

Project 1 : Database for a Mars Habitat

SCICOMP201 Database Management

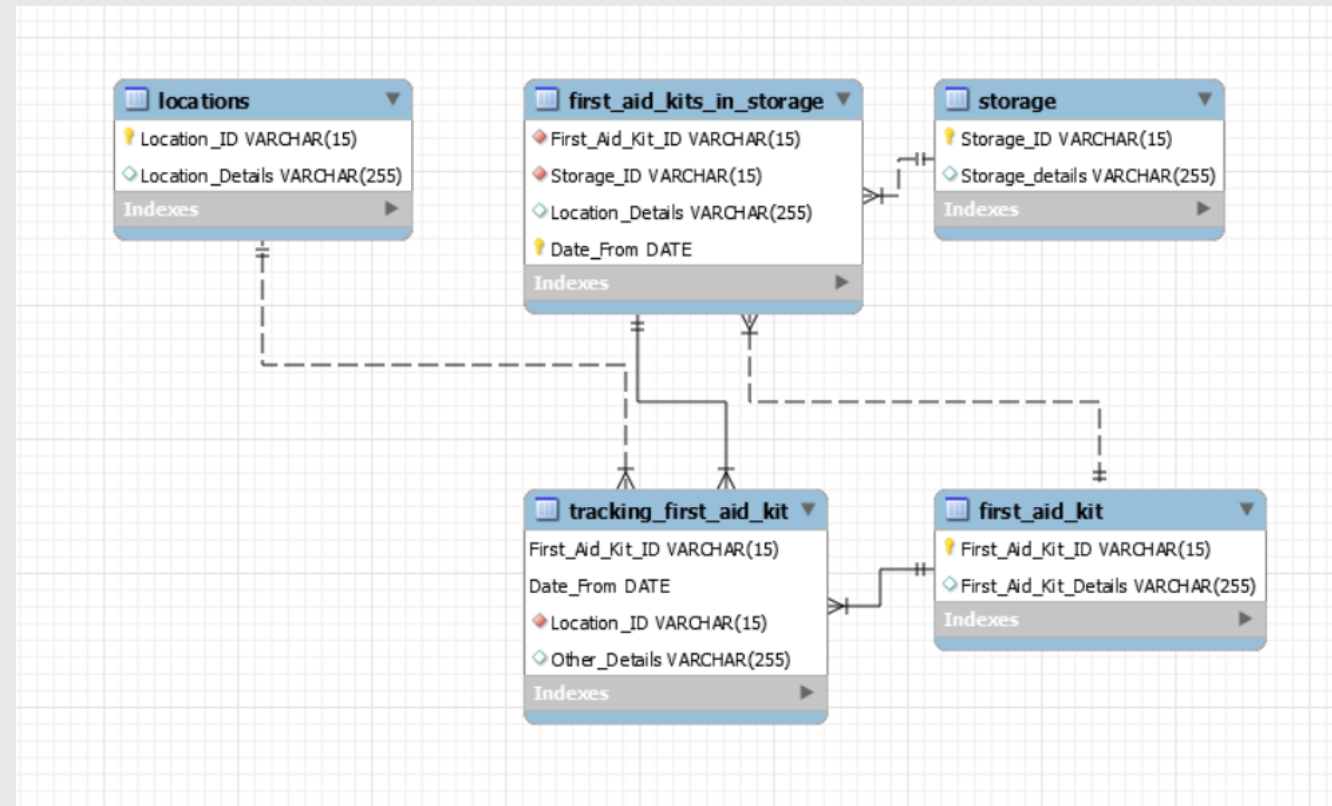
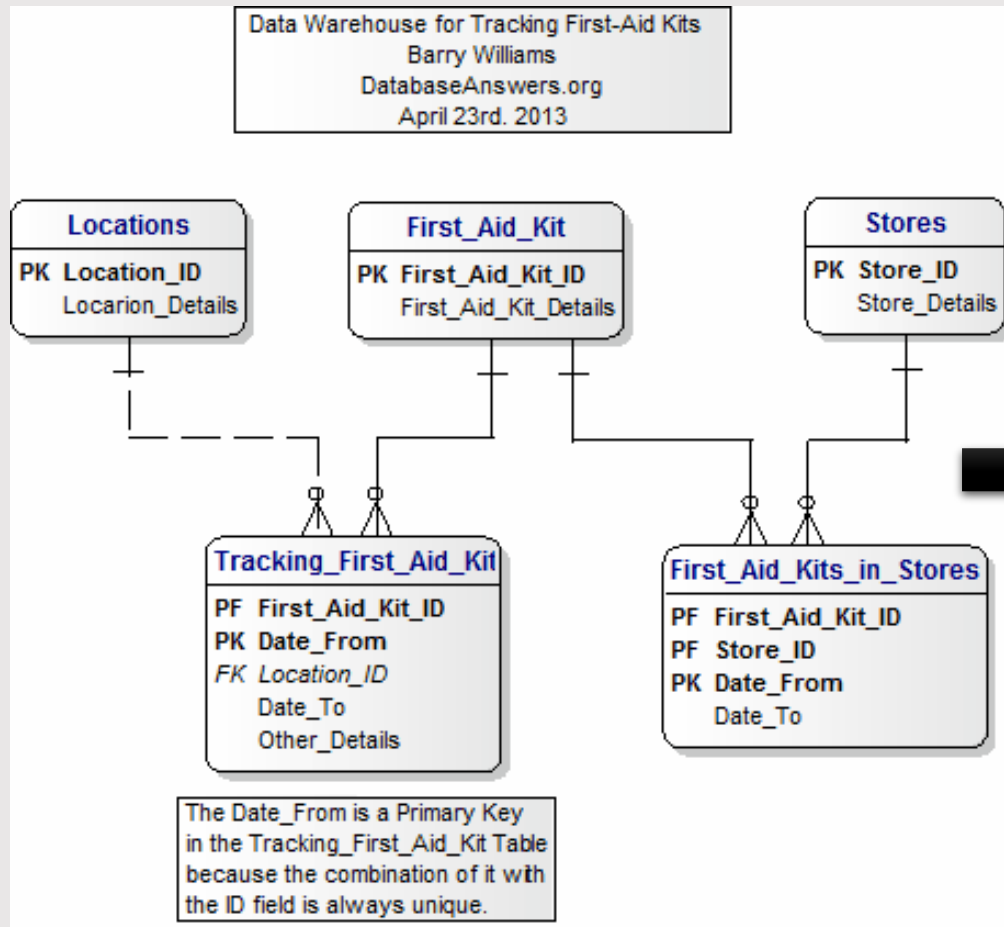
Irakli Bagratishvili

“This is all my own work. I have not knowingly allowed others to copy my work. This work has not been submitted in any other context.”

Database Schema

- Database Schema works as a blueprint to show us how the given database is constructed.
- Set of formulas (sentences) called integrity constraints imposed on a database

Schema 1 – Tracking First Aid Kits

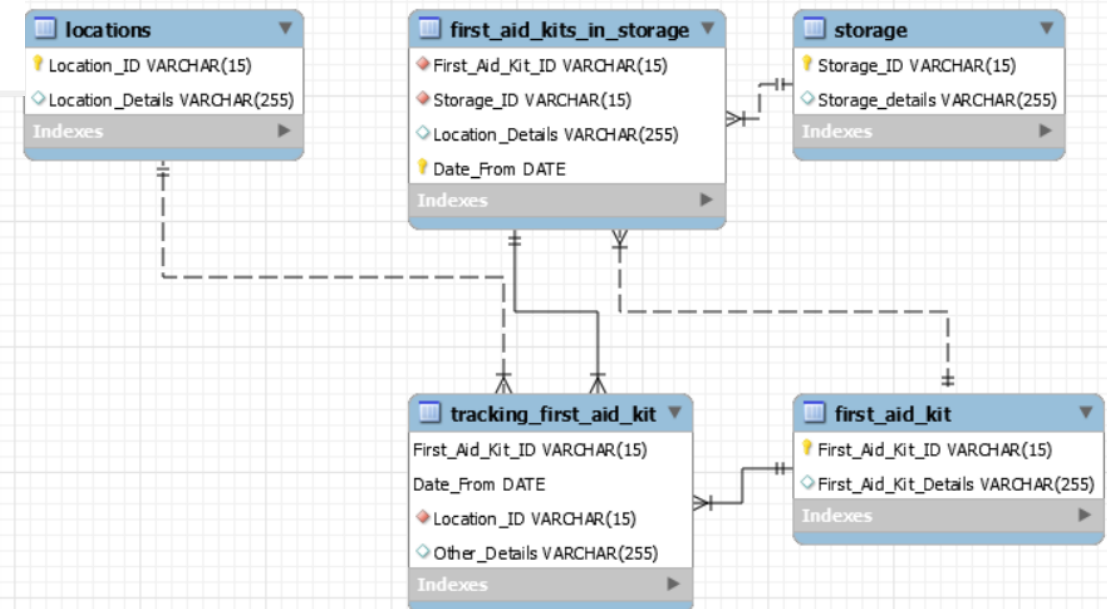


Purpose – Keeping track of the limited amount of First Aid Kits on Mars.

Schema 1 – Tracking First-Aid Kits

```
26
27 • CREATE TABLE Tracking_First_Aid_Kit(
28     First_Aid_Kit_ID VARCHAR(15) NOT NULL,
29     Date_From DATE NOT NULL,
30     Location_ID VARCHAR(15) NOT NULL,
31     Other_Details VARCHAR(255),
32     PRIMARY KEY (First_Aid_Kit_ID, Date_From),
33     FOREIGN KEY (Location_ID) REFERENCES Locations(Location_ID) ON DELETE CASCADE,
34     FOREIGN KEY (First_Aid_Kit_ID) REFERENCES First_Aid_Kit(First_Aid_Kit_ID) ON DELETE CASCADE,
35     FOREIGN KEY (Date_From) REFERENCES First_Aid_Kits_in_Storage(Date_From) ON DELETE CASCADE);
36
37
38
39
```

Using NOT NULL & ON DELETE CASCADE Constraints to make sure that entry deletion in one of the tables will not affect the integrity of the Primary table.



Provide snapshots of data in one or more tables. (Create at least 10 rows of made up data in each table.)

Location_ID	Location_Details
100	Kaiser Sea
101	Lockyer Land
102	Main Sea
103	Herschell II Strait
104	Dawes Continent
105	De La Rue Ocean
106	Lockyer Sea
107	Dawes Sea
108	Cassini Land
109	Secchi Continent

First_Aid_Kit_ID	First_Aid_Kit_Details
300	Kit A
301	Kit B
302	Kit C
303	Kit D
304	Kit B
305	Kit B
306	Kit A
307	Kit A
308	Kit C
309	Kit D
310	Kit B
311	Kit C
312	Kit A
313	Kit D
314	Kit C
315	Kit B
316	Kit B
317	Kit A

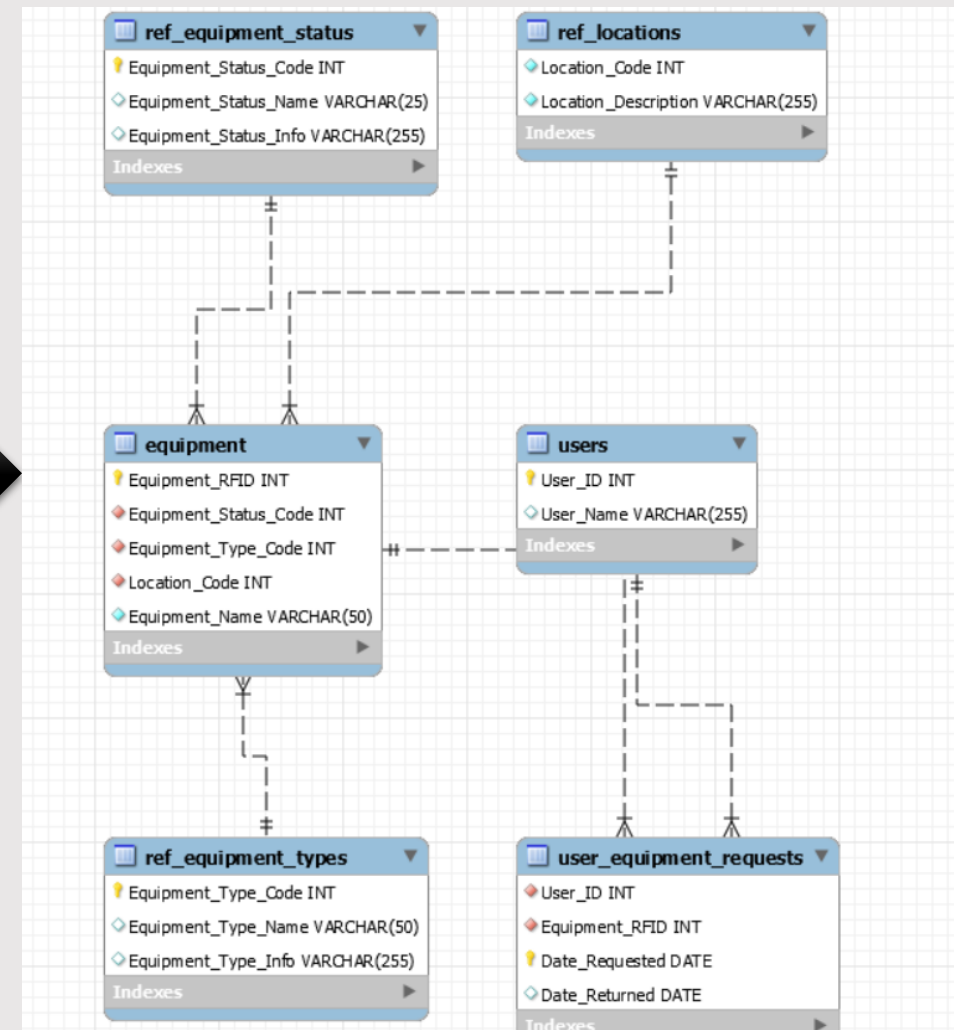
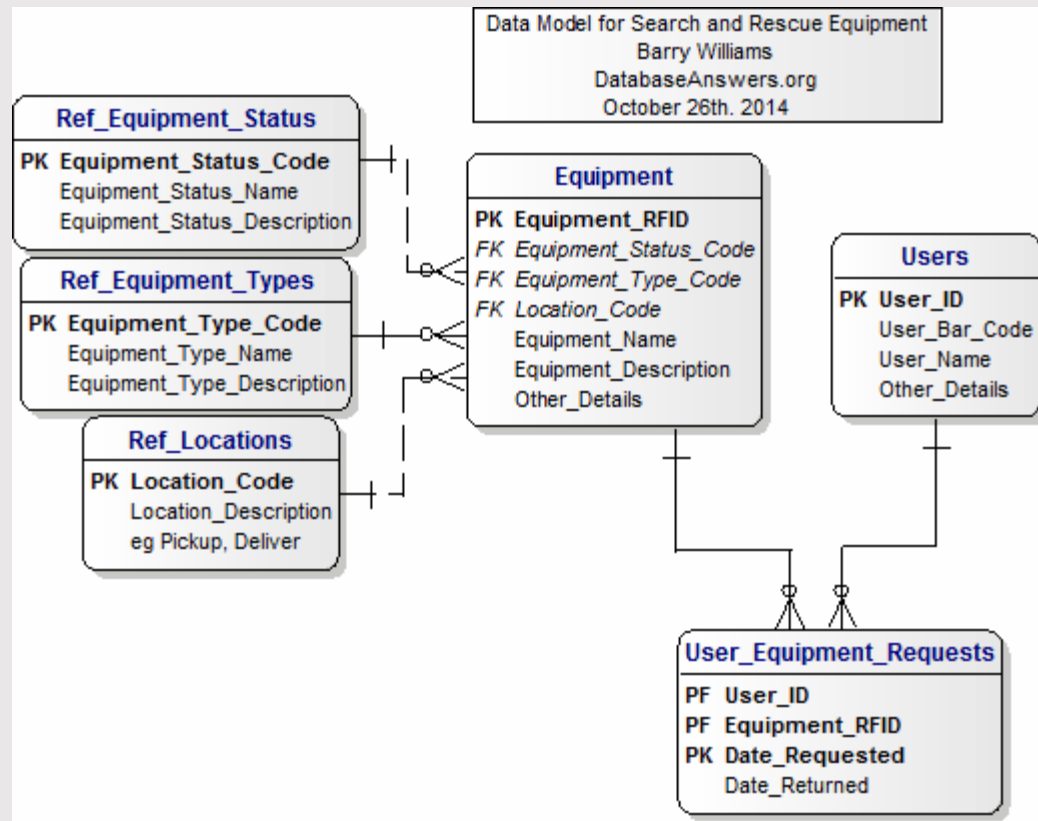
Storage_ID	Storage_details
10	Storage A
11	Storage B
12	Storage C
13	Storage D
14	Storage E
15	Storage F
16	Storage G
17	Storage H
18	Storage I
19	Storage J

Provide an example SQL query and a snapshot of the result. This query should be typical of a query made by someone actually living in the Mars Habitat. The query should be non-trivial.

First_Aid_Kit_Details	Location_details	First_Aid_Kit_ID
Kit B	Kaiser Sea	301
Kit A	Lockyer Land	300
Kit C	Lockyer Land	302
Kit D	Lockyer Land	309
Kit D	Main Sea	303
Kit A	Main Sea	306
Kit C	Herschell II Strait	308
Kit A	Dawes Continent	317
Kit B	Lockyer Sea	310
Kit A	Dawes Sea	312
Kit B	Cassini Land	315
Kit B	Secchi Continent	316

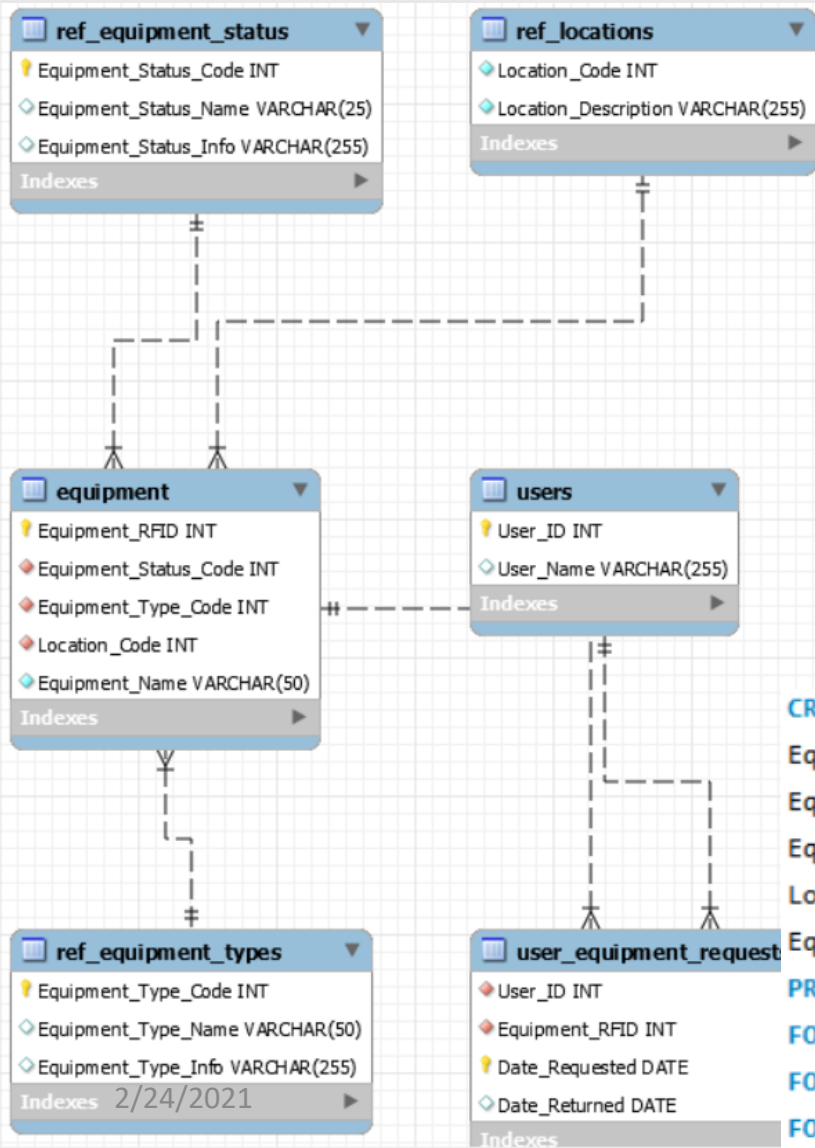
An Astronaut got injured while working on the excavation site in Main Sea. Based on the type of injuries, they will need the First Aid Kit 'A'. Our task is to check where the Kit As are currently.

Schema 2 – Search & Rescue Equipment Tracking



The goal was to adapt Search & Rescue Equipment Tracking Schema to Martian reality. Keeping control over equipment is important to survival when the resources are limited.

Schema 2 – Search and Rescue Equipment



Using NOT NULL & ON DELETE CASCADE Constraints to make sure that entry deletion in one of the tables will not affect the integrity of the Primary table.

```
CREATE TABLE Equipment(  
    Equipment_RFID INTEGER UNIQUE AUTO_INCREMENT,  
    Equipment_Status_Code INTEGER NOT NULL,  
    Equipment_Type_Code INTEGER NOT NULL,  
    Location_Code INTEGER NOT NULL,  
    Equipment_Name VARCHAR(50) NOT NULL,  
    PRIMARY KEY (Equipment_RFID),  
    FOREIGN KEY (Equipment_Status_Code) REFERENCES Ref_Equipment_Status(Equipment_Status_Code) ON DELETE CASCADE,  
    FOREIGN KEY (Equipment_Type_Code) REFERENCES Ref_Equipment_Types(Equipment_Type_Code) ON DELETE CASCADE,  
    FOREIGN KEY (Location_Code) REFERENCES Ref_Locations(Location_Code) ON DELETE CASCADE);
```


Schema 2 Data Tables

	Equipment_RFID	Equipment_Name	Equipment_Status_Code	Equipment_Status_Name	Location_Code
▶	1	Band-Aid	1	Available	1
	2	First-Aid Kit	1	Available	1
	3	Oxygen Tank	1	Available	1
	4	Aspirin	1	Available	1
	5	Foil Blanket	2	In Use	2
	6	CO2 Tank	1	Available	2
	7	Extinguisher	1	Available	3
	8	Excavator	1	Available	4
	9	Sample Robot	1	Available	5
	10	Hammer	2	In Use	6
	11	Screwdriver	1	Available	7
	12	Safety Manual	2	In Use	8
	13	Safety Box	1	Available	9
	14	Search Drone	2	In Use	1
	15	Maintenance Dr...	1	Available	1
	16	Sample Robot	1	Available	2
	17	DNA Robot	1	Available	4

	Location_Code	Location_Description
▶	1	Main Base
	2	Departure Point A
	3	Departure Point B
	4	Arrival Point C
	5	Arrival Point D
	6	Hangar
	7	Barracks
	8	Hospital
	9	Warehouse
	10	Office Area
	11	Lab 1
	12	Lab 2

	Equipment_Type_Code	Equipment_Type_Name	Equipment_Type_Info
▶	1	Medic	Medical
	2	Fire	Fire Hazards
	3	Ground	Tools For Ground Work
	4	Handyman	All-round tools
	5	Safety	Safety equipment
	6	Drones	AI Drones for search and rescue
	7	Robots	S & R Robots
	8	Flares	Flares for rescue
	9	Rope	Ropes
	10	Ascender	ascender for rock climbing

Schema 2 Query #1

User_ID	User_Name	Date_Requested	Equipment_Name
15	Marise Wijk van	2031-02-05	Maintenance Drone
6	Derin Guven	2031-03-05	CO2 Tank
7	Annelies Jong de	2031-03-12	Extinguisher
8	Maggie Maliszewski	2031-03-26	Excavator
13	Kalle Valkeakari	2031-05-02	Search Drone
2	Sebastiaan Bye	2031-05-11	First-Aid Kit
3	Thrisha Dev	2031-06-01	Oxygen Tank
10	India Nunes	2031-06-02	Hammer
12	Jan Stelmaszczyk	2031-06-12	Safety Box
4	Zoe El-Hage	2031-07-01	Aspirin
5	Darong Groot	2031-08-11	Foil Blanket
14	Sukhuma Wattana...	2031-10-04	Safety Manual
11	Elise Steenvoorden	2031-12-09	Screwdriver
9	Veerle Maslowski	2031-12-12	Sample Robot

```
CREATE TABLE User_Equipment_Requests(
  User_ID INTEGER UNIQUE NOT NULL,
  Equipment_RFID INTEGER NOT NULL,
  Date_Requested DATE,
  Date_Returned DATE DEFAULT NULL,
  PRIMARY KEY (Date_Requested),
  FOREIGN KEY (User_ID) REFERENCES Users(User_ID) ON DELETE CASCADE,
  FOREIGN KEY(Equipment_RFID) REFERENCES Equipment(Equipment_RFID) ON DELETE CASCADE);
```

/* Checking Which User Requested The Item.*/

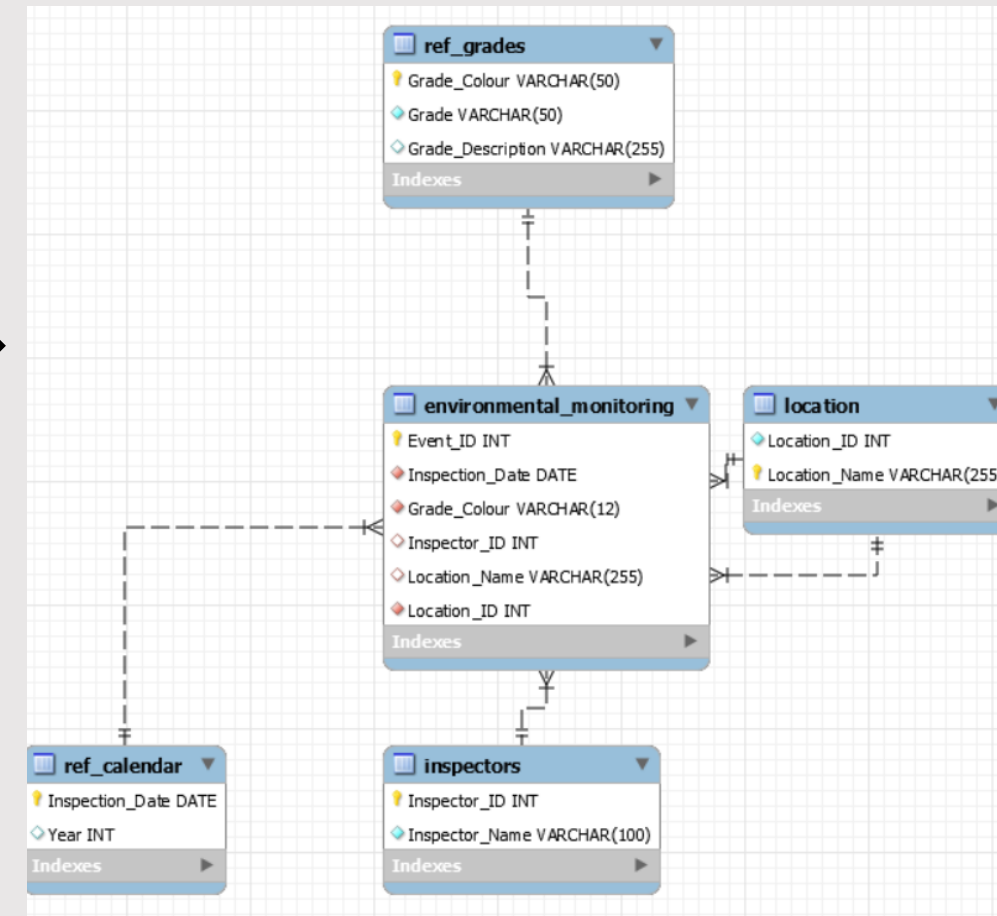
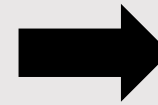
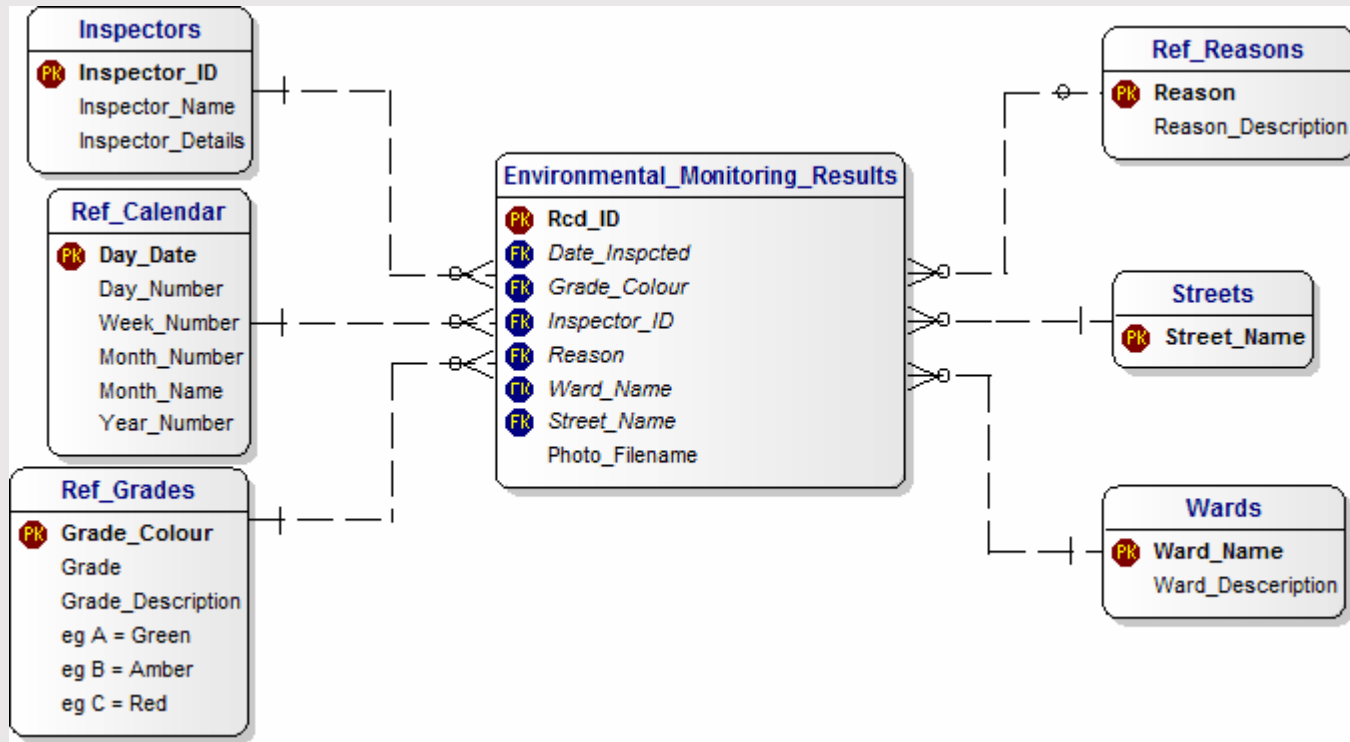
```
SELECT Users.User_ID, Users.User_Name, user_equipment_requests.Date_Requested, Equipment.Equipment_Name
FROM User_Equipment_Requests
LEFT JOIN Users ON Users.User_ID = user_equipment_requests.User_ID
LEFT JOIN Equipment ON Equipment.Equipment_RFID = User_equipment_Requests.Equipment_RFID;
```

Equipment_RFID	Equipment_Name	Equipment_Status_Code	Equipment_Status_Name	Location_Code
4	Aspirin	1	Available	1
5	Foil Blanket	2	In Use	2
6	CO2 Tank	1	Available	2
7	Extinguisher	1	Available	3
8	Excavator	1	Available	4
9	Sample Robot	1	Available	5
10	Hammer	2	In Use	6
11	Screwdriver	1	Available	7
12	Safety Manual	2	In Use	8
13	Safety Box	1	Available	9
14	Search Drone	2	In Use	1
15	Maintenance Dr...	1	Available	1
16	Sample Robot	1	Available	2
17	DNA Robot	1	Available	4
18	Hound	3	Damaged	3
19	Safety Rope	3	Damaged	7
20	Basic Ascender	4	Missing	8
21	Ultra Ascender	4	Missing	8

/* Checking Status of an item*/

```
SELECT Equipment.Equipment_RFID, Equipment.Equipment_Name, Ref_Equipment_Status.Equipment_Status_Code,
Ref_Equipment_Status.Equipment_Status_Name, Ref_Locations.Location_Code
FROM Equipment
LEFT JOIN Ref_Equipment_Status ON Ref_Equipment_Status.Equipment_Status_Code = Equipment.Equipment_Status_Code
LEFT JOIN Ref_Locations ON Ref_Locations.Location_Code = Equipment.Location_Code;
```

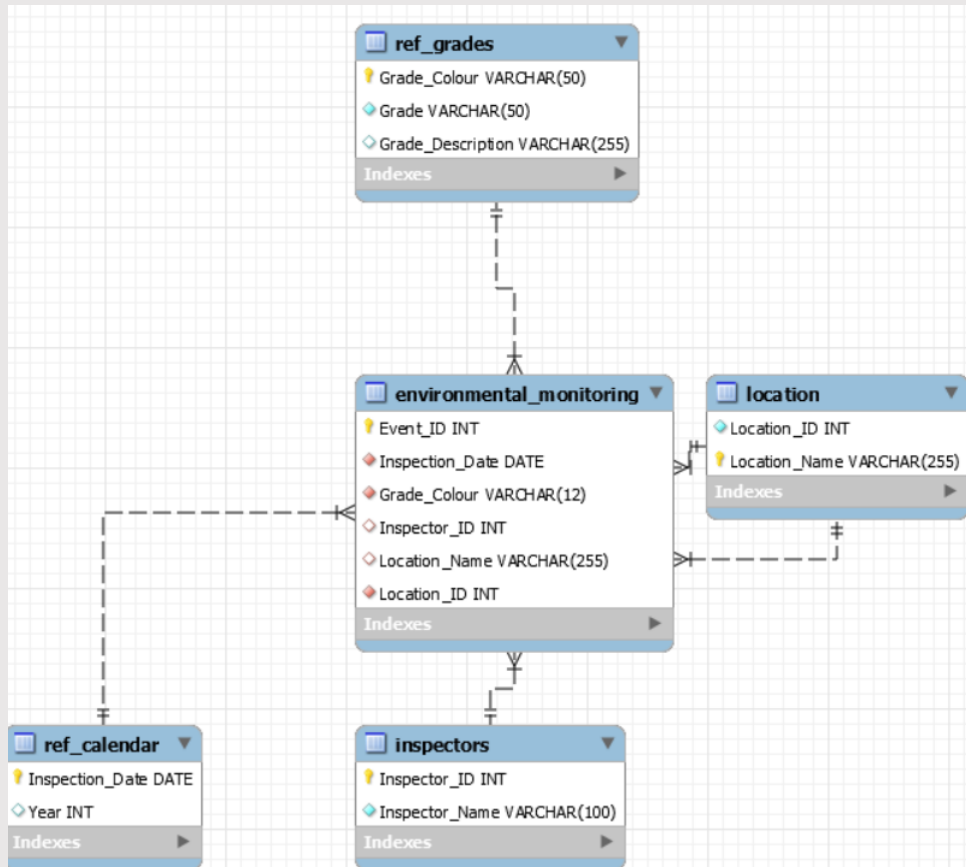
Schema 3 – Environmental Monitoring



Strong winds are part of everyday life on Mars. It is important to keep track of the speed of wind for safety reasons. The schema has been adapted to monitor wind categories on Mars habitat.

Schema 3 – Environmental Monitoring

Using NOT NULL & ON DELETE CASCADE Constraints to make sure that entry deletion in one of the tables will not affect the integrity of the Primary table.



```
CREATE TABLE Environmental_Monitoring(  
    Event_ID INTEGER UNIQUE AUTO_INCREMENT,  
    Inspection_Date DATE NOT NULL,  
    Grade_Colour VARCHAR(12) NOT NULL,  
    Inspector_ID INTEGER,  
    Location_Name VARCHAR(255),  
    Location_ID INTEGER NOT NULL,  
    Primary KEY(Event_ID),  
    FOREIGN KEY(Inspection_Date) REFERENCES Ref_Calendar(Inspection_Date) ON DELETE CASCADE,  
    FOREIGN KEY(Grade_Colour) REFERENCES Ref_Grades(Grade_Colour) ON DELETE CASCADE,  
    FOREIGN KEY(Inspector_ID) REFERENCES Inspectors(Inspector_ID) ON DELETE CASCADE,  
    FOREIGN KEY(Location_Name) REFERENCES Location(Location_Name) ON DELETE CASCADE,  
    FOREIGN KEY(Location_ID) REFERENCES Location(Location_ID) ON DELETE CASCADE);
```

Schema 3 Data Tables

Grade_Colour	Grade	Grade_Description
Blue	D	D = Blue
Green	A	A = Green
Orange	B	B = Orange
Red	C	C = Red

Location_ID	Location_Name
1	Kaiser Sea
2	Lockyer Land
3	Main Sea
4	Herschell II Strait
5	Dawes Continent
6	De La Rue Ocean
7	Lockyer Sea
8	Dawes Sea
9	Cassini Land
10	Secchi Continent

Inspection_Date
2031-03-04
2031-03-11
2031-07-01
2031-08-04
2031-08-11
2031-10-10
2031-11-01
2031-11-03
2031-12-06
2032-01-01
2032-03-01
2033-05-05

Inspector_ID	Inspector_Name
1	Andrew Brooks
2	Irakli Bagratishvili
3	Sebastiaan Bye
4	Thrisha Dev
5	Zoe El-Hage
6	Darong Groot
7	Derin Guven
8	Annelies Jong de
9	Maggie Maliszewski
10	Veerle Maslowski
11	India Nunes
12	Elise Steenvoorden
13	Jan Stelmaszczyk
14	Kalle Valkeakari
15	Sukhuma Wattana...
16	Marise Wijk van

Schema 3 Query

Event_ID	Inspection_Date	Grade_Colour	Inspector_Name	Inspector_ID
4	2031-03-04	Orange	Thrisha Dev	4
2	2031-03-11	Green	Andrew Brooks	1
7	2031-07-01	Green	Elise Steenvoorden	12
12	2031-08-04	Green	Darong Groot	6
9	2031-08-11	Green	Annelies Jong de	8
10	2031-10-10	Blue	India Nunes	11
5	2031-11-01	Red	Irakli Bagratishvili	2
11	2031-11-03	Green	Jan Stelmaszczyk	13
6	2031-12-06	Orange	Maggie Maliszewski	9
1	2032-01-01	Green	Andrew Brooks	1
8	2032-03-01	Orange	Derin Guven	7
3	2033-05-05	Green	Sebastiaan Bye	3

```
CREATE TABLE Environmental_Monitoring(
Event_ID INTEGER UNIQUE AUTO_INCREMENT,
Inspection_Date DATE NOT NULL,
Grade_Colour VARCHAR(12) NOT NULL,
Inspector_ID INTEGER NOT NULL,
Location_Name VARCHAR(255),
Location_ID INTEGER NOT NULL,
Primary KEY(Event_ID),
FOREIGN KEY(Inspection_Date) REFERENCES Ref_Calendar(Inspection_Date) ON DELETE CASCADE,
FOREIGN KEY(Grade_Colour) REFERENCES Ref_Grades(Grade_Colour) ON DELETE CASCADE,
FOREIGN KEY(Inspector_ID) REFERENCES Inspectors(Inspector_ID) ON DELETE CASCADE,
FOREIGN KEY(Location_Name) REFERENCES Location(Location_Name) ON DELETE CASCADE,
FOREIGN KEY(Location_ID) REFERENCES Location(Location_ID) ON DELETE CASCADE);
```

```
SELECT Environmental_Monitoring.Event_ID, Ref_Calendar.Inspection_Date, Ref_grades.Grade_Colour, Inspectors.Inspector_Name,
Inspectors.Inspector_ID
FROM Environmental_Monitoring
LEFT JOIN Ref_Calendar ON Ref_Calendar.Inspection_Date = Environmental_Monitoring.Inspection_Date
LEFT JOIN Ref_Grades ON Ref_grades.Grade_Colour = Environmental_Monitoring.Grade_Colour
LEFT JOIN Location ON Location.Location_Name = Environmental_Monitoring.Location_Name
LEFT JOIN Inspectors ON Inspectors.Inspector_ID = Environmental_Monitoring.Inspector_ID
ORDER BY inspection_date;
```

Evaluation of Schema 1 – Tracking First-Aid Kits

- 1. 4 FKs in Schema 1
- 2. How deep is the referential tree?
- Trace foreign key connections between tables (not counting cycles).
- A high value for depth of the referential means the schema is more complicated to work with.
- 3. No tables without PK
- 4. No attributes have a short name
- 5. No table has a short name
- 6. Attribute naming is inconsistent with a mix of styles in use.
- 7. Table naming is inconsistent with a mix of styles in use?
- 8. Less than 80% Nullable Rate
- 9. No tables with more than 40 attributes
- 10. One table (first_aid_kits_in_storage) has a composite PK

Evaluation of Schema 2 – Search & Rescue Equipment

- 1. 4 FKs in Schema 1
- 2. How deep is the referential tree?
- Trace foreign key connections between tables (not counting cycles).
- A high value for depth of the referential means the schema is more complicated to work with.
- 3. No tables without PK
- 4. No attributes have a short name
- 5. No table has a short name
- 6. Attribute naming is inconsistent with a mix of styles in use.
- 7. Table naming is inconsistent with a mix of styles in use?
- 8. Less than 80% Nullable Rate
- 9. No tables with more than 40 attributes
- 10. One table (user_equipment_requests) has a composite PK (PF)

Evaluation of Schema 3 – Environmental Monitoring

- 1. 5 FKs in Schema 1
- 2. Multiple Sub-trees
- 3. No tables without PK
- 4. No attributes have a short name
- 5. No table has a short name
- 6. Attribute naming is inconsistent with a mix of styles in use.
- 7. Table naming is inconsistent with a mix of styles in use
- 8. Less than 80% Nullable Rate
- 9. No tables with more than 40 attributes
- 10. No table with composite PK.

Conclusion

- Schemas used and developed in this project are quite simple but they can be further improved by adding more integrity constraints.
- It is possible to write several sample queries that can automate important tasks to simplify life on Mars.

Thank you for attention!