/\*Tables\*/

CREATE TABLE Projects (

projectID INTEGER PRIMARY KEY,

location TEXT,

name TEXT,

description TEXT DEFAULT 'General construction work',

startDate TEXT,

endDate TEXT

);

CREATE TABLE Subprojects (

subprojectID INTEGER PRIMARY KEY,

projectID TEXT REFERENCES Projects (projectID),

startDate TEXT,

endDate TEXT

);

CREATE TABLE Dependency (

comesAfterID INTEGER PRIMARY KEY REFERENCES Subprojects (subprojectID),

subprojectID INTEGER REFERENCES Subprojects (subprojectID)

);

CREATE TABLE Employees (

employeeID INTEGER PRIMARY KEY,

name TEXT NOT NULL

);

CREATE TABLE WorkingOn (

employeeID INTEGER REFERENCES Employees (employeeID),

subprojectID INTEGER REFERENCES Subprojects (subprojectID),

asQualification TEXT REFERENCES Qualifications(qualification),

PRIMARY KEY (employeeID, subprojectID)

);

CREATE TABLE EmployeeAbsences (

employeeID INTEGER REFERENCES Employees (employeeID),

startDate TEXT NOT NULL,

endDate TEXT,

absenceType TEXT NOT NULL DEFAULT 'Sick leave',

substituteID INTEGER REFERENCES Employees (employeeID), /\*Can be null\*/

PRIMARY KEY (employeeID, startDate)

);

CREATE TABLE Qualifications (

qualification TEXT PRIMARY KEY

);

CREATE TABLE QualifiedFor (

qualification TEXT REFERENCES Qualifications (qualification),

employeeID INTEGER REFERENCES Employees (employeeID),

PRIMARY KEY (qualification, employeeID)

);

CREATE TABLE NeededFor(

amount INTEGER,

qualification TEXT REFERENCES Qualifications (qualification),

subprojectID INTEGER REFERENCES Subprojects (subprojectID),

PRIMARY KEY (qualification, subprojectID)

);

CREATE TABLE Models (

modelName TEXT,

manufacturer TEXT,

description TEXT DEFAULT 'Truck',

size REAL CHECK (size > 0.0),

consumption REAL CHECK (size > 0.0),

PRIMARY KEY (modelName, manufacturer)

);

CREATE TABLE Machines (

serialNo INTEGER PRIMARY KEY ,

modelName TEXT,

manufacturer TEXT DEFAULT 'Caterpillar',

FOREIGN KEY (modelName, manufacturer) REFERENCES Models (modelName, manufacturer)

);

CREATE TABLE MachinesUnavailable (

serialNo INTEGER REFERENCES Machines (serialNo),

startDate TEXT,

endDate TEXT,

PRIMARY KEY (serialNo, startDate)

);

CREATE TABLE MachinesNeeded (

model TEXT,

manufacturer TEXT,

subprojectID INTEGER REFERENCES Subprojects (subprojectID),

neededNo INTEGER ,

assigned INTEGER DEFAULT 0 CHECK (assigned IN (0,1)), /\*1 is true, 0 is false\*/

startDate TEXT,

endDate TEXT,

PRIMARY KEY (model, manufacturer, subprojectID, neededNo)

FOREIGN KEY (model, manufacturer) REFERENCES Models (modelName, manufacturer)

);

CREATE TABLE MachinesAssigned (

serialNo INTEGER REFERENCES Machines (serialNo),

startDate TEXT,

endDate TEXT,

subprojectID INTEGER REFERENCES Subprojects (subprojectID),

PRIMARY KEY (serialNo, startDate)

);

/\*Indexes\*/

/\*For finding subprojects by date\*/

CREATE INDEX SubprojectDates ON Subprojects(startDate, endDate);

/\*For finding machines by their name\*/

CREATE INDEX ownedMachines ON Machines(ModelName);

/\*Views\*/

/\*All ongoing and future subProjects (date(‘now’) is today’s date) \*/

CREATE VIEW ongoingSubprojects AS

SELECT \*

FROM Subprojects

WHERE endDate > date('now') OR endDate is null;

/\*All ongoing and future projects\*/

CREATE VIEW ongoingProjects AS

SELECT \*

FROM Projects

WHERE NOT endDate > date('now') OR endDate is null;

/\*Use cases (start from the top and progress downwards)\*/

INSERT INTO Employees VALUES(12345, 'Pekka');

INSERT INTO Projects(projectID) VALUES(747);

INSERT INTO Subprojects(subprojectID, projectID) VALUES(999, 746); /\*Fails, because of references constraint. There is no 746 projectID in the projects relation\*/

INSERT INTO Subprojects(subprojectID, projectID) VALUES(999, 747);

INSERT INTO WorkingOn(employeeID, subprojectID) VALUES(12345, 999); /\*asQualification is null\*/

/\*Find employees working on a specific subproject\*/

SELECT \* FROM workingOn WHERE SubprojectID=999;

/\*Find ongoing and future subprojects (only the subprojects were a startDate has been assigned) \*/

SELECT \* FROM subprojects WHERE startDate > date('now'); /\*Nothing appears\*/

UPDATE Subprojects SET startDate=date('now') WHERE subprojectID=999;

SELECT \* FROM subprojects WHERE startDate >= date('now'); /\*Now the project appears\*/

UPDATE Projects SET startDate='2000-02-02' WHERE projectID=747;

/\*Find all past or older long-lasting projects and count them\*/

SELECT \*, Count(\*) FROM Projects WHERE startDate < '2005-1-1';

INSERT INTO Qualifications Values('fisher');

INSERT INTO Qualifications Values('demolition man');

INSERT INTO QualifiedFor Values('fisher',12345);

/\*This should return qualifications which none of our employees have.\*/

SELECT qualification FROM Qualifications WHERE qualification NOT IN (SELECT DISTINCT qualification FROM QualifiedFor);

/\*In case the previous doesn't work. This should work.\*/

SELECT qualification FROM Qualifications

EXCEPT

SELECT DISTINCT qualification FROM QualifiedFor;

INSERT INTO EmployeeAbsences(employeeID, startDate, absenceType) VALUES(12345, date('now'), 'Broken leg');

/\*For checking when a person has been and is going to be absent. A date could be chosen as well\*/

SELECT \* FROM Employees, EmployeeAbsences WHERE Employees.employeeID=EmployeeAbsences.employeeID AND Employees.employeeID='12345';

INSERT INTO Models VALUES('Pickup Truck 6K', 'Caterpillar', 'Pickup Truck',10,20);

INSERT INTO Machines(serialNo, modelName, manufacturer) VALUES (545, 'Pickup Truck 6K', 'Caterpillar');

/\*For checking which machines are available in a certain date range or on a certain date. E.g. 1.1.2021. A machine can be assigned to a project base on this information and MachinesNeeded can afterwards be modified accordingly\*/

SELECT \*

FROM Machines

WHERE serialNo IN (SELECT serialNo FROM Machines

EXCEPT

SELECT serialNo FROM MachinesUnavailable WHERE startDate<= '2021-01-01' AND endDate>= '2021-01-01'

EXCEPT

SELECT serialNo FROM MachinesAssigned WHERE startDate<= '2021-01-01' AND endDate>= '2021-01-01');

/\*Average consumption of our vehicles\*/

SELECT Avg(consumption) FROM Machines, Models WHERE Machines.manufacturer=Models.manufacturer AND Machines.modelName=Models.modelName;

/\*All vehicles that are small enough\*/

SELECT \* FROM Machines, Models WHERE Machines.manufacturer=Models.manufacturer AND Machines.modelName=Models.modelName AND Models.size<=10;

INSERT INTO Models VALUES('Road Roller 6K', 'Caterpillar', 'Road roller',10,20);

INSERT INTO Machines(serialNo, modelName, manufacturer) VALUES (565, 'Road Roller 6K', 'Caterpillar');

/\*All vehicles that are road rollers. Collate nocase ignores the case of the words (upper/lowercase)\*/

SELECT \* FROM Machines, Models WHERE Machines.manufacturer=Models.manufacturer AND Machines.modelName=Models.modelName AND Models.description='Road roller' COLLATE NOCASE;

/\*Find all names and qualifications of the employees\*/

SELECT \* FROM Employees, QualifiedFor WHERE Employees.employeeID=QualifiedFor.employeeID;

/\*Find the start and end dates of the projects which a worker is working on \*/

SELECT Employees.employeeID, WorkingOn.subprojectID, name, startDate, endDate FROM Employees, WorkingOn, Subprojects WHERE Employees.employeeID=WorkingOn.employeeID AND WorkingOn.subprojectID=Subprojects.subprojectID AND Employees.employeeID=12345;

/\*Find available employees in a certain date range\*/

SELECT \* FROM Employees WHERE employeeID IN

(SELECT employeeID FROM Employees

EXCEPT

SELECT employeeID FROM EmployeeAbsences WHERE startDate<'2.2.2021' AND endDate>'22.2.2021'

EXCEPT

SELECT WorkingOn.employeeID FROM WorkingOn, Subprojects WHERE WorkingOn.subprojectID=Subprojects.subprojectID AND startDate<'2.2.2021' AND endDate>'22.2.2021');

/\*Find all employees contributing to a project\*/

SELECT employeeID FROM WorkingOn, Subprojects, Projects WHERE WorkingOn.subprojectID=Subprojects.subprojectID AND Subprojects.projectID=Projects.projectID AND Projects.projectID=747;

/\*Find future or ongoing subprojects that need fishers\*/

SELECT DISTINCT S.subprojectID

FROM Subprojects AS S, NeededFor

WHERE S.subprojectID = NeededFor.subprojectID AND NeededFor.qualification = 'fisher' AND S.endDate >= date('now');

/\*Find all employees (id and name) that are substitutes right now\*/

SELECT Distinct Employees.employeeID, Employees.name

FROM Employees, EmployeeAbsences

WHERE Employees.employeeID = EmployeeAbsences.substituteID AND EmployeeAbsences.startDate <= date('now') AND EmployeeAbsences.endDate >= date('now');

INSERT INTO Employees

Values (5543, 'Tarja Niinistö');

UPDATE Employees

SET name = 'Tarja Kekkonen'

WHERE EmployeeID = 5543;

DELETE FROM Employees

WHERE EmployeeID = 5543;