Database System using SQLite

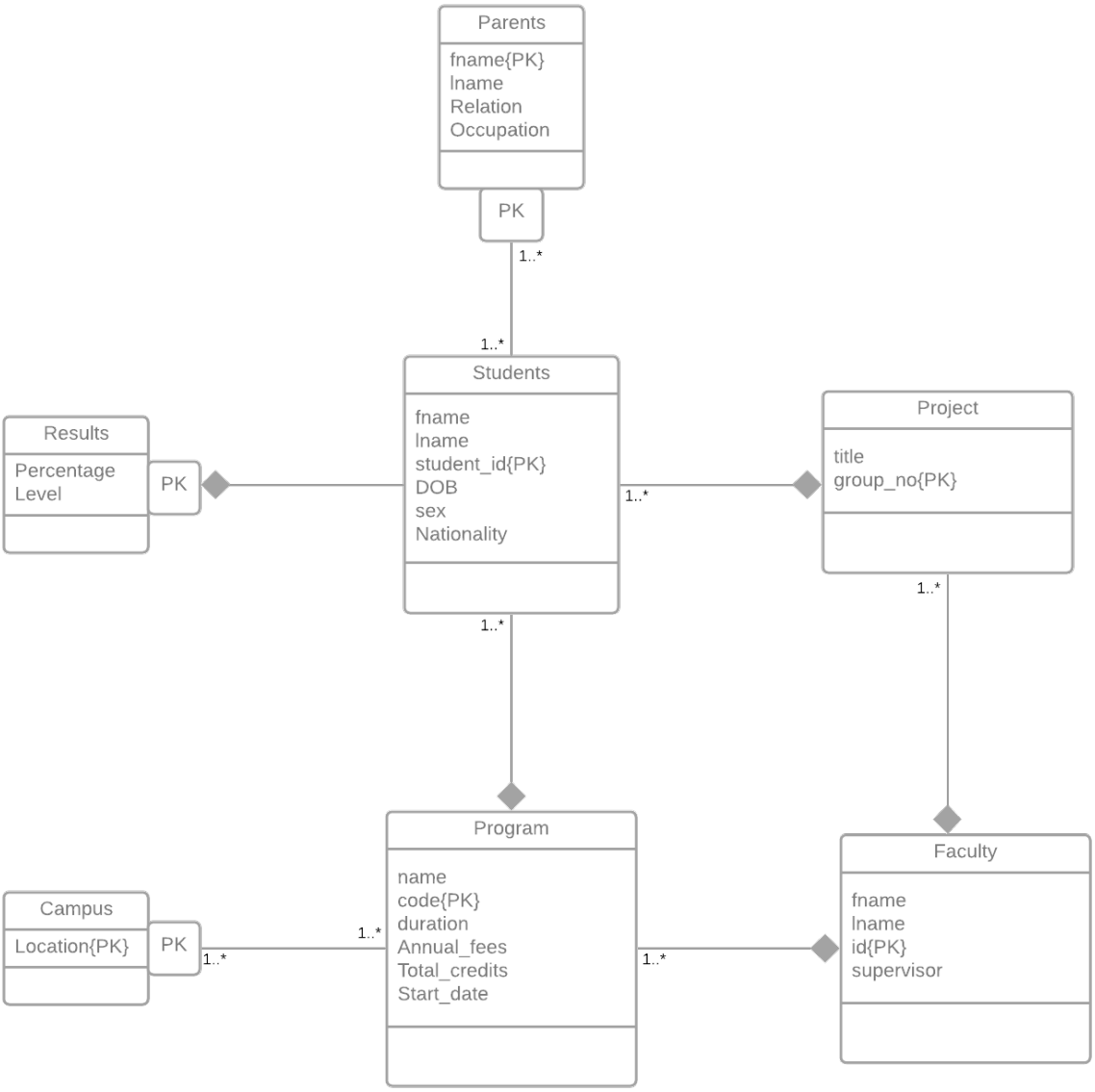
Prateek Kumar Singh

Section 1: Overview

The Database created is intended to store information regarding Students at a University. Description of the database is as follows:

- Each Student has a unique student_id and is strong entity set with following relationships with other entities:
    - 1:M with Parents, Since a student can have two parents.
    - 1:1 with Project, Since a student can only be assigned one Project.
    - 1:1 with Results, Since a student can have only one result.
    - 1:1 with Program, Since a student shall only be enrolled in one Program.
- Each Project has a Unique group_no and title. It is a strong entity set and has 1:M relationship with Students, since every project group is constituted from several students. Project has a 1:1 relationship with Faculties, since there can only one coordinator assigned to particular project.
- Each Faculty has a unique id and is a strong entity set. It has 1:M relationship with both Program and Project, since a faculty can coordinate more than one program and can also mentor more than one Project.
- Each Program has a unique code and is a strong entity set. It has a 1:M relationship with both campus and students, since a Program can be run in multiple campuses, and it can have several students in a single program.
- Results and Parents are weak entity set, since their existence is ensured by the students entityset. Parents has a 1:M relationship with students, since a parent can have more than one child at the university.
- Campus is a weak entity set, since its existence is ensured by the Program entityset. It has a 1:M relationship with program, since a campus can have more than one running programs.

Section 2: ER diagram and relational database schema.



Students(fname, lname, student_id, group_no*, code*, DOB, sex, Nationality)

Program(name, code, duration, Annual_fees, Total_credits, Start_date, id*)

Faculty(fname, lname, id, supervisor)

Project(title, group_no, id*)

Parents(fname, lname, student_id*, Relation, Occupation)

Campus(Location, code*)

Results(student_id*, Percentage, Level)

Section 3: Data analysis tasks and SQL queries.

1. List the details of the students who's one of the parents is a Physicist.	select s.student_id,s.fname as StudentName, s.Program, p.fname as ParentName, p.relation, p.occupation from Students as s, Parents as p where s.student_id==p.child and p.occupation=='Physicist';
2. List the campus locations where the course 'Master of Statistics and Operations Research' is available.	select c.location, c.Program_code, p.name from Campus as c, Program as p where c.Program_code==p.code and p.name=='Master of Statistics and Operations Research';
3 List the details of the Faculties who are Program Coordinators and also a mentor of a group project.	select f.fname, f.lname, f.id , f.supervisor from Faculties as f where f.id in (select pt.faculty from Program as pm, Project as pt where pm.Coordinator==pt.faculty);
4. List the Details of the faculty who is the mentor of the group with highest number of students. Also list their Project Title and the number of Students in that Group.	select Faculties.fname, Faculties.lname, Faculties.id as FacultyID , Faculties.supervisor, Project.title as Project_Title, count(*) as NumStudents from Project , Students, Faculties where Project.group_no==Students.Project_gp_no and Faculties.id==Project.faculty group by Faculties.id order by count(*) desc limit 1;

5. Give the Program Details of the course with the max no. of enrolled students, also list the program coordinator.	select p.name, p.code, p.duration, p.Annual_fees, f.fname as Coordinator_Givename, f.lname as Coordinator_Familyname from Program as p, Faculties as f where p.Coordinator==f.id and p.code in (select Program from Students group by Program having count(*) in (select max(Enrollments) as MaxEnrollments from(select count(*) as Enrollments from Students group by Program)));
6. Give the Project Titles of the top 3 scorers.	select s.fname,s.lname,s.Project_gp_no,p.title,r.Percentage,r.Level from Students as s, Project as p,Results as r where s.Project_gp_no==p.group_no and r.StudentID==s.student_id and s.student_id in (select StudentID from Results order by Percentage desc limit 3) order by r.Percentage desc;
7. List Faculty details of Faculties who mentor more than one Project Group. Also list the Total number of students under them.	select Faculties.fname, Faculties.lname, Project.faculty, count(*) as Students from Students, Project, Faculties where Students.Project_gp_no==Project.group_no and Project.faculty in (select faculty from Project group by faculty having count(*)>1) and Project.faculty==Faculties.id group by Project.faculty order by count(*) desc;

8. List all the Female Students who go to Bendigo Airfield Campus and achieved a Distinction in their Respective program. Also List course Name and duration.	select Students.fname, Students.lname, Students.student_id, Program.name, Program.duration, Results.Percentage, Results.Level from Students, Results, Program, Campus where Students.student_id==Results.StudentID and Program.code==Students.Program and Campus.Program_code==Program.code and Students.sex=='F' and Campus.Location='Bendigo Airfield'
9. List project details of the Students who hail from the country that has the most number of HDs.	select Students.fname, Students.lname, Project.group_no, Project.title, Students.Nationality from Students, Project where Students.Project_gp_no==Project.group_no and Students.Nationality in (select Students.Nationality from Students, Results where Students.student_id==Results.StudentID and Results.Level=='HD' group by Students.Nationality order by count(*) desc limit 1);
10. Analyse the Grades of the British Students enrolled in a Masters Program and who's Mothers are employed i.e. not a Home-maker. Generate a table and also list their Project titles, Program Name and Program Coordinator.	select Students.fname as StudentName, Parents.fname as Mother_Name, Parents.occupation, Project.title as Project_title, Program.name as Program_name, Faculties.fname as Coordinator, Results.Percentage, Results.Level from Students, Parents, Project, Program, Faculties, Results where Students.student_id==Parents.child and Project.group_no==Students.Project_gp_no and Program.code==Students.Program and Faculties.id==Program.Coordinator and Results.StudentID==Students.student_id and Parents.Relation=='Mother' and Parents.Occupation != 'Home Maker' and Students.Nationality=='British' and Program.name like 'Master %';

Section 4: Discussions.

The Database contains the data that revolve around the students from a University. The database could be enhanced by adding some reasonable entities in the future so that it permits meaningful queries.

Initially I used CSV files to import the data into the database. I used data that I fabricated to populate into the database, using the data from web would have required some cleaning and subsetting prior to usage, so I decided not to use them and found my own data a bit more convenient.

Some challenges were faced in making sure the database is substantial for the analysis. The task to correctly model the ER Diagram, to abide integrity constraints and to correctly assess the primary and foreign keys of the relations and map it to the relational database schema, these were some another major challenges that I faced.

APPENDIX

```
CREATE TABLE Faculties (
fname          VARCHAR (12),
lname          VARCHAR (12),
id             INTEGER NOT NULL,
supervisor     INTEGER,
PRIMARY KEY (id));
```

#	fname	lname	id	supervisor
1	Bret	Lee	16449	70741
2	Carla	Stone	39572	95372
3	Michael	Clarke	43777	73947
4	Shane	Watson	47307	72978
5	David	Warner	70741	73947
6	Andrew	Symonds	71479	72978
7	Pat	Cummins	72978	70741
8	Mitchel	Starc	73947	

```
CREATE TABLE Campus (
Location        VARCHAR (21),
Program_code VARCHAR (12),
PRIMARY KEY (Location,Program_code),
FOREIGN KEY (Program_code)
REFERENCES Program (code)
);
```

#	Location	Program_code
1	Brunswick Campus	MC242
2	Bundoora Campus	MC271
3	City Campus	MC267
4	Bendigo Airfield	MC004P12
5	Bundoora Campus	MC004P12
6	Hamilton	MC004P12
7	City Campus	MC271
8	Brunswick Campus	MC267
9	City Campus	BC541

```
CREATE TABLE Project (
title           VARCHAR (37),
group_no        CHAR (2),
faculty         INTEGER,
PRIMARY KEY (group_no),
FOREIGN KEY (faculty)
REFERENCES Faculties (id)
);
```

#	title	group_no	faculty
1	Fake News Detection	C3	47307
2	Road Lane Line Detection	B2	93572
3	Sentiment Analysis	D4	71479
4	Forest Fire Prediction	A1	93572
5	Color Detection with Python	E5	72978
6	Weather Prediction	C1	47307
7	Keyword generation for google ads	B1	16449
8	Traffic Signs Recognition	D1	85001
9	Wine Quality Analysis	A5	84900
10	Stock Market Prediction	E3	70741

```
CREATE TABLE Program (
name            VARCHAR (37),
code            VARCHAR (12),
duration        CHAR (7),
Annual_fees     INTEGER,
Total_credits   INTEGER,
Start_date      DATE,
Coordinator     INTEGER,
PRIMARY KEY (code),
FOREIGN KEY (Coordinator) REFERENCES Faculties (id)
);
```

#	name	code	duration	Annual_fees	Total_credits	Start_date	Coordinator
1	Master of Analytics	MC242	2 Years	33600	180	01/03/2021	16449
2	Master of Artificial Intelligence	MC271	2 Years	36480	150	15/03/2021	70741
3	Master of Data Science	MC267	2 Years	36480	200	22/02/2021	73947
4	Master of Statistics and Operations Research	MC004P12	2 Years	33600	190	07/03/2021	72978
5	Master of Aerospace Engineering	MC256	1 Year	21000	80	27/06/2021	47307
6	Master of Material Sciences	MC771	1 Year	23000	100	07/03/2021	73947
7	Bachelor of Computer Science	BC541	4 Years	53600	280	03/03/2021	84900
8	Master of Engineering Management	MC0042	2 Years	36480	150	17/04/2021	85001
9	Bachelor of Electronics Engineering	BC879	4 Years	56480	300	21/02/2021	93572
10	Bachelor of Mechanical Engineering	BC789	4 Years	53600	290	08/07/2021	43777

```
CREATE TABLE Results (
StudentID       INTEGER,
Percentage      REAL,
Level           VARCHAR (6),
PRIMARY KEY (StudentID),
FOREIGN KEY (StudentID)
REFERENCES students (student_id)
);
```

#	StudentID	Percentage	Level
1	1315643	79.5	DI
2	1448987	82.3	DI
3	1890259	62	PASS
4	2135461	89	HD
5	2892938	87	HD
6	3025443	65	CREDIT
7	3214431	76	DI
8	4103785	74.5	CREDIT
9	4210569	78.5	DI
10	5232190	88.2	HD

```
CREATE TABLE Students (
fname           VARCHAR (12),
lname           VARCHAR (14),
```

```
student_id      INTEGER,
DOB             DATE,
sex             CHAR (1),
Project_gp_no CHAR (2),
Program         VARCHAR (12),
Nationality     VARCHAR (21),
PRIMARY KEY (student_id),
FOREIGN KEY (Project_gp_no)
REFERENCES Project (group_no),
FOREIGN KEY (Program)
REFERENCES Program (code) );
```

#	fname	lname	student_id	DOB	sex	Project_gp_no	Program	Nationality
1	Robin	Stinson	1315643	17/05/1996	F	D1	MC0042	Canadian
2	Sakura	Haruno	1321513	13/03/1999	F	C1	MC771	Japan
3	Hinata	Hyuga	1448987	05/09/2001	F	B1	BC879	Japan
4	Amy	Fowler	1548242	12/12/1994	F	A1	MC004P12	Netherlands
5	John	Greene	1890259	26/11/1996	M	C3	MC242	Australian
6	Steve	Rogers	2131511	27/03/2002	M	B1	BC073	American
7	Tony	Banner	2135461	30/08/1997	M	A5	MC256	American
8	Colin	Stevens	2136547	18/02/1998	M	D1	MC771	Norwegian
9	Emliy	Tribbiani	2576996	05/05/1996	F	C3	MC271	American
10	Peter	Wolowitz	2892938	05/05/1996	M	D4	MC004P12	British

```
CREATE TABLE Parents (
fname           VARCHAR (12),
lname           VARCHAR (12),
child           INTEGER,
Relation        VARCHAR (12),
occupation      VARCHAR (21),
PRIMARY KEY (fname,child),
FOREIGN KEY (child)
REFERENCES students (student_id) );
```

#	fname	lname	child	Relation	occupation
1	Rachel	Greene	1890259	Mother	Fashion Designer
2	Ross	Geller	5232190	Father	Paleontologist
3	Phoebe	Buffay	4210569	Mother	Masseuse
4	Monica	Geller	5232190	Mother	Chef
5	Chandler	Bing	8384740	Father	Analyst
6	Joey	Tribbiani	2576996	Father	Actor
7	Jennifer	Simmons	7754631	Mother	Home Maker
8	Sheldon	Cooper	4103785	Father	Physicist
9	Howard	Wolowitz	2892938	Father	Engineer
10	Leonard	Hofstader	3025443	Father	Physicist