Q1

CREATE TABLE MOVIES (

movieid INT PRIMARY KEY,

title VARCHAR(35) NOT NULL,

releaseYear INT NOT NULL,

director INT NOT NULL,

score DECIMAL(3, 2) CHECK (score > 3 AND score < 10)

);

A screenshot of a computer

Description automatically generated

CREATE TABLE ACTORS (

actorid INT PRIMARY KEY,

firstName VARCHAR(20) NOT NULL,

lastName VARCHAR(30) NOT NULL

);

A screenshot of a computer

Description automatically generated

CREATE TABLE CASTINGS (

movieid INT,

actorid INT,

PRIMARY KEY (movieid, actorid),

FOREIGN KEY (movieid) REFERENCES MOVIES(movieid),

FOREIGN KEY (actorid) REFERENCES ACTORS(actorid)

); A screenshot of a computer

Description automatically generated with medium confidence

CREATE TABLE DIRECTORS (

directorid INT PRIMARY KEY,

firstname varchar(20) NOT NULL,

lastname varchar(30) NOT NULL

); A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Q2

ALTER TABLE MOVIES

ADD CONSTRAINT fk\_director

FOREIGN KEY (director)

REFERENCES directors(directorid);

A screenshot of a computer

Description automatically generated

Q3

ALTER TABLE MOVIES

ADD CONSTRAINT unique\_movie\_title UNIQUE (title); A screenshot of a computer

Description automatically generated with medium confidence

Q4

INSERT ALL

INTO DIRECTORS (directorid, firstname, lastname) VALUES (1010, 'Rob', 'Minkoff')

INTO DIRECTORS (directorid, firstname, lastname) VALUES (1020, 'Bill', 'Condon')

INTO DIRECTORS (directorid, firstname, lastname) VALUES (1050, 'Josh', 'Cooley')

INTO DIRECTORS (directorid, firstname, lastname) VALUES (2010, 'Brad', 'Bird')

INTO DIRECTORS (directorid, firstname, lastname) VALUES (3020, 'Lake', 'Bell')

SELECT \* FROM DUAL;

A screenshot of a computer

Description automatically generated with medium confidence

INSERT ALL

INTO MOVIES (MOVIEID, TITLE, RELEASEYEAR, DIRECTOR, SCORE) VALUES (100, 'The Lion King', 2019, 3020, 3.50)

INTO MOVIES (MOVIEID, TITLE, RELEASEYEAR, DIRECTOR, SCORE) VALUES (200, 'Beauty and the Beast', 2017, 1050, 4.20)

INTO MOVIES (MOVIEID, TITLE, RELEASEYEAR, DIRECTOR, SCORE) VALUES (300, 'Toy Story 4', 2019, 1020, 4.50)

INTO MOVIES (MOVIEID, TITLE, RELEASEYEAR, DIRECTOR, SCORE) VALUES (400, 'Mission Impossible', 2018, 2010, 5.00)

INTO MOVIES (MOVIEID, TITLE, RELEASEYEAR, DIRECTOR, SCORE) VALUES (500, 'The Secret Life of Pets', 2016, 1010, 3.90)

SELECT \* FROM DUAL;

A screenshot of a computer

Description automatically generated

Q5

DROP TABLE CASTINGS; A screenshot of a computer

Description automatically generated

DROP TABLE MOVIES;

A screenshot of a computer

Description automatically generated

DROP TABLE DIRECTORS; A screenshot of a computer

Description automatically generated

When removing tables without any dependencies, the order doesn't matter. However, if there are dependencies like foreign key constraints, you need to drop the dependent tables first before dropping the referenced tables to avoid any constraint violations and errors.

Q6

CREATE TABLE employeecopy AS

SELECT \*

FROM RETAILEMPLOYEES

WHERE 1=2;

A screenshot of a computer

Description automatically generated

Q7

ALTER TABLE employeecopy

ADD username VARCHAR(50) NULL;

A screenshot of a computer

Description automatically generated

Q8

INSERT INTO EMPLOYEECOPY (EMPLOYEENUMBER, LASTNAME, FIRSTNAME, EXTENSION, EMAIL, OFFICECODE, REPORTSTO, JOBTITLE, USERNAME)

SELECT EMPLOYEENUMBER, LASTNAME, FIRSTNAME, EXTENSION, EMAIL, OFFICECODE, REPORTSTO, JOBTITLE, NULL

FROM RETAILEMPLOYEES;

A screenshot of a computer

Description automatically generated

Q9

UPDATE EMPLOYEECOPY

SET USERNAME = CONCAT(EMPLOYEENUMBER, '@seneca.ca');

A screenshot of a computer

Description automatically generated

Q10

DELETE FROM employeecopy;

A screenshot of a computer

Description automatically generated SELECT \* FROM EMPLOYEECOPY

A screenshot of a computer

Description automatically generated

Q11

INSERT INTO RETAILEMPLOYEES (EMPLOYEENUMBER, LASTNAME, FIRSTNAME, EXTENSION, EMAIL, OFFICECODE, REPORTSTO, JOBTITLE)

VALUES (1234, 'Gyeongrok', 'Oh', 'x1234', 'goh3@seneca.ca', 4, 1088, 'Cashier');

A screenshot of a computer

Description automatically generated

Q12

SELECT \*

FROM RETAILEMPLOYEES

WHERE EMPLOYEENUMBER = 1234;

A screenshot of a computer

Description automatically generated

Q13

UPDATE RETAILEMPLOYEES

SET JOBTITLE = 'Head Cashier'

WHERE EMPLOYEENUMBER = 1234;

A screenshot of a computer

Description automatically generated with medium confidence

Q14

INSERT INTO RETAILEMPLOYEES (EMPLOYEENUMBER, LASTNAME, FIRSTNAME, EXTENSION, EMAIL, OFFICECODE, REPORTSTO, JOBTITLE)

VALUES (9876, 'Smith', 'John', 'x9876', 'john.smith@example.com', 4, 1234, 'Cashier');; A screenshot of a computer

Description automatically generated with medium confidence

Q15

DELETE FROM RETAILEMPLOYEES

WHERE EMPLOYEENUMBER = 1234;

A screenshot of a computer

Description automatically generated

to delete a row from the "RETAILEMPLOYEES" table where the "EMPLOYEENUMBER" is 1234. However, the deletion is not allowed because it would violate the foreign key constraint named "EMP\_RTEMP\_FK." This constraint signifies that there are child records in another table referencing specific rows in the "RETAILEMPLOYEES" table.

Q16

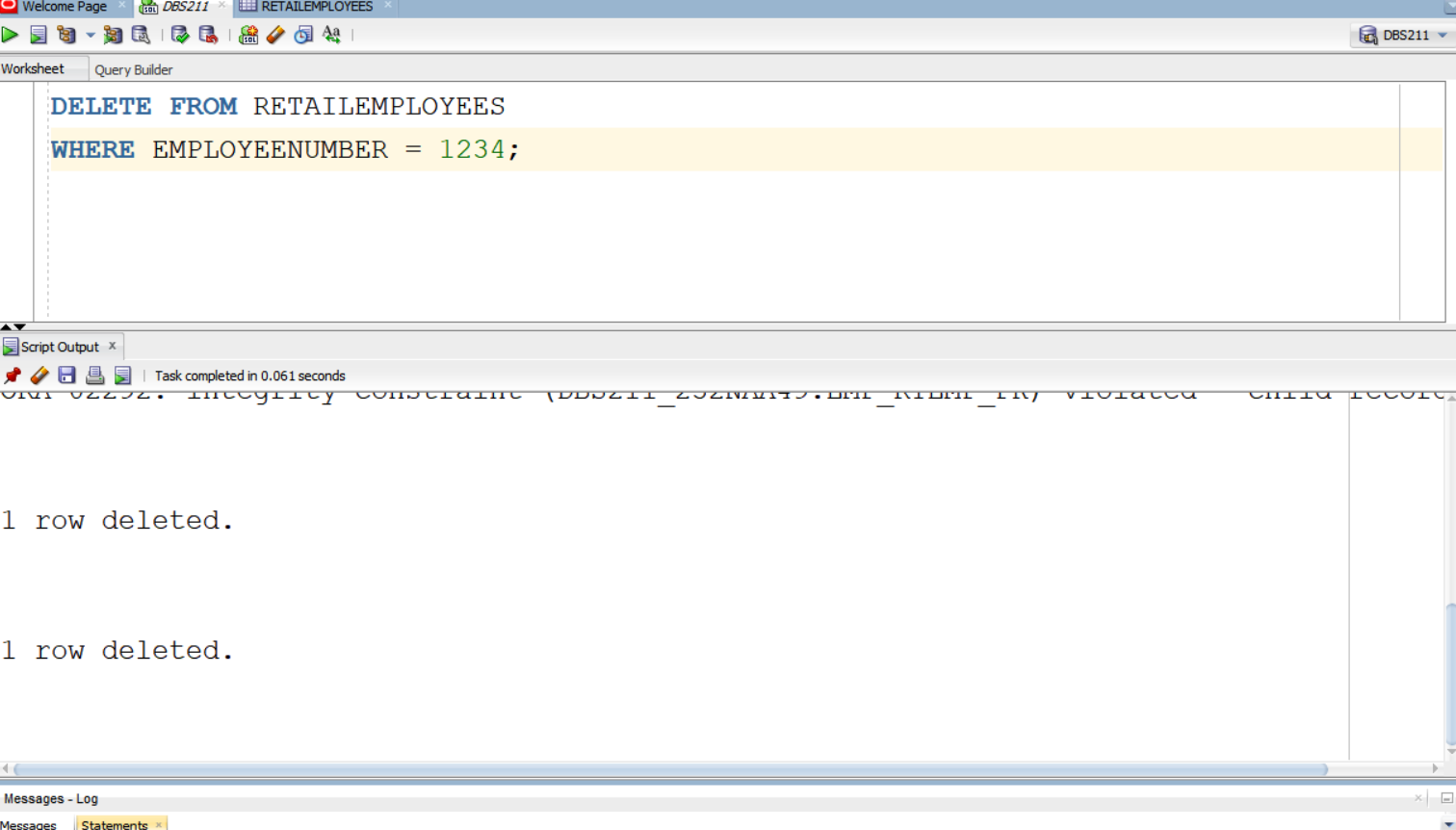
DELETE FROM RETAILEMPLOYEES

WHERE EMPLOYEENUMBER = 9876;



DELETE FROM RETAILEMPLOYEES

WHERE EMPLOYEENUMBER = 1234;



Q17

INSERT ALL

INTO RETAILEMPLOYEES VALUES (9876, 'Smith', 'John', 'x9876', 'john.smith@example.com', 4, 1088, 'Cashier')

INTO RETAILEMPLOYEES VALUES (1234, 'Gyeongrok', 'Oh', 'x1234', 'goh3@seneca.ca', 4, 1088, 'Head Cashier')

SELECT\* FROM DUAL

A screenshot of a computer

Description automatically generated with medium confidence

Q18

DELETE FROM RETAILEMPLOYEES

WHERE EMPLOYEENUMBER = 9876 OR EMPLOYEENUMBER = 1234; A screenshot of a computer

Description automatically generated

Q19

INSERT INTO RETAILORDERS (ORDERNUMBER, ORDERDATE, REQUIREDDATE, SHIPPEDDATE, STATUS, COMMENTS, CUSTOMERNUMBER)

VALUES (11914, TO\_DATE('2021-09-17', 'YYYY-MM-DD'), TO\_DATE('2021-09-22', 'YYYY-MM-DD'), NULL, 'Pending',NULL, 103); A screenshot of a computer

Description automatically generated with medium confidence A screenshot of a computer

Description automatically generated with medium confidence

Q20

INSERT INTO RETAILPRODUCTS (PRODUCTCODE, PRODUCTNAME, PRODUCTLINE, RETAILPRODUCTSCALE, PRODUCTVENDOR, PRODUCTDESCRIPTION, QUANTITYINSTOCK, BUYPRICE, MSRP)

VALUES ('S111\_111', '2020 Bugatti Veyron', 'Classic Cars', '1:12', 'Exquisite Cars', 'A stunning model of the iconic 2020 Bugatti Veyron.', 6800, 299.99, 599.99);

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence