (cid, sid) values (46, 428);
insert into Enroll (cid, sid) values (56, 915);
insert into Enroll (cid, sid) values (78, 148);
insert into Enroll (cid, sid) values (92, 915);
insert into Enroll (cid, sid) values (12, 915);
insert into Enroll (cid, sid) values (32, 428);
insert into Enroll (cid, sid) values (41, 340);

insert into Heads (did, iid) values (1, 6000);
insert into Heads (did, iid) values (2, 5000);
insert into Heads (did, iid) values (3, 7000);

```
insert into Heads (did, iid) values (4, 4000);
insert into Heads (did, iid) values (5, 4000);
insert into Heads (did, iid) values (6, 1000);
insert into Heads (did, iid) values (7, 2000);
insert into Heads (did, iid) values (8, 7000);
insert into Heads (did, iid) values (9, 1000);
insert into Heads (did, iid) values (10, 3000);
```

```
insert into Offer (cid, did) values (78, 1);
insert into Offer (cid, did) values (56, 2);
insert into Offer (cid, did) values (92, 3);
insert into Offer (cid, did) values (32, 4);
insert into Offer (cid, did) values (46, 5);
insert into Offer (cid, did) values (22, 6);
insert into Offer (cid, did) values (25, 7);
insert into Offer (cid, did) values (12, 8);
insert into Offer (cid, did) values (41, 9);
insert into Offer (cid, did) values (35, 10);
```

```
insert into Teaches(cid, iid) values (78, 6000);
insert into Teaches(cid, iid) values (56, 5000);
insert into Teaches(cid, iid) values (92, 7000);
insert into Teaches(cid, iid) values (32, 4000);
insert into Teaches(cid, iid) values (46, 4000);
insert into Teaches(cid, iid) values (22, 1000);
insert into Teaches(cid, iid) values (25, 2000);
insert into Teaches(cid, iid) values (12, 7000);
insert into Teaches(cid, iid) values (41, 1000);
insert into Teaches(cid, iid) values (35, 3000);
```

```
insert into Works_In (did, iid) values (1, 6000);
insert into Works_In(did, iid) values (2, 5000);
insert into Works_In(did, iid) values (3, 7000);
insert into Works_In(did, iid) values (4, 4000);
insert into Works_In(did, iid) values (5, 4000);
insert into Works_In(did, iid) values (6, 1000);
insert into Works_In(did, iid) values (7, 2000);
insert into Works_In(did, iid) values (8, 7000);
insert into Works_In(did, iid) values (9, 1000);
insert into Works_In(did, iid) values (10, 3000);
```

Queries:

1. Find the names of all students taking Biology 101.

```
SELECT S.sname  
FROM Student S, Course C  
WHERE S.sid = C.sid AND C.cname = 'Biology 101'
```

2. Find the names of all instructors that are less than 40 years old.

```
SELECT I.iname  
FROM Instructor I  
WHERE I.age < 40
```

3. Find the names of the instructors that are in the Computer Science Department.

```
SELECT I.iname  
FROM Instructor I, Department D  
WHERE I.iid = D.iid AND D.dname = 'Computer Science'
```

4. Find the names of the courses that the Chemistry Department offers that have Course ID's larger than 30.

```
SELECT C.cname  
FROM Department D, Course C  
WHERE D.did = 2 AND D.cid > 30 AND C.cid = D.cid
```

5. Find the names of students who are in grade 13 or 14 and enrolled in Math 123 or Physics 133.

```
SELECT DISTINCT S.sname  
FROM Department D, Course C, Student S  
WHERE (S.grade = 13 OR S.grade = 14) AND (C.sid = S.sid AND (C.cname = 'Math 123' OR  
C.cname = 'Physics 133'))
```

6. Find the names of the instructors that are in the English Department that are older than 30.

```
SELECT DISTINCT I.iname  
FROM Instructor I, Department D  
WHERE I.age > 30 AND (D.iid = I.iid)
```

7. Find the names of instructors who have taught at least two courses.

```

SELECT DISTINCT I.iname
FROM Instructor I, Course C
WHERE C.iid = I.iid
GROUP BY I.iname
HAVING COUNT(*)>1

```

8. Find the IDs of all students taking Astronomy 208 and are named "Bill".

```

SELECT DISTINCT S.sid
FROM Student S, Course C
WHERE S.sid = C.sid AND C.cname = 'Astronomy 208' AND S.sname = 'Bill'

```

9. Find the names of the Instructors in the Economics Department that have an ID higher than 2000.

```

SELECT DISTINCT I.iname
FROM Instructor I, Department D
WHERE I.iid > 2000 AND D.did = 4 AND D.iid = I.iid

```

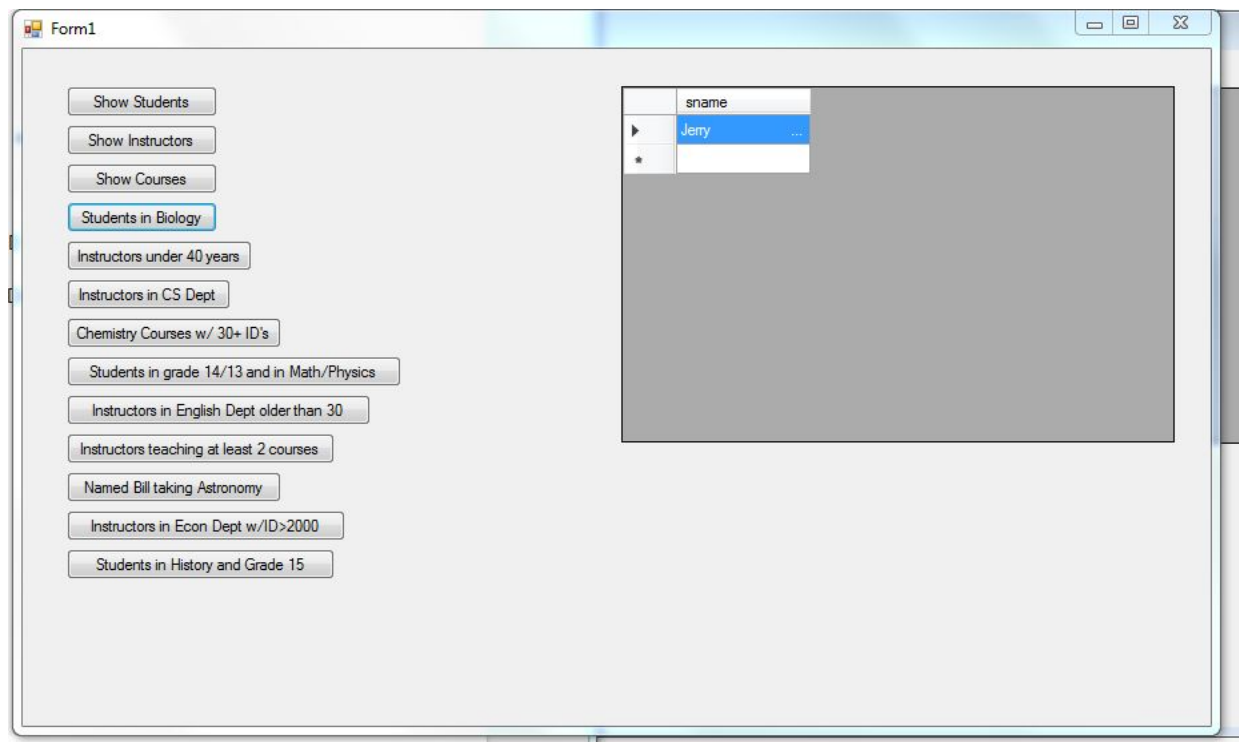
10. Find the names of all students enrolled in History 111 and in grade 15.

```

SELECT DISTINCT S.sname
FROM Student S, Course C
WHERE S.grade = 15 AND C.cname = 'History 111' AND S.sid = C.sid

```

Screenshots for Results of Queries via GUI:



Form1

Show Students

Show Instructors

Show Courses

Students in Biology

Instructors under 40 years

Instructors in CS Dept

Chemistry Courses w/ 30+ ID's

Students in grade 14/13 and in Math/Physics

Instructors in English Dept older than 30

Instructors teaching at least 2 courses

Named Bill taking Astronomy

Instructors in Econ Dept w/ID>2000

Students in History and Grade 15

	iname	
▶	Jason	...
	Liam	...
*		

Form1

Show Students

Show Instructors

Show Courses

Students in Biology

Instructors under 40 years

Instructors in CS Dept

Chemistry Courses w/ 30+ ID's

Students in grade 14/13 and in Math/Physics

Instructors in English Dept older than 30

Instructors teaching at least 2 courses

Named Bill taking Astronomy

Instructors in Econ Dept w/ID>2000

Students in History and Grade 15

	iname	
▶	Joe	...
*		

Form1

Show Students

Show Instructors

Show Courses

Students in Biology

Instructors under 40 years

Instructors in CS Dept

Chemistry Courses w/ 30+ ID's

Students in grade 14/13 and in Math/Physics

Instructors in English Dept older than 30

Instructors teaching at least 2 courses

Named Bill taking Astronomy

Instructors in Econ Dept w/ID>2000

Students in History and Grade 15

	cname
▶	Chemistry 405 ...
*	

Form1

Show Students

Show Instructors

Show Courses

Students in Biology

Instructors under 40 years

Instructors in CS Dept

Chemistry Courses w/ 30+ ID's

Students in grade 14/13 and in Math/Physics

Instructors in English Dept older than 30

Instructors teaching at least 2 courses

Named Bill taking Astronomy

Instructors in Econ Dept w/ID>2000

Students in History and Grade 15

	sname
▶	Anthony ...
	Ashley ...
*	

Form1

Show Students

Show Instructors

Show Courses

Students in Biology

Instructors under 40 years

Instructors in CS Dept

Chemistry Courses w/ 30+ ID's

Students in grade 14/13 and in Math/Physics

Instructors in English Dept older than 30

Instructors teaching at least 2 courses

Named Bill taking Astronomy

Instructors in Econ Dept w/ID>2000

Students in History and Grade 15

	iname
▶	Alfred ...
	Bob ...
	Ethan ...
	Jason ...
	Joe ...
	Matt ...
*	

Form1

Show Students

Show Instructors

Show Courses

Students in Biology

Instructors under 40 years

Instructors in CS Dept

Chemistry Courses w/ 30+ ID's

Students in grade 14/13 and in Math/Physics

Instructors in English Dept older than 30

Instructors teaching at least 2 courses

Named Bill taking Astronomy

Instructors in Econ Dept w/ID>2000

Students in History and Grade 15

	iname
▶	Alfred ...
	Ethan ...
	Jason ...
*	

Form1

Show Students

Show Instructors

Show Courses

Students in Biology

Instructors under 40 years

Instructors in CS Dept

Chemistry Courses w/ 30+ ID's

Students in grade 14/13 and in Math/Physics

Instructors in English Dept older than 30

Instructors teaching at least 2 courses

Named Bill taking Astronomy

Instructors in Econ Dept w/ID>2000

Students in History and Grade 15

	sid
▶	340
*	

Form1

Show Students

Show Instructors

Show Courses

Students in Biology

Instructors under 40 years

Instructors in CS Dept

Chemistry Courses w/ 30+ ID's

Students in grade 14/13 and in Math/Physics

Instructors in English Dept older than 30

Instructors teaching at least 2 courses

Named Bill taking Astronomy

Instructors in Econ Dept w/ID>2000

Students in History and Grade 15

	iname
▶	Ethan ...
*	

Form1

Buttons:

- Show Students
- Show Instructors
- Show Courses
- Students in Biology
- Instructors under 40 years
- Instructors in CS Dept
- Chemistry Courses w/ 30+ ID's
- Students in grade 14/13 and in Math/Physics
- Instructors in English Dept older than 30
- Instructors teaching at least 2 courses
- Named Bill taking Astronomy
- Instructors in Econ Dept w/ID>2000
- Students in History and Grade 15

	sname
▶	Jennifer
*	

Functional Dependencies:

Student:

sid->sname, grade

=>

sid->sname

sid->grade

Instructor:

iid->iname, age

=>

iid->iname

iid->age

Department:

did->dname, iid, cid

=>

did->dname

did->iid

did->cid

Course:

cid->cname, sid, iid

=>

cid->cname

cid->sid

cid->iid

Process Explanation:

I first began by creating the tables for the Students, Instructors, Courses, and Departments. I then created the tables for the relations as well such as Offer, Enroll, Heads, Works_In, and Teaches. I then put in sample data in order to test the results of my queries so that there would at least one piece of data that would signify that they functioned correctly. One query was changed from Project 1 because I tried to make it so that the ID numbers of instructors, departments, students, and courses were all different amounts of numbers such as student ID's being three digits while instructor ID's were 4 digits long. This made the data more distinguishable since the primary 4 tables all had their own ID system. I then found all the functional dependencies listed above for each table and realized that the schema was already normalized, therefore, it could not be decomposed any further. The GUI was then created for the queries and general table data. The GUI had three buttons that would display the entire data from some of the tables and then 10 buttons that would display data based on each inputted query. The names of the buttons give a brief description of what they will show. I then tested my code and screenshotted the results of the GUI and its buttons, which were correct. I learned a lot from this project about making a database to be as efficient as possible to extract data with the use of queries especially after inspecting the ER model

multiple times. This project also taught me a lot about GUI creation as I experimented with many other options in the Toolbox menu before deciding upon my final design.