CSC423

FINAL PROJECT

Busy-Bee Cleaning Company

Christian Allen

12/3/2020

a.        Identify the relations or entity types.

* Client (**client number**, first name, last name, address, and telephone number)
* Employee (**staff number**, first name, last name, address, salary, and telephone number)
* Equipment (**equipment identifier**, description, usage, cost)
* Cleaning service (**client service number**, client name, start date, end date,starting hour, ending hour)

b.        Identify the relationship types, as well as their participation and cardinality constraints.

* Client receives cleaning service (One-to-many)
* Cleaning service requires Equipment (One-to-one)

Employee performs Cleaning service(One-to-many)

Diagram

Description automatically generated

**Relationship Types:**

Client Receives cleaning service

Cleaning service requires equipment

Employee performs cleaning service

**Cardinality Constraints:**

* **Client Receives Cleaning Service**
  + A client receives cleaning services (1..\*)
  + A cleaning service is required by one client (1..1)
* **Cleaning Service requires Equipment**
  + A cleaning service requires the type and quantity of special  equipment used (1..\*)
  + If needed, a special equipment is required for services (0..\*)
* **Employees performs Cleaning Service**
  + An Employee performs multiple cleaning services (1..\*)
  + A service is performed by multiple employees (1..\*)

c.    Identify the attributes associated to the previous entity or relationship types.

* Client receives cleaning service
* Client (**client number**, first name, last name, address, and telephone number)
* Cleaning Service (**client service number**, client name, start date, end date,starting hour, ending hour)
* The two attributes previously associated are client number & client identifier
* Cleaning service requires Equipment
  + Cleaning service (**client service number**, start date, end date,starting hour, ending hour)
  + Equipment (**equipment service number**, description, usage, cost)
  + The two attributes previously associated are client identifier & equipment identifier
* Employee performs for  Cleaning service
  + Employee (**staff number**, first name, last name, address, salary, and telephone number)
  + Cleaning Service (**client service number**, client name, start date, end date,starting hour, ending hour)
  + The two attributes previously associated are staff number & client identifier

  Determine candidate and primary key attributes of entity types.

* Client (**client number**, first name, last name, address, and telephone number)
  + Primary key: client number
  + Candidate key: address, client number, telephone number
* Employee (**staff number**, first name, last name, address, salary, and telephone number)
  + Primary key: staff number
  + Candidate key: address, staff number, telephone number
* Equipment (**equipment identifier**, description, usage, cost)
  + Primary key: equipment identifier
  + Candidate key: equipment identifier, description, usage
* Cleaning service (**client service number**, client name, start date, end date, starting hour, ending hour)
  + Primary key: client identifier
  + Candidate key: client identifier, number of hours

Derive relations from the conceptual model.

* Client receives cleaning service
* Cleaning Service requires Equipment
* Employee performs  Cleaning Service

In the event of adjacent one-to-many relationships or zero-to-many relationships we will draw tables for the relations involved. The relations will have the parent keys of the adjoining tables.

Diagram

Description automatically generated

b.        Validate the logical model using normalization to 3NF.

The tables have been created in the conceptual model, convert it into 1NF:

* UNF: **Main Table**(**clientNum**, cFName, cLName, cAddress, cTeleNum, **staffNum** , eFName, eLName, eAddress, Salary, eTeleNum, **eqIdent**, descript, usage, cost, **clservNum**, clservName, startDate, endDate, startHour, endHour)
* 1NF:

Client(**cNum**, cFName, cLName, cAddress, cTeleNum)

Employee(**staffNum**, eFName, eLName, eAddress, salary, eTeleNum)

Equipment(**eqIdent**, descript, usage, cost)

Cleaning service(**clservNum**, clservName, startDate, endDate, startHour, endHour)

Requires(**clservNum**, **eqIdent**, ltUsed, Main, Damage)

Performs(**staffNum**, **clservNum**, Location, schedule)

**FUNCTIONAL DEPENDENCIES:**

cNum→ eFName, eLName, eAddress, salary, eTeleNum

staffNum→ eFName, eLName, eAddress, salary, eTeleNum

eqIdent→ descript, usage, cost

clservNum→clserName, startDate, endDate, startHour, endHour

clservNum,eqIdent→ltUsed, Main, Damage

staffNum,clservNum→Location, schedule

* 2NF:

**NO PARTIAL DEPENDENCIES WERE IDENTIFIED NO NEW TABLES:**

Client(**cNum**, cFName, cLName, cAddress, cTeleNum)

Employee(**staffNum**, eFName, eLName, eAddress, salary, eTeleNum)

Equipment(**eqIdent**, descript, usage, cost)

Cleaning service(**clservNum**, clserName, startDate, endDate, startHour, endHour)

Requires(**clservNum**, **eqIdent**, ltUsed, Main, Damage)

Performs(**staffNum**, **clservNum**, Location, schedule)

* 3NF:

**NO TRANSTITIVE DEPENDENCIES WERE IDENTIFIED NO NEW TABLES:**

Client(**cNum**, cFName, cLName, cAddress, cTeleNum)

Employee(**staffNum**, eFName, eLName, eAddress, salary, eTeleNum)

Equipment(**eqIdent**, descript, usage, cost)

Cleaning service(**clservNum**, clserName, startDate, endDate, startHour, endHour)

Requires(**clservNum**, **eqIdent**, ltUsed, Main, Damage)

Performs(**staffNum**, **clservNum**, Location, schedule)

c.        Validate the logical model against user transactions.

SINCE CLEANING SERVICE TABLE HAS ONE TO MANY RELATIONSHIP WITH CLIENT IT WILL HAVE ONE FOREIGN KEY

d.        Define integrity constraints:

i. Primary key constraints.

Primary keys for each entity cannot be null and must be unique for each value.

ii.     Referential integrity/Foreign key constraints.

If a foreign key contains a value, that value must refer to an existing tuple in the parent relation

iii.  Alternate key constraints (if any).

None.

iv. General constraints (if any).

* Cleaning Services must start and end within business hours and cannot be in the past.
* Cleaning services can be requested by one client from a company at a time.
* Cleaning services must end time greater than start time.
* Equipment description and usage are unique values for the purpose of efficiency
* Company names will be in the First Name column and the word "Company" will occupy the last name column where a company is the client.
* Telephone number has to be 10 digits no more or less
* Cleaning services are null or no option available for the weekend.
* Days of the week are represented as letters ‘M’,’T’,’W’,’H’,’F’.

Diagram

Description automatically generated

      Translate the logical data model for the Oracle Enterprise DBMS.

* 1. Develop SQL code to create the entire database schema, reflecting the constraints identified in previous steps.

CREATE TABLE CLIENTS

(

cNum INT,

cFName VARCHAR(255),

cLName VARCHAR(255),

cAddress VARCHAR(255),

cTeleNum INT CHECK(cTeleNum BETWEEN 1111111111 AND 9999999999),

PRIMARY KEY(cNum)

);

CREATE TABLE EMPLOYEE

(

staffNum INT,

eFName VARCHAR(255),

eLName VARCHAR(255),

eAddress VARCHAR(255),

salary INT,

eTeleNum INT CHECK(eTeleNum BETWEEN 1111111111 AND 9999999999),

PRIMARY KEY(staffNum)

);

CREATE TABLE CLEANINGSERVICE

(

clservNum INT,

startDate VARCHAR(255) CHECK(startDATE IN ('M', 'T', 'W','H', 'F')),

endDate VARCHAR(255) CHECK(endDATE IN ('M', 'T', 'W','H', 'F')),

startHour DATE,

endHour DATE,

cNum INT,

staffNum INT,

PRIMARY KEY(clservNum),

CONSTRAINT KAL CHECK (endHour > startHour),

FOREIGN KEY (cNum) REFERENCES CLIENTS (cNum) ON DELETE CASCADE

);

CREATE TABLE EQUIPMENT

(

eqIdent INT,

discript VARCHAR(255),

usage VARCHAR(255),

cost INT,

clservNum INT,

PRIMARY KEY(eqIdent),

FOREIGN KEY (clservNum) REFERENCES CLEANINGSERVICE (clservNum) ON DELETE CASCADE

);

CREATE TABLE PERFORMS

(

clservNum INT,

staffNum INT,

hours INT,

PRIMARY KEY(staffNum, clservNum),

FOREIGN KEY (staffNum) REFERENCES EMPLOYEE (staffNum) ON DELETE CASCADE,

FOREIGN KEY (clservNum) REFERENCES CLEANINGSERVICE (clservNum) ON DELETE CASCADE

);

CREATE TABLE REQUIRES

(

staffNum INT,

eqIdent INT,

lastUsed VARCHAR(255),

PRIMARY KEY(staffNum, eqIdent),

FOREIGN KEY (staffNum) REFERENCES EMPLOYEE (staffNum) ON DELETE CASCADE,

FOREIGN KEY (eqIdent) REFERENCES EQUIPMENT (eqIdent) ON DELETE CASCADE

);

* 1. Create at least 5 tuples for each relation in your database.

INSERT INTO CLIENTS VALUES(10,'Yénora','Meatyard','58681 Laurel Court',2478818284);

INSERT INTO CLIENTS VALUES(11,'Vérane','Jones','26 Burrows Hill',2807198204);

INSERT INTO CLIENTS VALUES(12,'Noémie','Godart','50 Fair Oaks Circle',8963514440);

INSERT INTO CLIENTS VALUES(13,'Danièle','Bargh','5 Hollow Ridge Alley',1306921052);

INSERT INTO CLIENTS VALUES(14,'Dafnée','Abad','9147 East Plaza',7067838933);

INSERT INTO CLIENTS VALUES(15,'Dorothée','McCroary','2400 Dakota Pass',1216460887);

-----------------------------

INSERT INTO EMPLOYEE VALUES(10,'Léana','Coysh','8841 Farwell Crossing',100000,9023353008);

INSERT INTO EMPLOYEE VALUES(11,'Léane','Ferenczy','7 New Castle Drive',200000,1809996171);

INSERT INTO EMPLOYEE VALUES(12,'Gérald','Mulvenna','01308 Corscot Road',300000,2256159006);

INSERT INTO EMPLOYEE VALUES(13,'Pò','Hennemann','474 Ramsey Terrace',400000,9849927898);

INSERT INTO EMPLOYEE VALUES(14,'Céline','Doodney','02539 5th Point',25000,1179813566);

INSERT INTO EMPLOYEE VALUES(15,'Alizée','Edworthy','33190 Leroy Hill',17000,1176611148);

-----------------------------

INSERT INTO CLEANINGSERVICE VALUES(10,'M','H',TO\_DATE('2020-12-10','YYYY-MM-DD'),TO\_DATE('2020-12-26','YYYY-MM-DD'),10,10);

INSERT INTO CLEANINGSERVICE VALUES(11,'H','F',TO\_DATE('2020-12-11','YYYY-MM-DD'),TO\_DATE('2020-12-17','YYYY-MM-DD'),12,12);

INSERT INTO CLEANINGSERVICE VALUES(12,'F','M',TO\_DATE('2020-12-13','YYYY-MM-DD'),TO\_DATE('2020-12-30','YYYY-MM-DD'),13,13);

INSERT INTO CLEANINGSERVICE VALUES(13,'T','F',TO\_DATE('2020-12-14','YYYY-MM-DD'),TO\_DATE('2020-12-27','YYYY-MM-DD'),14,14);

INSERT INTO CLEANINGSERVICE VALUES(14,'M','W',TO\_DATE('2020-12-19','YYYY-MM-DD'),TO\_DATE('2020-12-20','YYYY-MM-DD'),15,15);

-----------------------------

INSERT INTO EQUIPMENT VALUES(10,'Wax','Seven',18,10);

INSERT INTO EQUIPMENT VALUES(11,'Grout Cleaner','Seven',35,11);

INSERT INTO EQUIPMENT VALUES(13,'Tube Brush','Thirtyfive',15,13);

INSERT INTO EQUIPMENT VALUES(14,'Shiner','Ten',25,14);

INSERT INTO EQUIPMENT VALUES(15,'Flood Extractor','Three',110,10);

INSERT INTO EQUIPMENT VALUES(16,'Sanitizer','Eight',12,11);

-----------------------------

INSERT INTO PERFORMS VALUES(10,10,2);

INSERT INTO PERFORMS VALUES(13,13,6);

INSERT INTO PERFORMS VALUES(10,12,2);

INSERT INTO PERFORMS VALUES(14,14,5);

INSERT INTO PERFORMS VALUES(14,13,4);

-----------------------------

INSERT INTO REQUIRES VALUES(10,10,'MONDAY');

INSERT INTO REQUIRES VALUES(13,13,'TUESDAY');

INSERT INTO REQUIRES VALUES(15,15,'FRIDAY');

INSERT INTO REQUIRES VALUES(10,13,'WEDNESDAY');

INSERT INTO REQUIRES VALUES(10,12,'THURSDAY');

Clients:

Graphical user interface, text, application, table

Description automatically generated

Employee:

Graphical user interface, text, application, table

Description automatically generated

CleaningService:

Table

Description automatically generated

Equipment:

Table

Description automatically generated

Performs:

Table

Description automatically generated

Requires:

Graphical user interface, application

Description automatically generated

* 1. Develop 5 SQL queries using embedded SQL (see Python tutorial).

SELECT \*

FROM EMPLOYEE

WHERE SALARY > 30000

SELECT \*

FROM EQIUPMENT

WHERE COST < 30

Text

Description automatically generated

SELECECT cADDRESS

FROM CLIENTS

Graphical user interface, text, application

Description automatically generated

SELECT \*

FROM CLEANINGSERVICE

A picture containing text, scoreboard

Description automatically generated

SELECT cLNAME, cFNAME

FROM CLIENTS

Text

Description automatically generated

* 1. Upload all the code to GitHub.