COSC 3380: Design of Database Systems
Midterm Exam – [redacted]

91/100

Student's Name: [redacted]

Student ID: [redacted]

---

(1) What are the differences between base relations and virtual relations? Explain with an example each.

(5 points)

Base relations: where the actual data are store
example: Student James Potter with student id 77, lives at 123 Gothic Hollow, taking Divination and has a 4.0 GPA

Virtual relations: not actually store data but rather retrieve data and limit the amount of information (same as view)
example: a professor can get a report of James Potter's grades but should not be given his home address

(2) How does SQL facilitate implementation of the entity integrity constraint and referential integrity constraint in a relational database? Explain with an example.

-1

(5 points)

entity integrity: ensures uniqueness of entities: no 2 entities have the same information (may be similar), especially primary key

Primary Key

example: 2 students named Harry Potter must have different IDs to identify them

referential integrity: ensure the relationships between different entities, make sure an entity does not reference a non existing entity.

Foreign Key

example: grade report of [redacted] Hermione has to make sure to reference her id, meaning a grade has to belong to some student named Hermione and her data exists in database

(3) (10 points)

(i) Explain the difference between a key and a superkey.

A key is always a superkey, but not necessarily the other way around
A key should be a superkey with the least number of attributes (minimality)
A superkey is a combination of a number of attributes that together determine the uniqueness of a tuple
A set containing a key among other attributes is a superkey.
No two tuples have the same key

Super keys have redundant attributes.

(ii) Why do we designate one of the candidate keys of a relation to be the primary key?

Conventionally we use ID as the primary because it has only 1 values (instead of a combination of (date of birth, address, name) for example). We need to designate a candidate key to be a primary key because we need to use it to uniquely identify a data entry. Other candidate keys would only be used to ensure the uniqueness of data

row in a table.

(4) Recent changes in privacy laws have disallowed organizations from using Social Security numbers to identify individuals unless certain restrictions are satisfied. As a result, most U.S. universities cannot use SSNs as primary keys (except for financial data). In practice, Student_id, a unique identifier assigned to every student, is likely to be used as the primary key rather than SSN since Student_id can be used throughout the system.

a. Some database designers are reluctant to use generated keys (also known as *surrogate keys*) for primary keys (such as Student_id) because they are artificial. Can you propose any natural choices of keys that can be used to identify the student record in a UNIVERSITY database?

b. Suppose that you are able to guarantee uniqueness of a natural key that includes last name. Are you guaranteed that the last name will not change during the lifetime of the database? If last name can change, what solutions can you propose for creating a primary key that still includes last name but remains unique?

c. What are the advantages and disadvantages of using generated (surrogate) keys?

(10 points)

a, ~~A student~~ Normally, no 2 students with the same name, date of birth would live in the same house, so a possible primary key could be a combination of (name, address, date of birth)

b, Last name might change due to different reasons, including marriage so it would not by itself ensure uniqueness. One solution would be to have a primary key be a combination of (first name, last name, address, date of birth)

c, Advantages:
- Easy to generate (e.g. auto increment)
- Ensure privacy (e.g. avoid stating your ~~social~~ SSN for id → avoid identity theft)
- Simple (e.g. use 1 ID number instead of typing all your name, date of birth, phone number for identification)

Disadvantages:
- Database has to store more data besides natural data like name, address, ...
- Users would have to memorize a random sequence for ID that doesn't make sense to them

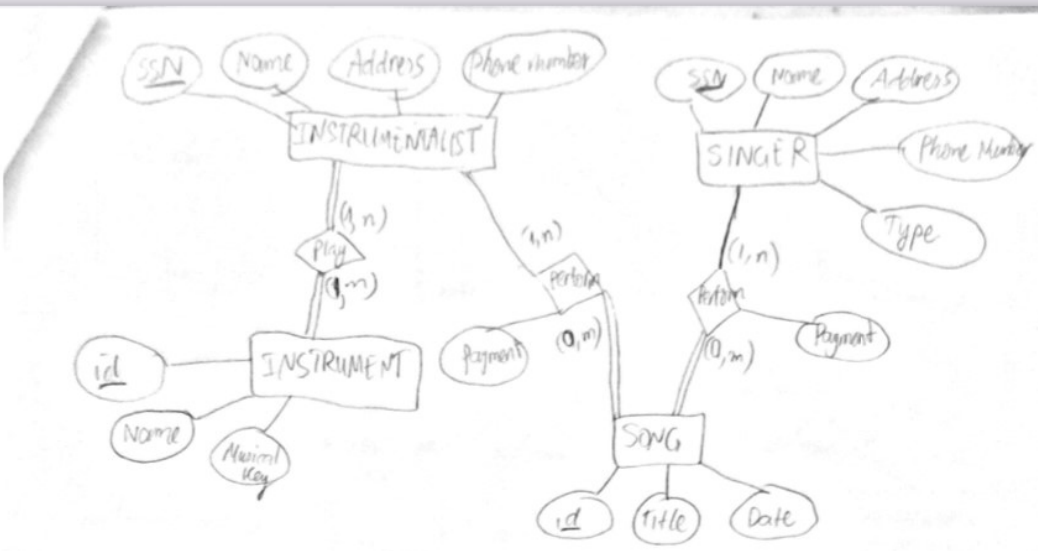
(5) Sun City Records has decided to store information about musicians who record songs in their studio in a database. Design a relational database schema for Sun City Records (including a diagrammatic representation). The following specification describes the mini-world that your database must model. Be sure to indicate all keys and constraints, and any assumptions you make.

- Each musician that records at Sun City has an SSN, a name, an address, and a phone number. SSN uniquely identifies a musician.
- Each instrument used in songs recorded at Sun City has a unique identification number, a name (eg. guitar, flute) and a musical key (eg. C, B-flat, E-flat).
- Each song recorded at Sun City has a title and a date. Songs are uniquely identified by a song id.
- There are two types of musicians recording a song: vocalists and instrumentalists.
- Each instrumentalist may play several instruments, and a given instrument may be played by several instrumentalists. Each instrumentalist plays at least one musical instrument. Each instrument must be played by some instrumentalist.
- Each song is performed by one or more musicians, and a musician may perform a number of songs.
- Each musician who performs a song receives a payment.
- Each song must be performed by some musician.
- A singer has a type (eg. baritone, tenor, and soprano).
- Singers do not play any instruments.

The set of singers and the set of instrumentalists are disjoint.

-5

(25 points)



INSTRUMENTALIST

SSN: char(9) : assume it's U.S. SSN system

Name: Varchar (50): assume name does not exceed 50 characters

Address: Varchar (50): same ?

Phone number: char (10) : assume US phone number no country code

INSTRUMENT:

id: int

Name: varchar (15)

Musical Key: varchar (10): assume not exceeding 10 characters

PLAY:

SSN char (9) : same as instrumentalist's

inst_id int : same as instrument's

PK (SSN, inst_id)

SINGER

SSN, Name, Address, Phone Number : same as instrumentalist

Type: varchar(15), assume Singer type not exceeding 15 characters

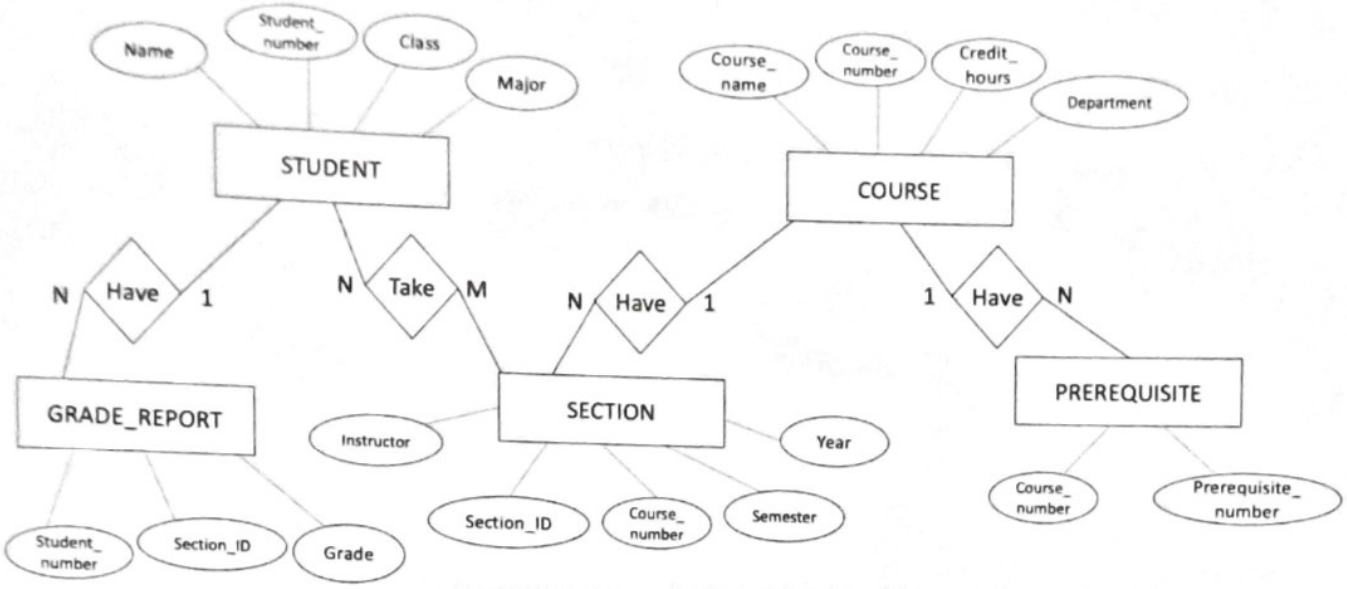
SONG

id int

Title varchar(40)

Date Date data type format YYYY-MM-DD

(6)



For the above CS-Department database schema, write SQL queries for the following data retrievals: (15 points)

(a) Retrieve the names of courses and the semesters they were taught in for all courses taught by Dr. Anderson in 2007 and 2008.

```
SELECT C.Course_name, S.Semester
FROM COURSE AS C, SECTION AS S
WHERE C.Course_number = S.Course_number AND S.Instructor = 'Anderson'
    AND S.Year IN ('2007', '2008');
```

(b) Provide the list of course names and the prerequisites for each course.

```
SELECT C.Course_name, P.Course_name
FROM COURSE AS C, PREREQUISITE AS P, COURSE AS PC
WHERE P.Course_number = C.Course_number AND
      P.Prerequisite_number = PC.Course_number;
```

(c) Retrieve the names of students who scored an 'A' grade.

```
SELECT S.Name
FROM STUDENT AS S, GRADE_REPORT AS G
WHERE S.Student_number = G.Student_number AND
      G.Grade = 'A';
```

(7)

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

WORKS_ON

Essn	Pno	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

Using the data from the COMPANY database snapshot above, provide the data output for the following SQL retrieval commands:

(a) **SELECT** E.FName, E.Minit, E.LName, E.BDate as Date of Birth, E.Sex as Gender
FROM EMPLOYEE as E, DEPARTMENT as D
WHERE D.DName='Research' **AND** D.DNumber=E.DNO;

(5 points)

FName	Minit	LName	Date of Birth	Gender
John	B	Smith	1965-01-09	M
Franklin	T	Wong	1955-12-08	M
Ramesh	K	Narayan	1962-09-15	M
Joyce	A	English	1972-07-31	F

(b) **SELECT** PNumber, DNum, LName, Address, BDate
FROM PROJECTS, DEPARTMENT, EMPLOYEE
WHERE DNum=DNumber **AND** Mgr_SSN=SSN **AND** PLocation in ('Houston','Stafford');

(5 points)

PNumber	DNum	LName	Address	BDate
3	5	Wong	638 Voss, Houston, TX	1955-12-08
10	4	Wallace	291 Berry, Bellaire, TX	1941-06-20
20	1	Borg	450 Stone, Houston, TX	1937-11-10
30	4	Wallace	291 Berry, Bellaire, TX	1941-06-20

7c.

Fname	Minit	LName	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno	Dname	Dnumber	Mgr_ssn	Mgr_start_date
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	33344555	5	Research	5	333445555	1988-05-22
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	33344555	5	Administration	4	987654321	1995-01-01
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	33344555	5	Headquarters	1	888665555	1981-06-19
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5	Research	5	333445555	1988-05-22
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5	Administration	4	987654321	1995-01-01
Franklin	T	Wong	333445555	1955-12-08	638, Voss Houston, TX	M	40000	888665555	5	Headquarters	1	888665555	1981-06-19
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4	Research	5	333445555	1958-05-22
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring TX	F	25000	987654321	4	Administration	4	987654321	1995-01-01
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4	Headquarters	1	888665555	1981-06-19
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4	Research	5	333445555	1958-05-22
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4	Administration	4	987654321	1995-01-01
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4	Headquarters	1	888665555	1981-06-19
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5	Research	5	333445555	1988-05-22
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5	Administration	4	987654321	1995-01-01
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5	Headquarters	1	888665555	1981-06-19
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5				
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4				
James	E	Borg	888665555	1937-11-10	1150 Stone, Houston, TX	M	55000	NULL	1				

(c) **SELECT** *
FROM EMPLOYEE, DEPARTMENT;

(5 points)

Fname Minit tName Ssn Bdate Address Sex Salary Super_ssn Dno Dname Dnumber Mgr_ssn,

Cross product.

(d) (**SELECT DISTINCT** PNumber
FROM PROJECTS, DEPARTMENT, EMPLOYEE
WHERE DNum=DNumber **AND** Mgr_SSN=SSN **AND**
FName='James')

UNION ALL

(**SELECT DISTINCT** PNumber
FROM PROJECTS, WORKS_ON, EMPLOYEE
WHERE PNumber=PNO **AND** ESSN=SSN **AND** FName='James')

(5 points)

Pnumber
20
20

(iii) Insert the tuple ('Production', 4, '943775543', '2007-10-01') into the DEPARTMENT table.

Department with Dnumber = 4 already exists → violate entity integrity constraint (break uniqueness)

Solution : prevent insertion with Restrict or reject keyword