1. Complete the table below with the Data Type you would use for the listed attribute (5 pts).

|  |  |
| --- | --- |
| **Attribute** | **Data Types** |
| Social Security Number. | CHAR(9) |
| A field used to flag if a package has been picked up or not. Assume only two options. | BOOLEAN |
| The time field on a monthly police call log. | TIMESTAMP |
| Student Identification number. | CHAR(8) |
| Hourly temperature with up to 2 decimal places. | NUMERIC(DOUBLE) |
| Newspaper Names | VARCHAR(20) |

Use the following tables, ART and ORIGIN, to answer all parts of question 2 and 3.

**ART**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ITEM** | **NAME** | **ARTIST** | **ORIGIN\_ID** | **DATING** | **MEDIA** |
| 9182 | One Hundred Horses | Lang Shining | 1111 | 960 to 1127 | Painting |
| 6922 | The Great Wave off Kanagawa | Katsushika Hokusai | 6943 | 1829 to 1833 | Painting |
| 2049 | Toluvila statue | NULL | 8415 | 300 to 400 | Statuary |
| 2038 | Sasanian silver vessel | NULL | 1598 | 700 to 722 | Silver |
| 3964 | Nymph of the Luo River | Gu Kaizhi | 1111 | 317 to 420 | Painting |
| 3097 | The Hunt of the Unicorn | NULL | 3543 | 1680 | Tapestries |

**ORIGIN**

|  |  |
| --- | --- |
| **ORIGIN\_ID** | **LOCATION** |
| 1111 | China |
| 3543 | France |
| 6943 | Japan |
| 8415 | Sri Lanka |
| 1598 | Tajikistan |

1. Use the above tables ART and ORIGIN to answer parts a, b, c, and d.
   1. Complete the table below with the Data Type you would use for the listed columns (7 pts).

|  |  |
| --- | --- |
| **Attribute** | **Data Type** |
| ART.ITEM | CHAR(4) |
| ART.NAME | VARCHAR(25) |
| ART.ARTIST | VARCHAR(25) |
| ART.ORIGIN\_ID | CHAR(4) |
| ART.DATING | For this I would WANT to use a DATE data type but by the way its formatted with the word “to” in it, I would likely use CHAR(25) |
| ART.MEDIA | VARCHAR(15) |
| ORIGIN.ORIGIN\_ID | CHAR(4) |
| ORIGIN.LOCATION | VARCHAR(25) |

* 1. Assume the development policies of the corporation you are working for require you to add Primary Keys in an ALTER TABLE commands separate from the CREATE TABLE. Write the statements that you would use to create ART and ORIGIN in a database and add the Primary Keys (10 pts).

DROP TABLE ART;

DROP TABLE ORIGIN;

CREATE TABLE ORIGIN

(

Origin\_ID CHAR(4) NOT NULL,

Location VARCHAR(25) NOT NULL

);

ALTER TABLE ORIGIN ADD PRIMARY KEY (Origin\_ID);

CREATE TABLE ART

(

Item CHAR(4) NOT NULL,

Origin\_ID CHAR(4) NOT NULL,

Name VARCHAR(35),

Artist VARCHAR(35),

Dating VARCHAR(35),

Media VARCHAR(35)

);

ALTER TABLE ART ADD PRIMARY KEY (Item);

* 1. The ORIGIN\_ID in the ART table is a Foreign Key the reference the ORIGIN\_ID in the ORIGIN table. Write the ALTER TABLE statement you would use to define the Foreign Key constraint (5 pts).

ALTER TABLE ART

ADD CONSTRAINT fk\_origin

FOREIGN KEY(Origin\_ID)

REFERENCES ORIGIN(Origin\_ID)

ON DELETE CASCADE;

* 1. Execute your CREATE TABLE statements and ALTER TABLE statements from part b and c on the SCSP Oracle database and grant KOOBP access to the tables with the following two statements (4 pts).

GRANT SELECT ON ART TO KOOBP;

GRANT SELECT ON ORIGIN TO KOOBP;

1. Use the above tables ART and ORIGIN to answer the following parts:
   1. The traveling “Art of China” exhibit at the very small museum using the database detailed above has moved to another museum. So, all artwork from China is no longer at the museum. Write the DELETE statement you would use to remove China from the ORIGIN table (5 pts).

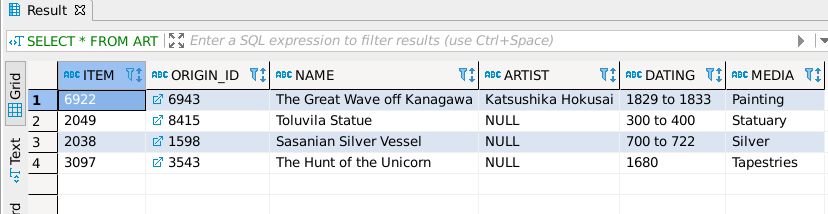
DELETE

FROM

ORIGIN

WHERE

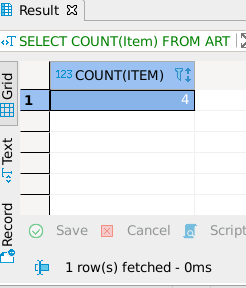
Location = ‘China’;



* 1. Assuming the Foreign Key constraint from part c had ON DELETE CASCADE, if you were to execute the DELETE statement from part e, what would be the results of the following statement (3 pts):

SELECT COUNT(ITEM)

FROM ART;



The output is 4.

1. Use the above tables ART and ORIGIN to answer the following:
   1. Write the transaction to DELETE all items from the ART table and INSERT 5 of your favorite Art pieces into the database. Show the statements use and execute them on SCSP (10 pts).

DELETE

FROM ART;

INSERT ALL

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('9182','The Thinker','Auguste Rodin','3543','1902','Statuary')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('6922','Nine Dragons','Chen Rong','1111','1244','Painting')

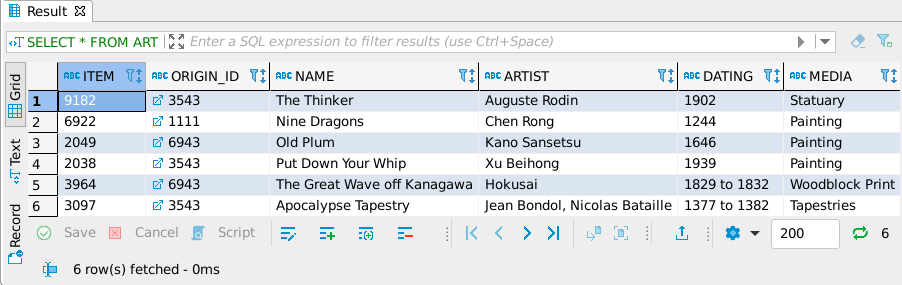
INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('2049','Old Plum','Kano Sansetsu','6943','1646','Painting')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('2038','Put Down Your Whip','Xu Beihong','3543','1939','Painting')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('3964','The Great Wave off Kanagawa','Hokusai','6943','1829 to 1832','Woodblock Print')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('3097','Apocalypse Tapestry','Jean Bondol, Nicolas Bataille','3543','1377 to 1382','Tapestries')

SELECT \* FROM DUAL;



1. Use the above tables ART and ORIGIN to answer this question:

You are asked to convert the LOCATION column in the ORIGIN table to two separate columns, COUNTRY and CITY. Write the statements you would need to:

* Add two new columns COUNTRY and CITY to the ORIGIN table,
* Copy the data in the LOCATION column to the COUNTRY column,
* Delete the LOCATION column.

Show the statements used and then execute them on SCSP (12 pts).

ALTER TABLE ORIGIN

ADD

(

COUNTRY VARCHAR(35),

CITY VARCHAR(35)

);

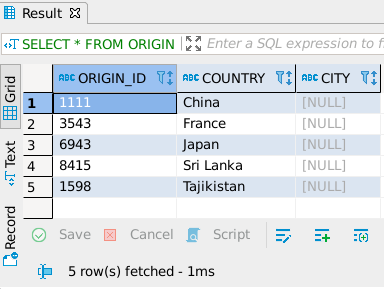
UPDATE ORIGIN

SET COUNTRY=LOCATION;

ALTER TABLE ORIGIN

DROP COLUMN LOCATION;

SELECT \* FROM ORIGIN;



1. Connect to the scsp database and use the os\_hr tables to answer the following questions.

KOOBP.OS\_HR\_JOBS

KOOBP.OS\_HR\_JOB\_HISTORY

KOOBP.OS\_HR\_DEPARTMENTS

KOOBP.OS\_HR\_EMPLOYEES

KOOBP.OS\_HR\_LOCATIONS

KOOBP.OS\_HR\_REGIONS

KOOBP.OS\_HR\_COUNTRIES

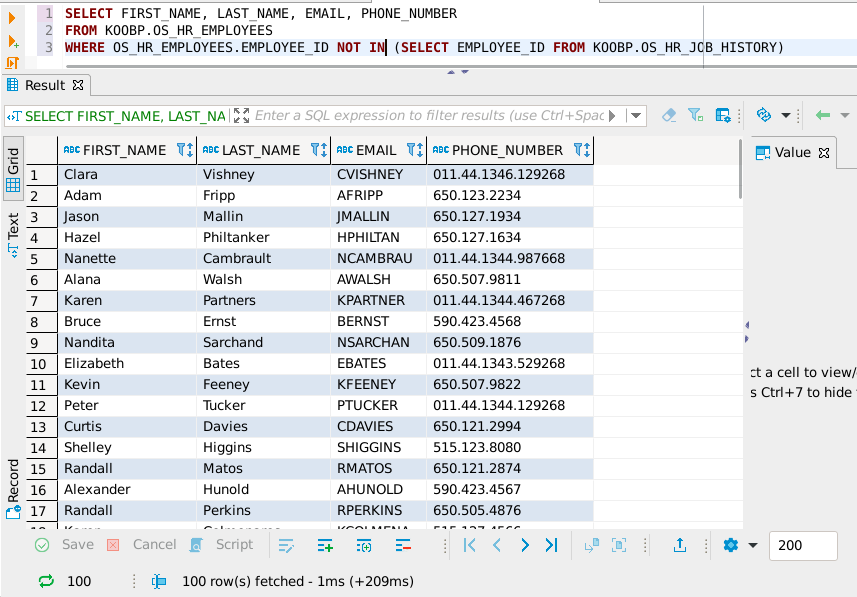
For each answer, show the SQL statement you used to get it and a screen shot of the results box.

* 1. New information has been found pertaining to the employee contact list, from the previous homework, and the query we used for the employee contact list is wrong.   
       
     The contact list has people that no longer work for the company. For example Lex De Haan in on the contact list, but no longer with the company. The KOOBP.OS\_HR\_JOB\_HISTORY table has a list of employees who are no longer working from the company. Using a nested query, get a corrected employee contact list with first name, last name, email, and phone number for each currently employed person. Show the statement used and a screen shot of the results box (3 pts).

SELECT FIRST\_NAME, LAST\_NAME, EMAIL, PHONE\_NUMBER

FROM KOOBP.OS\_HR\_EMPLOYEES

WHERE OS\_HR\_EMPLOYEES.EMPLOYEE\_ID NOT IN (SELECT EMPLOYEE\_ID FROM KOOBP.OS\_HR\_JOB\_HISTORY)



* 1. Using nested queries, a GROUP BY clause, and Aggregate functions, get a list of manager’s first names, last names, email, manager\_id, and the count of employees who report to them. Show the statement used and a screen capture of the results box (5 pts).

SELECT FIRST\_NAME, LAST\_NAME, EMAIL,EMPLOYEES.MANAGER\_ID, EMP\_COUNT

FROM KOOBP.OS\_HR\_EMPLOYEES EMPLOYEES

RIGHT JOIN (

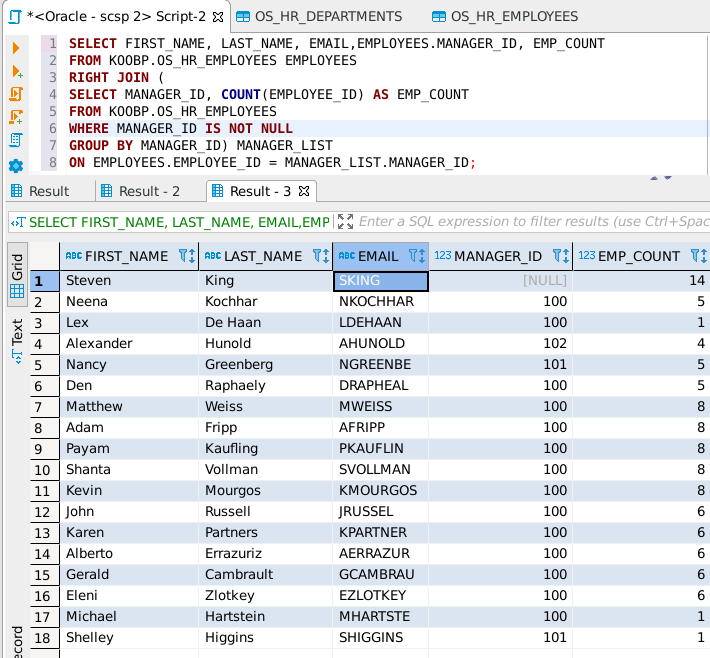
SELECT MANAGER\_ID, COUNT(EMPLOYEE\_ID) AS EMP\_COUNT

FROM KOOBP.OS\_HR\_EMPLOYEES

WHERE MANAGER\_ID IS NOT NULL

GROUP BY MANAGER\_ID) MANAGER\_LIST

ON EMPLOYEES.EMPLOYEE\_ID = MANAGER\_LIST.MANAGER\_ID;



* 1. Using correlated nested queries, show the employee\_id, first name, last name, email, job id, department name, and hire date for the most recent hires for each department, ordered by department name. Show the statement and a screen capture of the results box (8 pts).

SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, EMAIL, JOB\_ID, HIRE\_DATE, DEPT.DEPARTMENT\_NAME

FROM KOOBP.OS\_HR\_EMPLOYEES E

INNER JOIN (

SELECT DEPARTMENT\_ID, DEPARTMENT\_NAME

FROM KOOBP.OS\_HR\_DEPARTMENTS

ORDER BY DEPARTMENT\_NAME) DEPT

ON E.DEPARTMENT\_ID = DEPT.DEPARTMENT\_ID

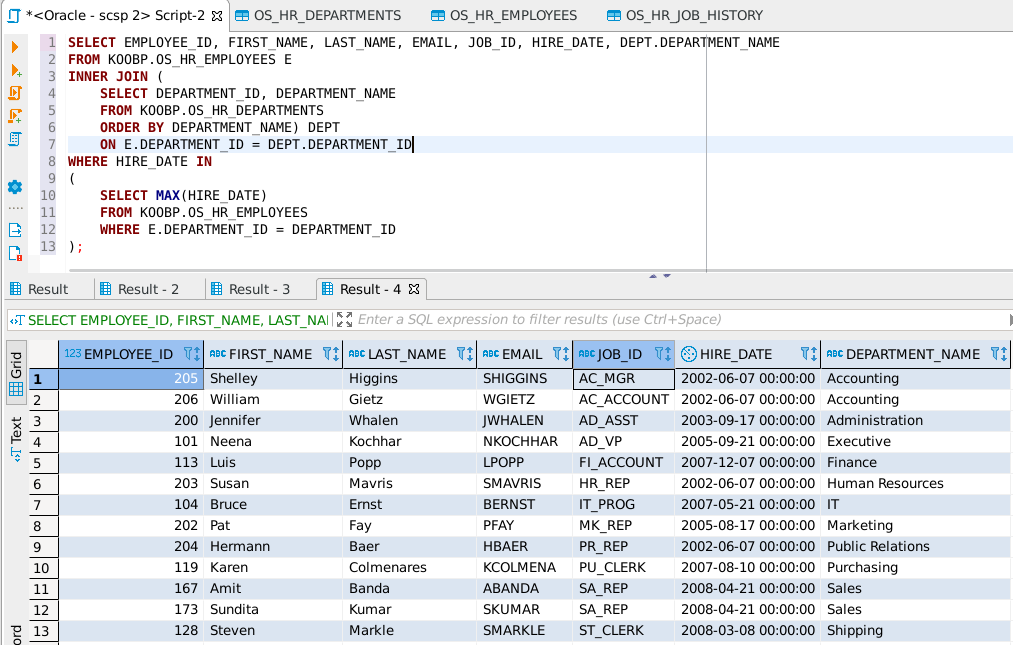
WHERE HIRE\_DATE IN

(

SELECT MAX(HIRE\_DATE)

FROM KOOBP.OS\_HR\_EMPLOYEES

WHERE E.DEPARTMENT\_ID = DEPARTMENT\_ID

);

BELOW IS SCRIPT USED TO GENERATE ANSWERS TO QUESTIONS 1-5

DROP TABLE ART;

DROP TABLE ORIGIN;

CREATE TABLE ORIGIN

(

Origin\_ID CHAR(4) NOT NULL,

Location VARCHAR(25) NOT NULL

);

ALTER TABLE ORIGIN ADD PRIMARY KEY (Origin\_ID);

CREATE TABLE ART

(

Item CHAR(4) NOT NULL,

Origin\_ID CHAR(4) NOT NULL,

Name VARCHAR(35),

Artist VARCHAR(35),

Dating VARCHAR(25),

Media VARCHAR(15)

);

ALTER TABLE ART ADD PRIMARY KEY (Item);

ALTER TABLE ART

ADD CONSTRAINT fk\_origin

FOREIGN KEY(Origin\_ID)

REFERENCES ORIGIN(Origin\_ID)

ON DELETE CASCADE;

GRANT SELECT ON ART TO KOOBP;

GRANT SELECT ON ORIGIN TO KOOBP;

INSERT ALL

INTO ORIGIN (Origin\_ID,Location) VALUES('1111','China')

INTO ORIGIN (Origin\_ID,Location) VALUES('3543','France')

INTO ORIGIN (Origin\_ID,Location) VALUES('6943','Japan')

INTO ORIGIN (Origin\_ID,Location) VALUES('8415','Sri Lanka')

INTO ORIGIN (Origin\_ID,Location) VALUES('1598','Tajikistan')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('9182','One Hundred Horses','Lang Shining','1111','960 to 1127','Painting')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('6922','The Great Wave off Kanagawa','Katsushika Hokusai','6943','1829 to 1833','Painting')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('2049','Toluvila Statue','NULL','8415','300 to 400','Statuary')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('2038','Sasanian Silver Vessel','NULL','1598','700 to 722','Silver')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('3964','Nymph of the Luo River','Gu Kaizhi','1111','317 to 420','Painting')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('3097','The Hunt of the Unicorn','NULL','3543','1680','Tapestries')

SELECT \* FROM DUAL;

SELECT \* FROM ORIGIN;

SELECT \* FROM ART;

/\*

DELETE

FROM

ORIGIN

WHERE

LOCATION = 'China';

\*/

SELECT COUNT(Item)

FROM ART;

DELETE

FROM ART;

INSERT ALL

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('9182','The Thinker','Auguste Rodin','3543','1902','Statuary')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('6922','Nine Dragons','Chen Rong','1111','1244','Painting')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('2049','Old Plum','Kano Sansetsu','6943','1646','Painting')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('2038','Put Down Your Whip','Xu Beihong','3543','1939','Painting')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('3964','The Great Wave off Kanagawa','Hokusai','6943','1829 to 1832','Woodblock Print')

INTO ART (Item,Name,Artist,Origin\_ID,Dating,Media) VALUES('3097','Apocalypse Tapestry','Jean Bondol, Nicolas Bataille','3543','1377 to 1382','Tapestries')

SELECT \* FROM DUAL;

SELECT \* FROM ART;

ALTER TABLE ORIGIN

ADD

(

COUNTRY VARCHAR(35),

CITY VARCHAR(35)

);

UPDATE ORIGIN

SET COUNTRY=LOCATION;

ALTER TABLE ORIGIN

DROP COLUMN LOCATION;

SELECT \* FROM ORIGIN;