

# Homework 2.

1)

Exercise 1. Consider the instance of the Students relation shown in the following table

sid	name	login	age	gpa
S001	Malcolm	malcom@cs.cmu	19	3.8
S002	Chloe	chlo@cs.cmu	19	3.7
S003	David	david@cs.cmu	19	3.5
S004	Leah	leah@cs.cmu	19	3.9
S005	Eve	eve@cs.cmu	19	3.6
S006	Dan	dan@cs.cmu	19	3.5

- Give an example of an attribute (or set of attributes) that you can deduce is not a candidate key, based on this instance being legal
- Is there any example of an attribute (or set of attributes) that you can deduce is a candidate key, based on this instance being legal?

1a) Non-candidate keys can be name or age. Based on the data that is given, we cannot tell if gpa can be a non-candidate key even though technically more than one student could have the same gpa.

1b) We cannot make predictions on the instances based only on what we have been given. This is because the instance that has been shown for this problem is just one possibility of the relation because at some other time there could be an instance that contains different tuples completely.

**Exercise 2.** Consider the following relations:

Students(*sid*: string, *name*: string, *login*: string,  
*age*: integer, *gpa*: real)

Faculty(*fid*: string, *fname*: string, *sal*: real)

Courses(*cid*: string, *cname*: string, *credits*: integer)

Rooms(*rno*: integer, *address*: string, *capacity*: integer)

Enrolled(*sid*: string, *cid*: string, *grade*: string)

Teaches(*fid*: string, *cid*: string)

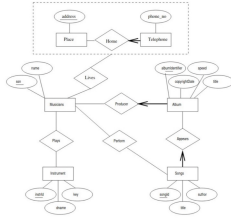
Meets\_In(*cid*: string, *rno*: integer, *time*: string)

- List all the foreign key constraints among these relations.
- Give an example of a (plausible) constraint involving one or more of these relations that is not a primary key or foreign key constraint.

2a) Courses - cid  
Faculty - fid  
Rooms - rno  
Students - sid

2b) Domain constraints : example → cid because it has a string domain constraint.

Exercise 3. Given the following ER diagram (see exercise 1 of homework 1)



Show the SQL statements for creating relations corresponding to the entity sets and relationships. Identify any constraints in the ER diagram that you are unable to capture in the SQL statements and briefly explain why you could not express them.

constraints:

```
CREATE TABLE Musicians (ssn CHAR(10), name CHAR(30), PRIMARY KEY(ssn));
```

```
CREATE TABLE Instruments (instrId CHAR(10), dname CHAR(30), key CHAR(5),
PRIMARY KEY (instrId));
```

```
CREATE TABLE Plays (ssn CHAR(10), instrId INTEGER, PRIMARY KEY (ssn,
instrId), FOREIGN KEY(ssn) REFERENCES Musicians, FOREIGN KEY (instrId)
REFERENCES Instrument);
```

```
CREATE TABLE Songs-Appears (songID INTEGER, author CHAR(30), title
CHAR(30), albumIdentifier INTEGER NOT NULL, PRIMARY KEY (phone),
FOREIGN KEY(address) REFERENCES Place);
```

```
CREATE TABLE Lives (ssn CHAR(10), phone CHAR(11), address CHAR(30),
PRIMARY KEY(ssn, address), FOREIGN KEY(phone, address) References
TelephoneHome, FOREIGN KEY (ssn) REFERENCES Musicians);
```

```
CREATE TABLE Place (address CHAR(30), PRIMARY KEY(address));
```

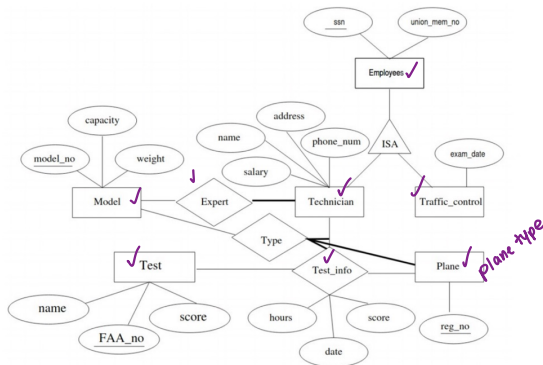
```
CREATE TABLE Perform (songId INTEGER, ssn (CHAR(10), PRIMARY KEY (ssn,
songId), FOREIGN KEY (songId) REFERENCES Songs, FOREIGN KEY (ssn)
REFERENCES Musicians);
```

```
CREATE TABLE Album Producer (albumIdentifier INTEGER, ssn CHAR(10),
copyrightDate DATE, speed INTEGER, title CHAR(30), PRIMARY KEY
(albumIdentifier), FOREIGN KEY (ssn) REFERENCES Musicians);
```

not NULL!

4)

**Exercise 4.** Given the following ER diagram (see exercise 2 of homework 1)



Translate it into a relational schema. If there are constraints that cannot be captured by your translation, explain why.

- CREATE TABLE Expert(ssn CHAR(11), model\_no INTEGER, PRIMARYKEY(ssn, model\_no), FOREIGN KEY(ssn) REFERENCES Technician, FOREIGN KEY (model\_no) REFERENCES Models);
- CREATE TABLE Model (model\_no INTEGER, capacity INTEGER, weight INTEGER, PRIMARY KEY (model\_no));
- CREATE TABLE Employees (ssn CHAR(11), union\_mem\_no INTEGER, PRIMARY KEY(ssn));
- CREATE TABLE Technician (ssn CHAR(11), name CHAR(20), address CHAR(20), phone\_no CHAR(14), PRIMARY KEY (ssn), FOREIGN KEY (ssn) REFERENCES Employees ON DELETE CASCADE);
- CREATE TABLE traffic\_control (ssn CHAR(11), exam\_date DATE, PRIMARY KEY (ssn), FOREIGN KEY (ssn) REFERENCES Employees ON DELETE CASCADE);
- CREATE TABLE Plane\_Type (Reg\_no INTEGER, Model\_no INTEGER, Primary KEY (reg\_no), FOREIGN KEY(model\_no) REFERENCES Models);
- CREATE TABLE Test\_info (FAA\_no INTEGER, ssn CHAR(11), reg\_no INTEGER, hours INTEGER, date DATE, score INTEGER, PRIMARY KEY (ssn, reg\_no, FAA\_no), FOREIGN KEY(reg\_no) REFERENCES Plane\_Type, FOREIGN KEY(FAA\_no) REFERENCES Tesr, FOREIGN KEY (ssn) REFERENCES Employees);
- CREATE TABLE Test (FAA\_no INTEGER, name CHAR(10), max\_score INTEGER, hours INTEGER, date DATE, score INTEGER, PRIMARY KEY(FAA\_no);

For this, I accidentally did it using CREATE TABLE like in problem 3, but I redid it in terms of a relational schema and have attached it below.

## Question 4: ACTUAL ANSWER:

Expert (ssn CHAR(11),  
model\_no INTEGER,  
PRIMARY KEY (SSN, model\_no)  
FOREIGN KEY (SSN) REFERENCES Technician,  
FOREIGN KEY (model\_no) REFERENCES Models)

Models (model\_no INTEGER,  
capacity INTEGER, Weight INTEGER,  
PRIMARY KEY (model\_no))

Employees (ssn CHAR (11),  
union\_mem\_no INTEGER,  
PRIMARY KEY (ssn))

Technician\_Employees (ssn CHAR (11),  
name CHAR(20),  
address CHAR (20),  
phone\_no CHAR (14),  
PRIMARY KEY (ssn),  
FOREIGN KEY (ssn) REFERENCES Employees)

Traffic\_control\_Employees (ssn CHAR (11),  
exam\_date DATE,  
PRIMARY KEY (ssn),  
FOREIGN KEY (ssn) REFERENCES Employees)

Plane\_Type (reg\_no INTEGER,  
model\_no INTEGER,  
PRIMARY KEY (reg\_no),  
FOREIGN KEY (model\_no) REFERENCES Models)

Test\_info(FAA\_no INTEGER,  
ssn CHAR (11),  
reg\_no INTEGER,  
hours INTEGER,  
date DATE,  
Score INTEGER,  
PRIMARY KEY (ssn, reg\_no, FAA\_no),  
FOREIGN KEY (reg\_no) REFERENCES Plane\_Type,  
FOREIGN KEY (FAA\_no) REFERENCES Test,  
FOREIGN KEY (ssn) REFERENCES Employees)