Project – SQL Statements

R Marx

Lecturer: Linda Redelinghuys

36581852

GitHub: https://github.com/Tanaille/cmpg223-project

Table of Contents

[1. Table Creation 3](#_Toc84804607)

[1.1 City 3](#_Toc84804608)

[1.2 User Type 3](#_Toc84804609)

[1.3 Animal Type 3](#_Toc84804610)

[1.4 Users 3](#_Toc84804611)

[1.5 Passwords 3](#_Toc84804612)

[1.6 Adopter 4](#_Toc84804613)

[1.7 Animal 4](#_Toc84804614)

[1.8 Adoption 4](#_Toc84804615)

[2. SELECT Queries 5](#_Toc84804616)

[2.1 User authentication 5](#_Toc84804617)

[2.2 Get NULL values 5](#_Toc84804618)

[2.3 Check if record exists 5](#_Toc84804619)

[2.4 Value retrieval 5](#_Toc84804620)

[2.5 Populate and filter form controls 7](#_Toc84804621)

[2.6 Reporting 8](#_Toc84804622)

[3. Maintenance Queries (INSERT, UPDATE, DELETE) 10](#_Toc84804623)

[3.1 Adopters 10](#_Toc84804624)

[3.2 Adoptions 10](#_Toc84804625)

[3.3 Animals 10](#_Toc84804626)

[3.4 Animal Types 11](#_Toc84804627)

[3.5 Cities 11](#_Toc84804628)

[3.6 Passwords 11](#_Toc84804629)

[3.7 Users 11](#_Toc84804630)

[3.8 User Types 12](#_Toc84804631)

# 1. Table Creation

## 1.1 City

CREATE TABLE CITY (

City\_ID int IDENTITY(1,1) NOT NULL PRIMARY KEY,

City\_Name varchar(35)

);

## 1.2 User Type

CREATE TABLE USER\_TYPE (

User\_Type tinyint IDENTITY(1,1) NOT NULL PRIMARY KEY,

Type\_Description varchar(20)

);

## 1.3 Animal Type

CREATE TABLE ANIMAL\_TYPE (

Type\_Number int IDENTITY(1,1) NOT NULL PRIMARY KEY,

Type\_Name varchar(35)

);

## 1.4 Users

CREATE TABLE USERS (

User\_Number int IDENTITY(1,1) NOT NULL,

User\_Type tinyint,

User\_Name varchar(35),

User\_Surname varchar(35),

CONSTRAINT PK\_USERS PRIMARY KEY (User\_Number),

CONSTRAINT FK\_USERS\_USER\_TYPE FOREIGN KEY (User\_Type) REFERENCES USER\_TYPE (User\_Type) ON DELETE SET NULL

);

## 1.5 Passwords

CREATE TABLE PASSWORDS (

User\_Number int NOT NULL,

Hash varchar(100),

Salt varchar(100),

CONSTRAINT PK\_PASSWORDS PRIMARY KEY (User\_Number),

CONSTRAINT FK\_PASSWORDS\_USER FOREIGN KEY (User\_Number) REFERENCES USERS (User\_Number) ON DELETE CASCADE

);

## 1.6 Adopter

CREATE TABLE ADOPTER (

Adopter\_Number int IDENTITY(1,1) NOT NULL,

Adopter\_Surname varchar(35),

Adopter\_Name varchar(35),

Adopter\_Contact\_Number char(10),

Adopter\_Street\_Number varchar(6),

Adopter\_Street\_Name varchar(50),

City\_ID int,

CONSTRAINT PK\_ADOPTER PRIMARY KEY (Adopter\_Number),

CONSTRAINT FK\_ADOPTER\_CITY FOREIGN KEY (City\_ID) REFERENCES CITY (City\_ID) ON DELETE SET NULL

);

## 1.7 Animal

CREATE TABLE ANIMAL (

Animal\_Number int IDENTITY(1,1) NOT NULL,

Animal\_Name varchar(35),

Type\_Number int,

Add\_Date date,

CONSTRAINT PK\_ANIMAL PRIMARY KEY (Animal\_Number),

CONSTRAINT FK\_ANIMAL\_ANIMAL\_TYPE FOREIGN KEY (Type\_Number) REFERENCES ANIMAL\_TYPE (Type\_Number) ON DELETE SET NULL

);

## 1.8 Adoption

CREATE TABLE ADOPTION (

Adoption\_Number int IDENTITY(1,1) NOT NULL,

Adopter\_Number int,

Animal\_Number int,

User\_Number int,

Result\_Description varchar(10),

Adoption\_Date date,

CONSTRAINT PK\_ADOPTION PRIMARY KEY (Adoption\_Number),

CONSTRAINT FK\_ADOPTION\_ADOPTER FOREIGN KEY (Adopter\_Number) REFERENCES ADOPTER (Adopter\_Number) ON DELETE CASCADE,

CONSTRAINT FK\_ADOPTION\_ANIMAL FOREIGN KEY (Animal\_Number) REFERENCES ANIMAL (Animal\_Number) ON DELETE CASCADE,

CONSTRAINT FK\_ADOPTION\_USERS FOREIGN KEY (User\_Number) REFERENCES USERS (User\_Number)

);

# 2. SELECT Queries

## 2.1 User authentication

SELECT USERS.User\_Name, PASSWORDS.Hash, PASSWORDS.Salt

FROM USERS

INNER JOIN PASSWORDS

ON USERS.User\_Number = PASSWORDS.User\_Number

WHERE USERS.User\_Name = @userName;

## 2.2 Get NULL values

SELECT \*

FROM table

WHERE + column IS NULL;

## 2.3 Check if record exists

SELECT TOP 1 column

FROM table

WHERE column = @value;

## 2.4 Value retrieval

**Animal type number from animal type name**

SELECT Type\_Number

FROM ANIMAL\_TYPE

WHERE Type\_Name = @typeName;

**Animal type number from animal name**

SELECT Type\_Number

FROM ANIMAL

WHERE Animal\_Name = @name AND Type\_Number IS NOT NULL;

**Animal number from animal name**

SELECT Animal\_Number

FROM ANIMAL

WHERE Animal\_Name = @animalName;

**Animal type name from animal type number**

SELECT Type\_Name

FROM ANIMAL\_TYPE

WHERE Type\_Number = @typeNumber;

**User type number from user type name**

SELECT User\_Type

FROM USER\_TYPE

WHERE Type\_Description = @typeName;

**User number from user name**

SELECT User\_Number

FROM USERS

WHERE User\_Name = @userName AND User\_Surname = @userSurname;

**User surname from user name**

SELECT User\_Surname

FROM USERS

WHERE User\_Name = @userName;

**City ID from city name**

SELECT City\_ID

FROM CITY

WHERE City\_Name = @cityName;

**Adopter surname from adopter name**

SELECT Adopter\_Surname

FROM ADOPTER

WHERE Adopter\_Name = @adopterName;

**Adopter number from adopter name**

SELECT Adopter\_Number

FROM ADOPTER

WHERE Adopter\_Name = @adopterName AND Adopter\_Surname = @adopterSurname;

**Adoption number from animal number**

SELECT Adoption\_Number

FROM ADOPTION

WHERE Animal\_Number = @animalNumber;

**Adoption number from adopter name**

SELECT Adoption\_Number

FROM ADOPTION

WHERE Adopter\_Number = @adopterNumber;

**User name and surname from user number**

SELECT User\_Name, User\_Surname

FROM USERS

WHERE User\_Number = @userNumber;

**User type description from user name**

SELECT Type\_Description

FROM USER\_TYPE

WHERE USERS.User\_Name = @userName;

**User type number from user name**

SELECT User\_Type

FROM USERS

WHERE User\_Name = @userName;

**User type description from type number**

SELECT Type\_Description

FROM USER\_TYPE

WHERE User\_Type = @typeNumber;

**Authenticate a user**

SELECT USERS.User\_Name, PASSWORDS.Hash, PASSWORDS.Salt

FROM USERS

INNER JOIN PASSWORDS

ON USERS.User\_Number = PASSWORDS.User\_Number

WHERE USERS.User\_Name = @userName;

## 2.5 Populate and filter form controls

**Populate combobox**

SELECT column

FROM table

ORDER BY column ASC;

**Populate datagridview**

SELECT columns

FROM tables[0];

SELECT columns

FROM tables[0] a, tables[1] b WHERE a.key = b.key;

SELECT columns

FROM tables[0] a, tables[1] b, tables[2] c

WHERE a.key = b.key AND b.key = c.key;

SELECT columns

FROM ADOPTION a, ADOPTER b, USERS c, ANIMAL d

WHERE a.Adopter\_Number = b.Adopter\_Number AND a.User\_Number = c.User\_Number AND a.Animal\_Number = d.Animal\_Number;

**Populate listbox**

SELECT columns[0]

FROM table a

ORDER BY columns[0] ASC;

SELECT columns[0], columns[1]

FROM table[0] a, table[1] b

WHERE a.key = b.key

ORDER BY columns[0] ASC;

**Filter listbox**

SELECT column

FROM table

WHERE column

LIKE '%" + filter + "%'

ORDER BY column ASC;

## 2.6 Reporting

**Animals adopter per time period**

SELECT a.Animal\_Name, a.Animal\_Number, b.Adopter\_Name, c.Adoption\_Date, d.User\_Name

FROM ANIMAL a, ADOPTER b, ADOPTION c, USERS d

WHERE a.Animal\_Number = c.Animal\_Number AND b.Adopter\_Number = c.Adopter\_Number AND d.User\_Number = c.User\_Number AND CAST(c.Adoption\_Date AS date) BETWEEN CAST(@start AS date) AND CAST(@end AS DATE)

ORDER BY a.Animal\_Name ASC;

**Total number of adoptions**

SELECT COUNT(1)

FROM ADOPTION

WHERE Adoption\_Date BETWEEN CAST(@start AS date) AND CAST(@end AS date);

**Animals added per time period**

SELECT a.Animal\_Number, a.Animal\_Name, b.Type\_Name, a.Add\_Date

FROM ANIMAL a, ANIMAL\_TYPE b

WHERE a.Type\_Number = b.Type\_Number AND a.Add\_Date BETWEEN CAST(@start AS date) AND CAST(@end AS date)

ORDER BY a.Animal\_Number ASC;

**Total number of animals**

SELECT COUNT(1)

FROM ANIMAL

WHERE Add\_Date BETWEEN CAST(@start AS date) AND CAST(@end AS date);

**All animal type names**

SELECT Type\_Name

FROM ANIMAL\_TYPE;

**Number of animals per type**

SELECT Type\_Name, COUNT(\*)

FROM ANIMAL a, ANIMAL\_TYPE b

WHERE a.Type\_Number = b.Type\_Number

GROUP BY Type\_Name;

**Animals added per time period**

SELECT Add\_Date, COUNT(\*)

FROM ANIMAL GROUP BY Add\_Date;

# 3. Maintenance Queries (INSERT, UPDATE, DELETE)

## 3.1 Adopters

INSERT INTO ADOPTER (Adopter\_Name, Adopter\_Surname, Adopter\_Contact\_Number, Adopter\_Street\_Number, Adopter\_Street\_Name, City\_ID)

VALUES (@name, @surname, @contactNumber, @streetNumber, @streetName, @cityID);

UPDATE ADOPTER

SET Adopter\_Name = @name, Adopter\_Surname = @surname, Adopter\_Contact\_Number = @contactNumber, Adopter\_Street\_Number = @streetNumber, Adopter\_Street\_Name = @streetName, City\_ID = @cityID

WHERE Adopter\_Number = @adopterNumber;

DELETE FROM ADOPTER

WHERE Adopter\_Number = @adopterNumber;

## 3.2 Adoptions

INSERT INTO ADOPTION (Adopter\_Number, Animal\_Number, User\_Number, Result\_Description, Adoption\_Date)

VALUES (@adopterNumber, @animalNumber, @userNumber, @resultDescription, @adoptionDate);

UPDATE ADOPTION

SET Adopter\_Number = @adopterNumber, Animal\_Number = @animalNumber, Result\_Description = @resultDescription

WHERE Adoption\_Number = @adoptionNumber;

DELETE FROM ADOPTION

WHERE Adoption\_Number = @adoptionNumber;

## 3.3 Animals

INSERT INTO ANIMAL (Animal\_Name, Type\_Number, Add\_Date)

VALUES (@animalName, @animalTypeNumber, @addDate);

UPDATE ANIMAL

SET Animal\_Name = @newName, Type\_Number = @newNumber

WHERE Animal\_Number = @oldNumber;

DELETE FROM ANIMAL

WHERE Animal\_Name = @name;

## 3.4 Animal Types

INSERT INTO ANIMAL\_TYPE (Type\_Name)

VALUES (@typeName);

UPDATE ANIMAL\_TYPE

SET Type\_Name = @newTypeName

WHERE Type\_Name = @oldTypeName;

DELETE FROM ANIMAL\_TYPE

WHERE Type\_Name = @typeName;

## 3.5 Cities

INSERT INTO CITY (City\_Name)

VALUES (@cityName);

UPDATE CITY

SET City\_Name = @cityName

WHERE City\_ID = @cityID;

DELETE FROM CITY

WHERE City\_Name = @cityName;

## 3.6 Passwords

INSERT INTO PASSWORDS (User\_Number, Hash, Salt)

VALUES (@userNumber, @hash, @salt);

UPDATE PASSWORDS

SET Hash = @hash, Salt = @salt

WHERE User\_Number = @userNumber;

DELETE FROM PASSWORDS

WHERE User\_Number = @userNumber;

## 3.7 Users

INSERT INTO USERS (User\_Type, User\_Name, User\_Surname)

VALUES (@typeNumber, @name, @surname);

UPDATE USERS

SET User\_Name = @userName, User\_Surname = @userSurname, User\_Type = @typeNumber

WHERE User\_Number = @userNumber;

DELETE FROM USERS

WHERE User\_Number = @userNumber;

## 3.8 User Types

INSERT INTO USER\_TYPE (Type\_Description)

VALUES (@typeDescription);

DELETE FROM USER\_TYPE

WHERE Type\_Description = @typeDescription;