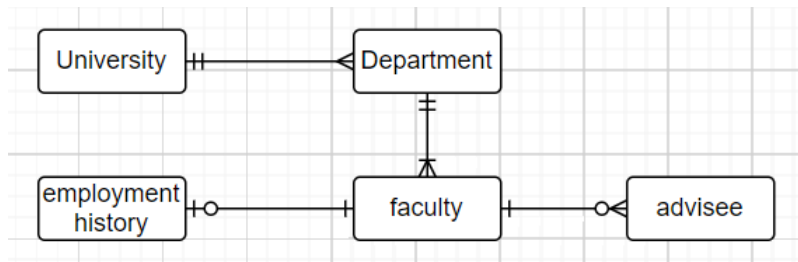


Noah Buchanan  
ua100  
Problem Set 2  
Database Systems II at 9:30 AM

January 26, 2021

### Entity Relationship Modeling



### Normalization

1. RESID→EMPID, ROOM, FNAME, LNAME, START, END, BLDG, ADDRESS  
EMPID→FNAME, LNAME  
BLDG→ADDRESS
2. Primary Key: RESID  
Alternate Key: (START,BLDG,ROOM), (START,ADDRESS,ROOM),  
(and lastly any combination that contains RESID as a determinant works obviously because it can uniquely identify every row alone)
3. The table is not presently in 3NF
4. modification anomaly: changing the address of a building would require you to individually change it for all rows  
insertion anomaly: you would have no place to put a new address either if a new building or address was added, you would need an empty row with filler data just to put it in  
deletion anomaly: deleting a employee from the database would once again

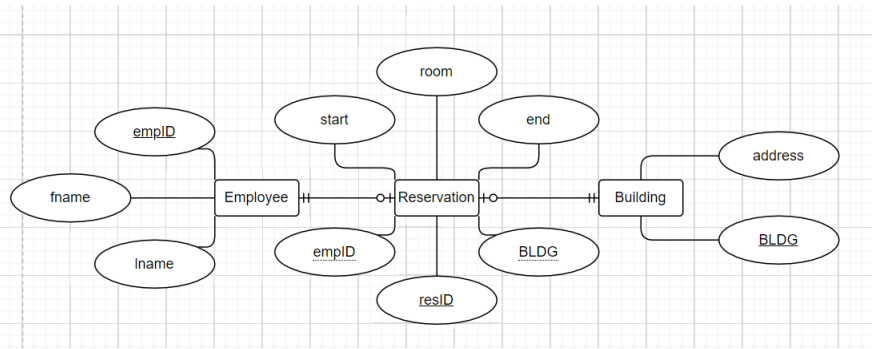
require you to go through every single row with them and individually delete them

5. 1NF: RESID->EMPID, ROOM, FNAME, LNAME, START, END, BLDG, ADDRESS  
EMPID->FNAME, LNAME  
BLDG->ADDRESS

2NF: RESID->EMPID, ROOM, FNAME, LNAME, START, END, BLDG, ADDRESS  
EMPID->FNAME, LNAME  
BLDG->ADDRESS

(it was already in 2NF, there were no partial dependencies)

3NF: RESID->EMPID, ROOM, START,END,BLDG  
EMPID->FNAME, LNAME  
BLDG->ADDRESS



6.

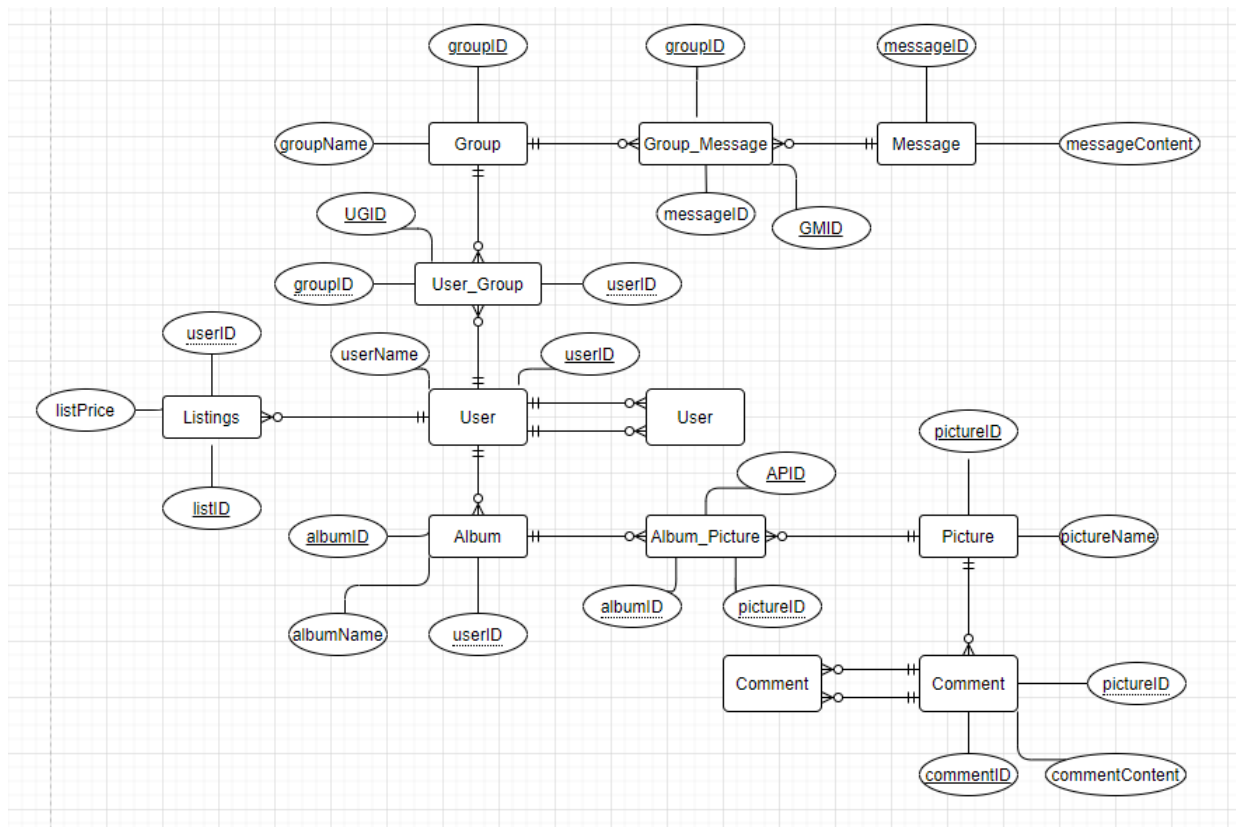
```
CREATE TABLE RESERVATION(  
  RESID INT NOT NULL,  
  EMPID INT,  
  ROOM VARCHAR(255),  
  START TIME,  
  END TIME,  
  BLDG VARCHAR(255),  
  PRIMARY KEY(RESID)  
);
```

```
CREATE TABLE BUILDING(  
  BLDG VARCHAR(255) NOT NULL,  
  ADDRESS VARCHAR(255),  
  PRIMARY KEY(BLDG)
```

```
);
```

```
CREATE TABLE EMPLOYEE(  
  EMPID INT NOT NULL,  
  FNAME VARCHAR(255),  
  LNAME VARCHAR(255),  
  PRIMARY KEY(EMPID)  
);
```

## Social Networking Schema



```
CREATE TABLE USER(  
  USERID INT NOT NULL,  
  USERNAME VARCHAR(255),  
  PRIMARY KEY(USERID)  
);
```

```
CREATE TABLE ALBUM(  
  ALBUMID INT NOT NULL,
```

```

    ALBUMNAME VARCHAR(255),
    USERID INT REFERENCES USER(USERID),
    PRIMARY KEY(ALBUMID)
);

CREATE TABLE ALBUM_PICTURE(
    APID INT NOT NULL,
    ALBUMID INT REFERENCES ALBUM(ALBUMID),
    PICTUREID INT REFERENCES PICTURE(PICTUREID),
    PRIMARY KEY(APID)
);

CREATE TABLE PICTURE(
    PICTUREID INT NOT NULL,
    PICTURENAME VARCHAR(255),
    PRIMARY KEY(PICTUREID)
);

CREATE TABLE COMMENT(
    COMMENTID INT NOT NULL,
    PICTUREID INT REFERENCES PICTURE(PICTUREID),
    COMMENTCONTENT VARCHAR(255),
    PRIMARY KEY(COMMENTID)
);

CREATE TABLE LISTING(
    LISTID INT NOT NULL,
    USERID INT REFERENCES USER(USERID),
    LISTPRICE VARCHAR(255),
    PRIMARY KEY(LISTID)
);

CREATE TABLE USER_GROUP(
    UGID INT NOT NULL,
    GROUPID INT REFERENCES GROUP(GROUPID),
    USERID INT REFERENCES USER(USERID),
    PRIMARY KEY(UGID)
);

CREATE TABLE GROUP(
    GROUPID INT NOT NULL,
    GROUPNAME VARCHAR(255),
    PRIMARY KEY(GROUPID)
);

CREATE TABLE GROUP_MESSAGE(

```

```
GMID INT NOT NULL,  
GROUPID INT REFERENCES GROUP(GROUPID),  
MESSAGEID INT REFERENCES MESSAGE(MESSAGEID),  
PRIMARY KEY(GMID)  
);
```

```
CREATE TABLE MESSAGE(  
MESSAGEID INT NOT NULL,  
MESSAGECONTENT VARCHAR(255),  
PRIMARY KEY(MESSAGEID)  
);
```

(not necessarily in this exact order this is just the order they were pasted in)