

Marist College

Albany High School Database

Genevieve Anderson
CMPT 308: Database Management, section 901
Professor Shwartz
November 11th, 2021

Table of Contents

3: Business Rules:

4: ER Diagram:

5: FINTEACHER, table description, normalization, definition of keys, create statement

6: FINCOLLEGE, table description, normalization, definition of keys, create statement

7: FINCLASSROOM, table description, normalization, definition of keys, create statement

8: FINCLASS, table description, normalization, definition of keys, create statement

9: FINSTUDENT, table description, normalization, definition of keys, create statement

10: FINFAMILY, table description, normalization, definition of keys, create statement

11: FINSPORTS, table description, normalization, definition of keys, create statement

12: FINCOACH, table description, normalization, definition of keys, create statement

13: FINGAMES, table description, normalization, definition of keys, create statement

14: FINPARTICIPATES, table description, normalization, definition of keys, create statement

15: FINTAKES, table description, normalization, definition of keys, create statement

16: FINQuery1 (Universal Quantifier), Query description, SQL, table, cardinality

17: FINQuery2 (Only), Query description, SQL, table, cardinality

18: FINQuery3 (None), Query description, SQL, table, cardinality

19: FINQuery4 (Left Join), Query description, SQL, table, cardinality

20: FINQuery5 (Right Join), Query description, SQL, table, cardinality

21: FINQuery6 (Full Join), Query description, SQL, table, cardinality

22: FINQuery7 (6 Tables), Query description, SQL, table, cardinality

23: FINQuery8 (Lab 2/3 Level), Query description, SQL, table, cardinality

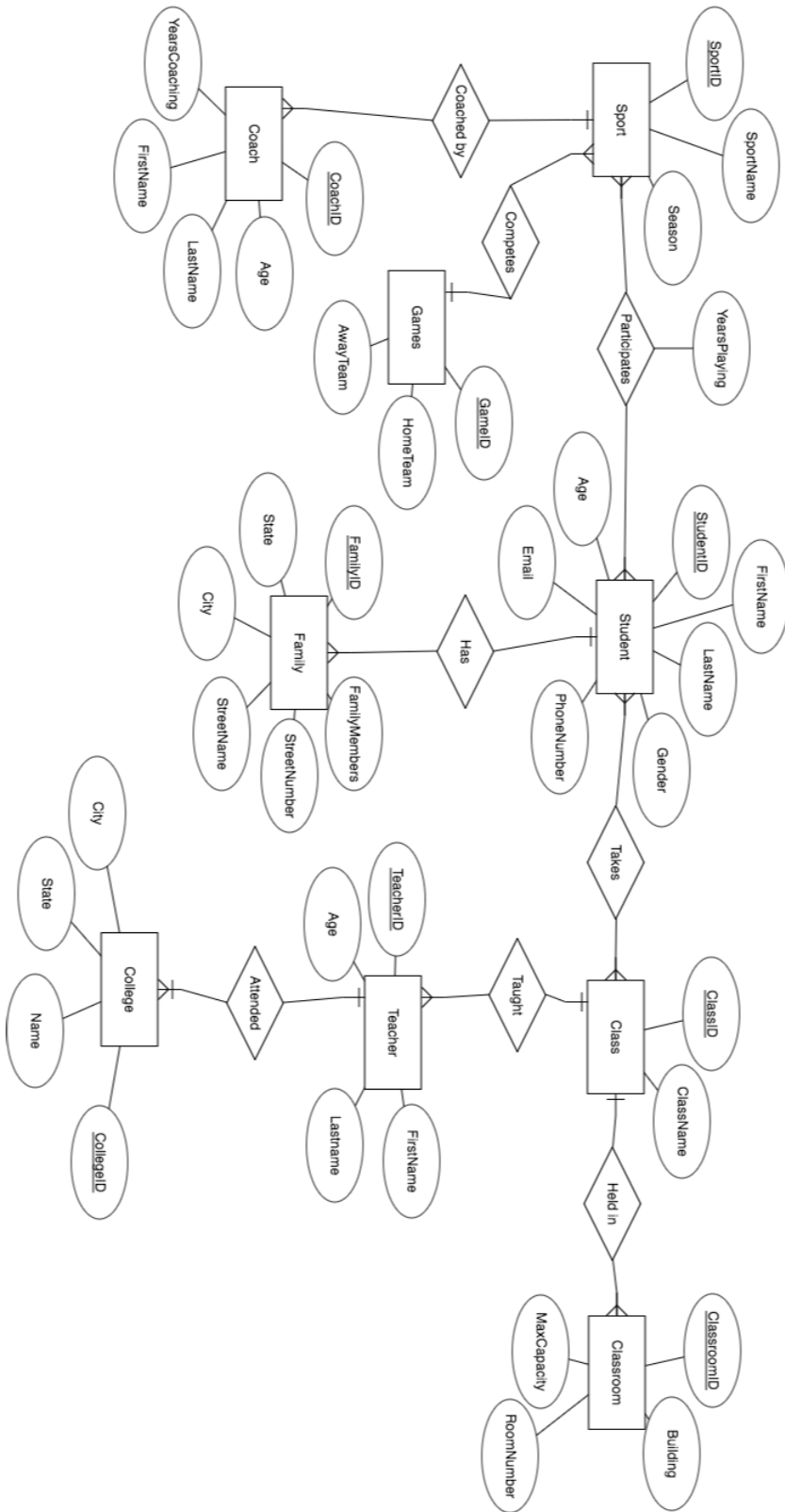
24: FINQuery9 (Lab 2/3 Level), Query description, SQL, table, cardinality

25: FINQuery10 (Lab 2/3 Level), Query description, SQL, table, cardinality

Business rules: Albany High School 2021

- All **students** that attend Albany High School have a **family**.
- Each family can have more than one child at Albany High School.
- Each family has a total number of family members, and an address in which they reside.
- Each family's address is composed of a street number, a street name, a city, and a state.
- Each student that attends Albany High School attends several **classes** .
- Each student has an ID number, a first name, a last name, age, gender, email and a phone number.
- Each class that students take has an ID, and a name which describes the material of the class.
- Students take several classes.
- Each class takes place in a **classroom**.
- Each classroom has an unique ID, a building, a room number, and a maximum capacity of people that can be in the room at once.
- Each class is taught by a **teacher**.
- A teacher can teach more than one class.
- Each teacher has an unique ID, a first name, a last name, an age, and a **college** in which they attended.
- Each college has an unique ID, a name, a city, and a state in which the college is located in .
- For each of the three seasons, every student has the option to play a **sport**.
- Each sport has a unique ID, a name, and a specific season in which it is played.
- Each **coach** is able to coach up to one sport each season
- Each coach has a unique iD, a first name, a last name, an age, and a specific number of years that they have been coaching.
- Each sport participates in **games**.
- Each game has a unique ID, a home team, and an away team.

ER Diagram



FINTEACHER: Provides data pertaining to first and last names of teachers, and their age

PK: TeacherID

FK: CollegeID references FINCOLLEGE

This table is in 3NF because there are no partial or transitive dependencies.

```
CREATE TABLE "901ANDERSONG"."FINTEACHER"
(
  "TEACHERID" NUMBER(*,0),
  "FIRSTNAME" NVARCHAR2(30),
  "LASTNAME" NVARCHAR2(30),
  "AGE" NUMBER(*,0),
  "COLLEGEID" NUMBER(*,0),
  CONSTRAINT "PK_FINTEACHER" PRIMARY KEY ("TEACHERID"),
  CONSTRAINT "FK_FINTEACHER_FINCOLLEGE" FOREIGN KEY
("COLLEGEID")
  REFERENCES "901ANDERSONG"."FINCOLLEGE" ("COLLEGEID") ENABLE
);
```

FINCOLLEGE: Provides data pertaining to where teachers attended college. Provides information on college name, city, and state in which the school is located.

This table is in 3NF because there are no partial or transitive dependencies.

PK: CollegeID

FK: none

```
CREATE TABLE "901ANDERSON"."FINCOLLEGE"
(
  "COLLEGEID" NUMBER(*,0),
  "NAME" NVARCHAR2(30),
  "CITY" NVARCHAR2(30),
  "STATE" NVARCHAR2(30),
  CONSTRAINT "PK_FINCOLLEGE" PRIMARY KEY ("COLLEGEID")
);
```

FINCLASSROOM: Provides data pertaining to the building in which the classroom is located along with room number and the maximum capacity of the room.

This table is in 3NF because there are no partial or transitive dependencies.

PK: ClassroomID

FK: none

```
CREATE TABLE "901ANDERSON"."FINCLASSROOM"
(
  "CLASSROOMID" NUMBER(*,0),
  "BUILDING" NVARCHAR2(30),
  "ROOMNUMBER" NUMBER(*,0),
  "MAXCAPACITY" NUMBER(*,0),
  CONSTRAINT "PK_FINCLASSROOM" PRIMARY KEY ("CLASSROOMID")
);
```

FINCLASS: Provides data pertaining to the name of the class

This table is in 3NF because there are no partial or transitive dependencies.

PK: ClassID

FK: ClassroomID references FINCLASSROOM, TeacherID references FINTEACHER

```
CREATE TABLE "901ANDERSONG"."FINCLASS"
(
  "CLASSID" NUMBER(*,0),
  "CLASSROOMID" NUMBER(*,0),
  "TEACHERID" NUMBER(*,0),
  "CLASSNAME" NVARCHAR2(30),
  CONSTRAINT "PK_FINCLASS" PRIMARY KEY ("CLASSID"),
  CONSTRAINT "FK_FINCLASS_FINCLASSROOM" FOREIGN KEY
("CLASSROOMID")
  REFERENCES "901ANDERSONG"."FINCLASSROOM" ("CLASSROOMID")
  ENABLE,
  CONSTRAINT "FK_FINCLASS_FINTEACHER" FOREIGN KEY ("TEACHERID")
  REFERENCES "901ANDERSONG"."FINTEACHER" ("TEACHERID") ENABLE
);
```


FINSTUDENT: Provides data pertaining to the students first and last name, their email, age, gender, and phone number

This table is in 3NF because there are no partial or transitive dependencies.

PK: StudentID

FK: FamilyID references FINFAMILY

```
CREATE TABLE "901ANDERSONG"."FINSTUDENT"  
(  
    "STUDENTID" NUMBER(*,0),  
    "FIRSTNAME" NVARCHAR2(30),  
    "LASTNAME" NVARCHAR2(30),  
    "FAMILYID" NUMBER(*,0),  
    "EMAIL" NVARCHAR2(30),  
    "AGE" NUMBER(*,0),  
    "GENDER" NVARCHAR2(30),  
    "PHONENUMBER" NUMBER(*,0),  
    CONSTRAINT "PK_FINSTUDENT" PRIMARY KEY ("STUDENTID"),  
    CONSTRAINT "FK_FINSTUDENT_FINFAMILY" FOREIGN KEY ("FAMILYID")  
        REFERENCES "901ANDERSONG"."FINFAMILY" ("FAMILYID") ENABLE  
);
```

FINFAMILY: Provides data pertaining to the number of family members in each family, along with the address in which the family resides broken down into street number, street name, city, and state

This table is in 3NF because there are no partial or transitive dependencies.

PK: FamilyID

FK: none

```
CREATE TABLE "901ANDERSONG"."FINFAMILY"
(   "FAMILYID" NUMBER(*,0),
    "FAMILYMEMBERS" NUMBER(*,0),
    "STREETNUMBER" NUMBER(*,0),
    "STREETNAME" NVARCHAR2(30),
    "CITY" NVARCHAR2(30),
    "STATE" NVARCHAR2(30),
    CONSTRAINT "PK_FINFAMILY" PRIMARY KEY ("FAMILYID")
);
```

FINSPTS: provides data pertaining to the name of the sport and what season it is played

This table is in 3NF because there are no partial or transitive dependencies.

PK: SportID

FK: CoachID references FINCOACH

```
CREATE TABLE "901ANDERSONG"."FINSPTS"  
(  
    "SPORTID" NUMBER(*,0),  
    "SPORTNAME" NVARCHAR2(30),  
    "SEASON" NVARCHAR2(30),  
    "COACHID" NUMBER(*,0),  
    CONSTRAINT "PK_FINSPTS" PRIMARY KEY ("SPORTID"),  
    CONSTRAINT "FK_FINSPTS_FINCOACH" FOREIGN KEY ("COACHID")  
        REFERENCES "901ANDERSONG"."FINCOACH" ("COACHID") ENABLE  
) ;
```

FINCOACH: Provides data pertaining to the coaches first and last name, how long they have been coaching, and their age

This table is in 3NF because there are no partial or transitive dependencies.

PK: CoachID

FK: none

```
CREATE TABLE "901ANDERSON"."FINCOACH"
(
  "COACHID" NUMBER(*,0),
  "FIRSTNAME" NVARCHAR2(30),
  "LASTNAME" NVARCHAR2(30),
  "YEARS COACHING" NUMBER(*,0),
  "AGE" NUMBER(*,0),
  CONSTRAINT "PK_FINCOACH" PRIMARY KEY ("COACHID")
);
```

FINGAMES: Provides data pertaining to the home and away team at each game

This table is in 3NF because there are no partial or transitive dependencies.

PK: GameID

FK: SportID references FINSPORTS

```
CREATE TABLE "901ANDERSONG"."FINGAMES"  
(  
    "GAMEID" NUMBER(*,0),  
    "HOMETEAM" NVARCHAR2(30),  
    "AWAYTEAM" NVARCHAR2(30),  
    "SPORTID" NUMBER(*,0),  
    CONSTRAINT "PK_FINGAMES" PRIMARY KEY ("GAMEID"),  
    CONSTRAINT "FK_FINGAMES_FINSPORTS" FOREIGN KEY ("SPORTID")  
        REFERENCES "901ANDERSONG"."FINSPORTS" ("SPORTID") ENABLE  
);
```

FINPARTICIPATES: Provides data pertaining to what students participate in what sports, along with how many years the student has been playing each sport they play

This table is in 3NF because there are no partial or transitive dependencies.

PK: StudentId, SportID

FK: StudentID references FINSTUDENT, SportID references FINSPORTS

```
CREATE TABLE "901ANDERSONG"."FINPARTICIPATES"
(
  "STUDENTID" NUMBER(*,0),
  "SPORTID" NUMBER(*,0),
  "YEARSPLAYING" NUMBER(*,0),
  CONSTRAINT "FK_FINPARTICIPATES_FINSTUDENT" FOREIGN KEY
("STUDENTID")
  REFERENCES "901ANDERSONG"."FINSTUDENT" ("STUDENTID") ENABLE,
  CONSTRAINT "FK_FINPARTICIPATES_FINSPTS" FOREIGN KEY
("SPORTID")
  REFERENCES "901ANDERSONG"."FINSPTS" ("SPORTID") ENABLE
);
```

FINTAKES: Provides data pertaining to what students take what classes.

This table is in 3NF because there are no partial or transitive dependencies.

PK: StudentID, ClassID

FK: StudentID references FINSTUDENT, ClassID references FINCLASS

```
CREATE TABLE "901ANDERSONG"."FINTAKES"
(
  "STUDENTID" NUMBER(*,0),
  "CLASSID" NUMBER(*,0),
  CONSTRAINT "PK_FINTAKES" PRIMARY KEY ("STUDENTID", "CLASSID")
  CONSTRAINT "FK_FINTAKES_FINSTUDENT" FOREIGN KEY ("STUDENTID")
    REFERENCES "901ANDERSONG"."FINSTUDENT" ("STUDENTID") ENABLE,
  CONSTRAINT "FK_FINTAKES_FINCLASS" FOREIGN KEY ("CLASSID")
    REFERENCES "901ANDERSONG"."FINCLASS" ("CLASSID") ENABLE
);
```

FINQuery1: Give the full name of students that take every class

```
CREATE OR REPLACE VIEW FINQuery1 AS
SELECT STU.FIRSTNAME, STU.LASTNAME
FROM FINSTUDENT STU
WHERE NOT EXISTS
  (SELECT *
   FROM FINCLASS CLASS
   WHERE NOT EXISTS
     (SELECT *
      FROM FINTAKES TAKE
      WHERE STU.STUDENTID = TAKE.STUDENTID
      AND TAKE.CLASSID = CLASS.CLASSID));
```

FIRSTNAME	LASTNAME
Genevieve	Anderson

Cardinality: 1

FINQuery2: Give the full name of students that only take STEM classes (Calculus, Engineering, Computer Science, Statistics, Chemistry, Biology)

```
CREATE OR REPLACE VIEW FinQuery2 AS
SELECT STU.FIRSTNAME, STU.LASTNAME
FROM FINSTUDENT STU
WHERE STU.STUDENTID NOT IN
  (SELECT TAKE.STUDENTID
   FROM FINTAKES TAKE
   WHERE TAKE.CLASSID NOT IN
     (SELECT CLASS.CLASSID
      FROM FINCLASS CLASS
      WHERE CLASS.CLASSNAME = 'Calculus'
      OR CLASS.CLASSNAME = 'Engineering'
      OR CLASS.CLASSNAME = 'Computer Science'
      OR CLASS.CLASSNAME = 'Statistics'
      OR CLASS.CLASSNAME = 'Chemistry'
      OR CLASS.CLASSNAME = 'Biology'));
```

FIRSTNAME	LASTNAME
Hannah	Chaput
Jacob	Brooks
Olivia	Anderson
Natalie	Tidler
Mackenzie	Mitchell

Cardinality: 5

FINQuery3: Give the full name of coaches who coach none of the fall sports

```
CREATE OR REPLACE VIEW FINQuery3 AS
SELECT COACH.FIRSTNAME, COACH.LASTNAME
FROM FINCOACH COACH
WHERE COACH.COACHID NOT IN
  (SELECT SPORT.COACHID
   FROM FINSPORTS SPORT
   WHERE SPORT.SEASON = 'Fall');
```

FIRSTNAME	LASTNAME
Tom	Sanford

Cardinality: 1

FINQuery4: Give the full name of all students along with what sports they play, if they play a sport at all

```
CREATE OR REPLACE VIEW FINQuery4 AS
SELECT STU.FIRSTNAME, STU.LASTNAME, SPORT.SPORTNAME
FROM FINSTUDENT STU
LEFT JOIN FINPARTICIPATES PART ON STU.STUDENTID = PART.STUDENTID
LEFT JOIN FINSPORTS SPORT ON PART.SPORTID = SPORT.SPORTID;
```

FIRSTNAME	LASTNAME	SPORTNAME
Genevieve	Anderson	Swimming
Jacob	Brooks	Soccer
Timothy	Anderson	Soccer
Ashley	Surette	Cross County
Ryan	Brooks	Cross County
Genevieve	Anderson	Skiing
Caroline	Crowell	Skiing
Timothy	Anderson	Hockey
Jacob	Brooks	Basketball
Hannah	Chaput	Track
Ashley	Surette	Lacrosse
Hannah	Brooks	Lacrosse
Genevieve	Anderson	Rowing
Caroline	Crowell	Rowing
Jacob	Brooks	Rowing
Olivia	Anderson	Rowing
Lauren	Muhr	
Caroline	Voorhis	
Jonathon	Crowell	
Natalie	Tidler	
Mackenzie	Mitchell	

Cardinality: 21

FINQuery5: Give the full name of all teachers, along with the classes they teach, if they teach any classes

```
CREATE OR REPLACE VIEW FINQuery5 AS
SELECT CLASS.CLASSNAME, TEACH.FIRSTNAME, TEACH.LASTNAME
FROM FINCLASS CLASS
RIGHT JOIN FINTEACHER TEACH ON TEACH.TEACHERID = CLASS.TEACHERID;
```

CLASSNAME	FIRSTNAME	LASTNAME
Biology	Cathy	Martensen
Chemistry	Cathy	Martensen
Statistics	Duy	Nguyen
Economics	Duy	Nguyen
Calculus	Duy	Nguyen
Computer Science	Donald	Schwartz
Engineering	Aaron	Kippins
Art	Scott	Frank
Business	Scott	Frank
History	Scott	Frank
	Matthew	Johnson
	Joe	Kirtland

Cardinality: 12

FINQuery6: Give all classes, their buildings and room numbers, and the full names of the teachers that teach the classes. Include classes, buildings and room numbers, and teachers for which are not associated with any class, building and room number, or teacher.

```
CREATE OR REPLACE VIEW FINQuery6 AS
SELECT CLASS.CLASSNAME, ROOM.BUILDING, ROOM.ROOMNUMBER,
TEACH.FIRSTNAME, TEACH.LASTNAME
FROM FINCLASS CLASS
FULL JOIN FINCLASSROOM ROOM ON CLASS.CLASSROOMID =
ROOM.CLASSROOMID
FULL JOIN FINTEACHER TEACH ON CLASS.TEACHERID = TEACH.TEACHERID;
```

CLASSNAME	BUILDING	ROOMNUMBER	FIRSTNAME	LASTNAME
Business	Fontaine	213	Scott	Frank
Economics	Hancock	1021	Duy	Nguyen
Biology	Dyson	118	Cathy	Martensen
Calculus	Hancock	2005	Duy	Nguyen
History	Fontaine	213	Scott	Frank
Engineering	Hancock	1021	Aaron	Kippins
Computer Science	Hancock	1021	Donald	Schwartz
Art	Fontaine	213	Scott	Frank
Statistics	Hancock	2005	Duy	Nguyen
Chemistry	Dyson	118	Cathy	Martensen
	Steel Plant	1030		
	Donnelly	101		
	Lowell Thomas	230		
			Matthew	Johnson
			Joe	Kirtland

Cardinality: 15

FINQuery7: Give the full name of students that take at least one class where the teacher attended Harvard

```
CREATE OR REPLACE VIEW FINQuery7 AS
SELECT STU.FIRSTNAME, STU.LASTNAME
FROM FINSTUDENT STU
WHERE STU.STUDENTID IN
  (SELECT TAKE.STUDENTID
   FROM FINTAKES TAKE
   WHERE TAKE.CLASSID IN
     (SELECT CLASS.CLASSID
      FROM FINCLASS CLASS
      WHERE CLASS.TEACHERID IN
        (SELECT TEACH.TEACHERID
         FROM FINTEACHER TEACH
         WHERE TEACH.COLLEGEID IN
           (SELECT COLL.COLLEGEID
            FROM FINCOLLEGE COLL
            WHERE COLL.NAME = 'Harvard'))));
```

FIRSTNAME	LASTNAME
Genevieve	Anderson
Caroline	Crowell
Hannah	Chaput
Lauren	Muhr
Jacob	Brooks
Timothy	Anderson
Olivia	Anderson
Ryan	Brooks
Jonathon	Crowell
Natalie	Tidler
Mackenzie	Mitchell

Cardinality: 11

FINQuery8: Give the full name of students that play more than 1 sport

```
CREATE OR REPLACE VIEW FINQuery8 AS
SELECT STU.FIRSTNAME, STU.LASTNAME
FROM FINSTUDENT STU, FINPARTICIPATES PART
WHERE STU.STUDENTID = PART.STUDENTID
GROUP BY STU.FIRSTNAME, STU.LASTNAME
HAVING COUNT(PART.STUDENTID) > 1;
```

FIRSTNAME	LASTNAME
Genevieve	Anderson
Ashley	Surette
Timothy	Anderson
Caroline	Crowell
Jacob	Brooks

Cardinality: 5

FINQuery9: Give the full name and email of students who played in a soccer game against hyde park

```
CREATE OR REPLACE VIEW FINQuery9 AS
SELECT STU.FIRSTNAME, STU.LASTNAME, STU.EMAIL
FROM FINSTUDENT STU, FINPARTICIPATES PART, FINSPORTS SPORT, FINGAMES
GAME
WHERE STU.STUDENTID = PART.STUDENTID
AND PART.SPORTID = SPORT.SPORTID
AND SPORT.SPORTID = GAME.SPORTID
AND SPORT.SPORTNAME = 'Soccer'
AND (GAME.AWAYTEAM = 'Hyde Park'
OR GAME.HOMETEAM = 'Hyde Park');
```

FIRSTNAME	LASTNAME	EMAIL
Jacob	Brooks	jacobbrooks@comcast.net
Timothy	Anderson	tjand123@hotmail.com

Cardinality: 2

FINQuery10: Give full names and age of teachers that teach students that have families that live in a city which is last in the alphabetic list of all cities that students' families live in.

```
CREATE OR REPLACE VIEW FINQuery10 AS
SELECT TEACH.FIRSTNAME, TEACH.LASTNAME, TEACH.AGE
FROM FINTEACHER TEACH
WHERE TEACH.TEACHERID IN
  (SELECT CLASS.TEACHERID
   FROM FINCLASS CLASS
   WHERE CLASS.CLASSID IN
     (SELECT TAKE.CLASSID
      FROM FINTAKES TAKE
      WHERE TAKE.STUDENTID IN
        (SELECT STU.STUDENTID
         FROM FINSTUDENT STU
         WHERE STU.FAMILYID IN
           (SELECT FAM.FAMILYID
            FROM FINFAMILY FAM
            WHERE FAM.CITY IN
              (SELECT MAX(FAMLAST.CITY)
               FROM FINFAMILY FAMLAST))))));
```

FIRSTNAME	LASTNAME	AGE
Cathy	Martensen	35
Duy	Nguyen	33
Donald	Schwartz	31
Aaron	Kippins	37
Scott	Frank	35

Cardinality: 5