



AMERICAN INTERNATIONAL UNIVERSITY - BANGLADESH

INTRODUCTION TO DATABASE

FINAL-TERM PROJECT

Section: [H]

Language Used: Oracle SQL and

[MySQL codes also added, you can use any or whichever you like]

Title: Orphanage Management System

Supervised By

IROM

EVALUATION:

INTRODUCTION TO DATABASE [H]

Total Mark :100; Passing Mark:50; Contributes:100%

*Course Teacher(s): ***** IROM ******

A+ (96/+3)

Submitted By:

Group no - 01

1. Viktor
2. Sudipto
3. Tuba
4. Kazi
5. Patrick

Date: August 18, 2021.

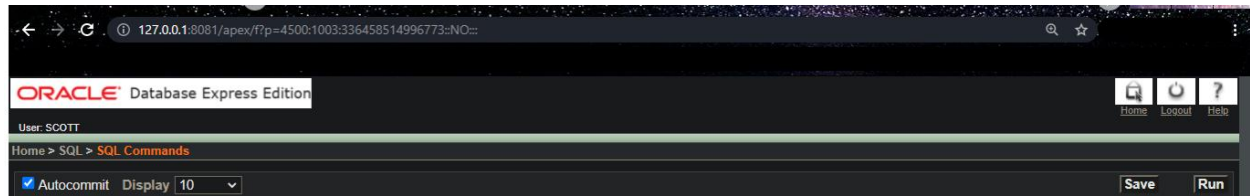
Orphanage Management System

Table of Content:

1. Case study
2. ER diagram
3. Normalization
4. Table creation with constraints
5. Data insertion
6. Query writings based on the tables
7. MySQL Code and example part. As [1 to 6] part is for oracle and from 7 it is Also done in MySQL.

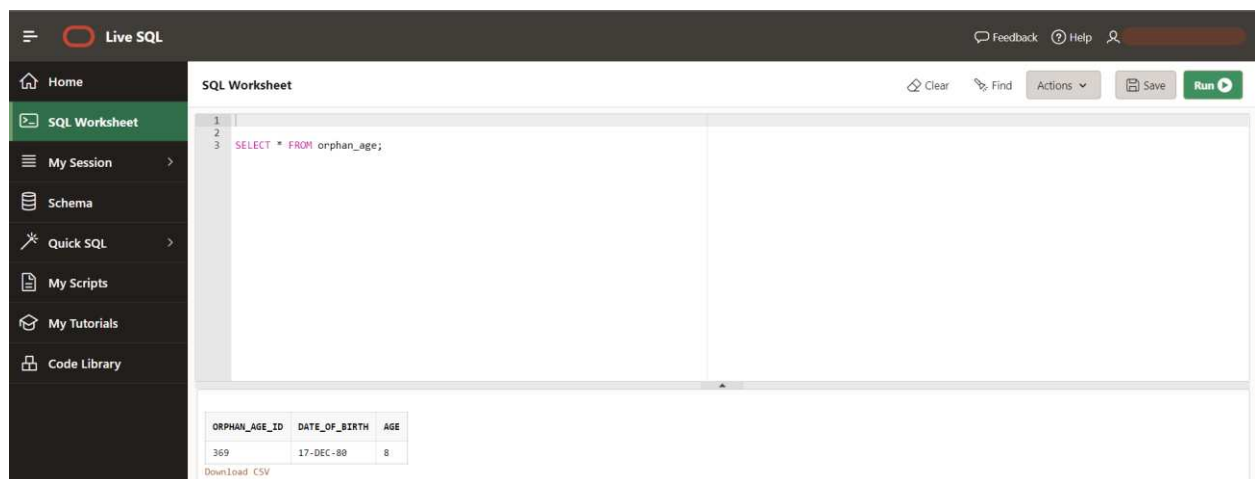
Tools used here:

Oracle10g XE



Alternative:

ORACLE LIVE SQL



ExtendsClass MySQL online



XMAPP with MySQL connection and phpMyAdmin

phpMyAdmin

Server: 127.0.0.1 » Database: project

Structure SQL Search Query Export Import Operations Privileges More

Recent Favorites

New

- information_schema
- mysql
- nibir
- nibir3
- performance_schema
- phpmyadmin
- project
 - New
 - adopter
 - orphan
 - orphan_age
 - owner
 - owner_phone
 - supervisor
- test

SELECT orphan.orphan_id,orphan.orphan_name,orphan_age.date_of_birth,orphan_age.age FROM orphan_age JOIN orphan ON orphan.orphan_id=orphan_age.orphan_ID

[Edit inline] [Edit] [Explain SQL] [Create PHP code]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

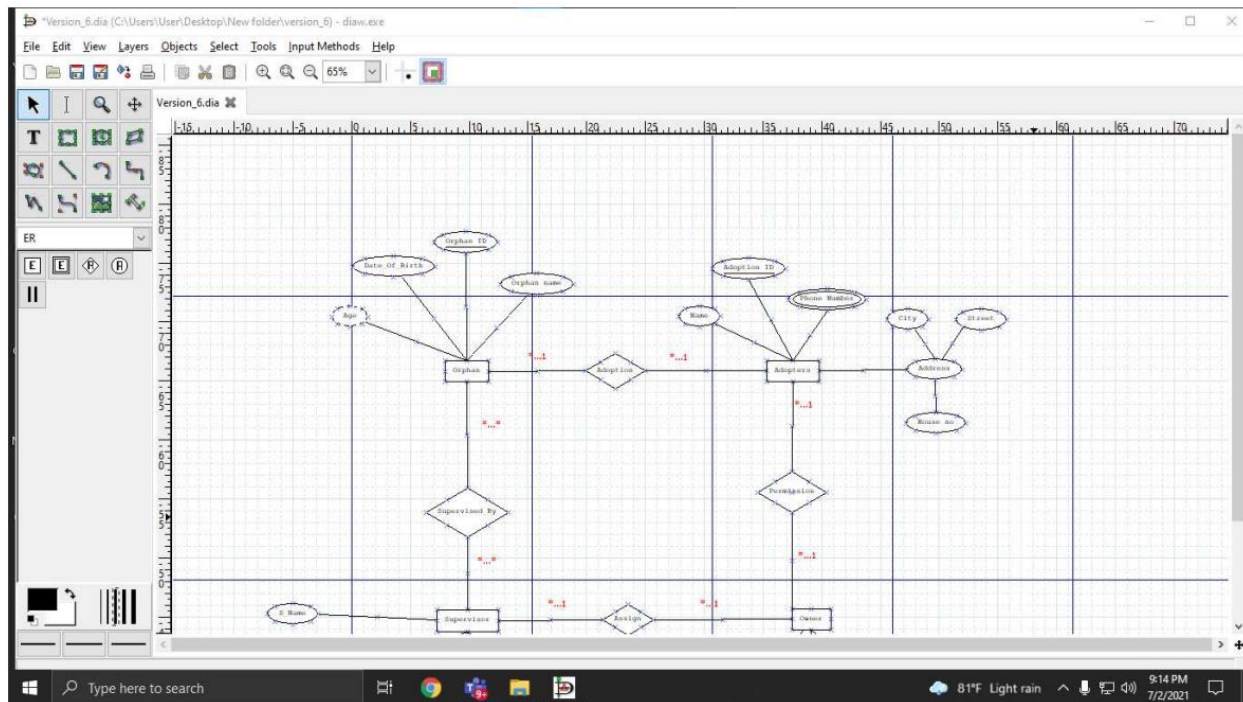
Sort by key: None

+ Options

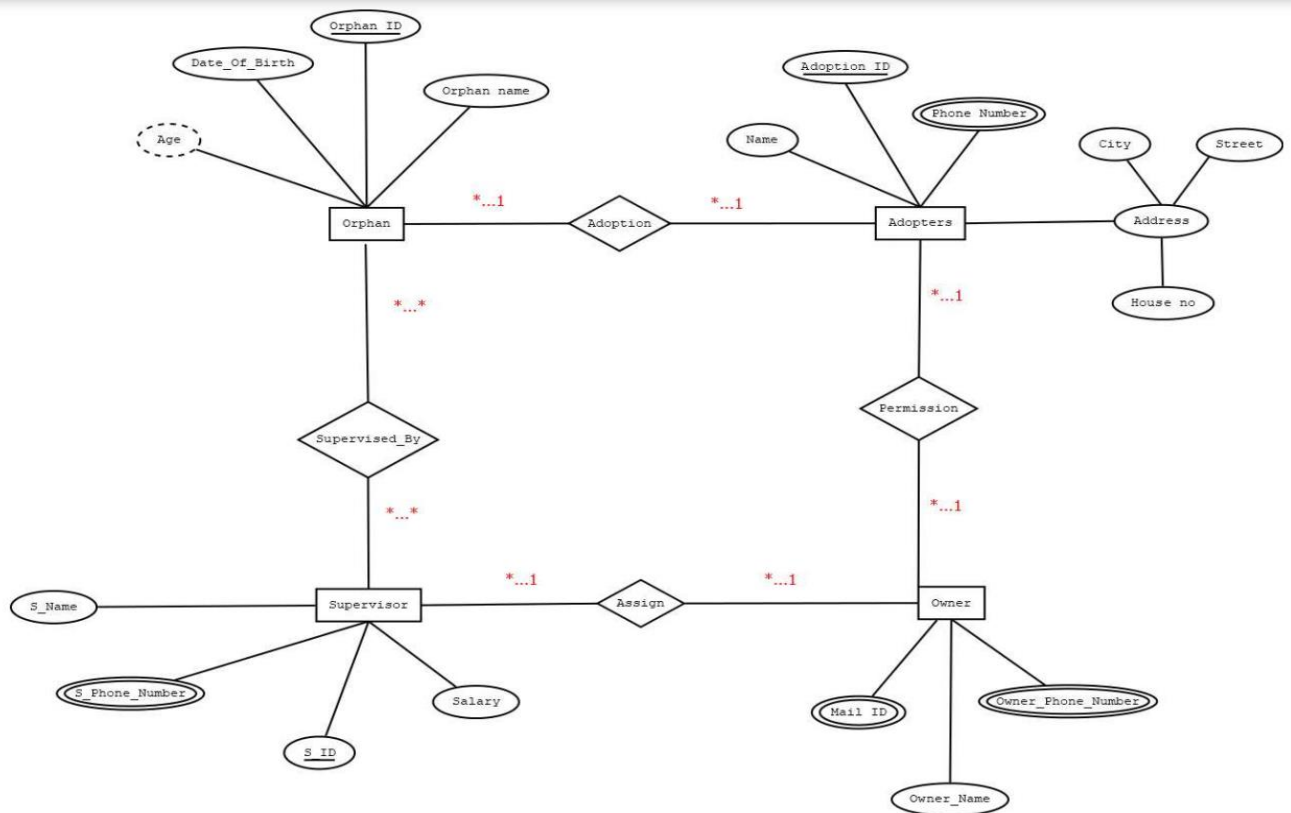
orphan_id	orphan_name	date_of_birth	age
601	Ria Hasan	18-01-2010	11
602	Joya Ahsan	19-02-2011	10
603	Joy Islam	12-03-2012	9
604	Rakib Hasan	13-03-2013	8
605	Emon pasha	14-04-2014	7
606	Musa Aman	18-05-2009	12
607	Robin Rogers	24-07-2014	7
608	Kishor Dutt	14-09-2015	6
609	Kishor Kumar	29-10-2016	5
610	Anupam Roy	20-11-2016	5

ER diagram of MID [Flawed Prototype]

Earliest stage of MID:



Final Stage in MID and used in submission:



Major Flaws:

1. Cardinalities were incorrectly represented.
2. Some attributes have vague names that are not user-friendly.

Update on fixing issues:

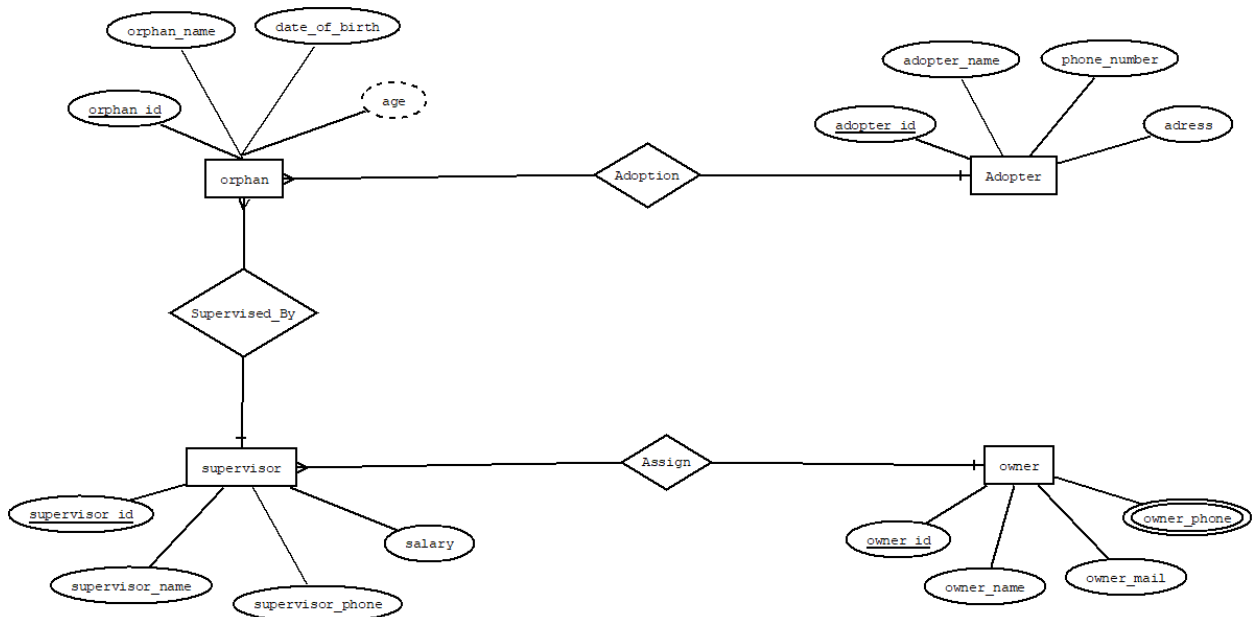
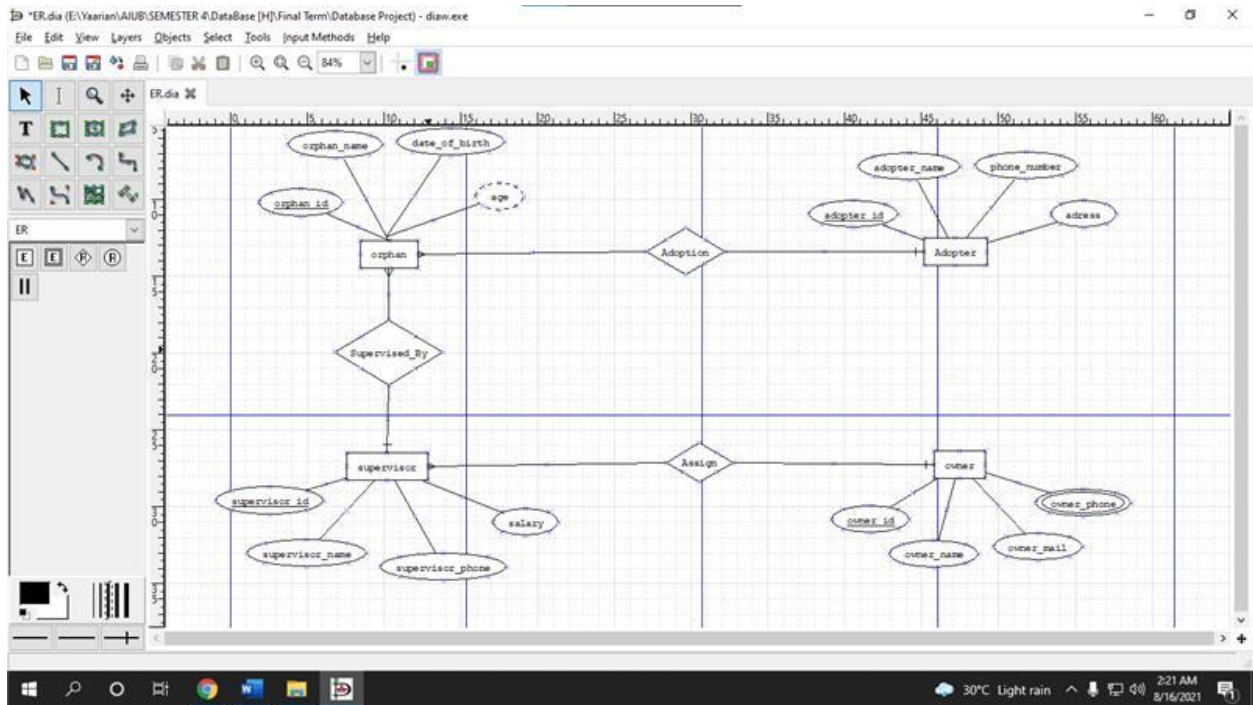
1. Cardinalities are fixed and properly described.
2. All attributes have meaningful names and user-friendly.
3. “permission” relationship is removed due to its additional complexity and entire structure were redesigned for simplified look and better suited for basic relational database system project and it is experimental.
4. Several “multivalued attributes” which are “phone numbers and mails” were removed and shown as simple attributes except in “owner” entity.
Multivalued attributes can generate more complexity when it is dealt in “Normalization” steps. And only owner is having backup numbers due to being the core and overseer of the orphanage.
5. Some attributes removed and structure is shortened and only the most necessary elements were used and utilized.

1. Case Study:

It is a residential facility dedicated to the care of orphans. Orphanages can care for more children than a widow can on her own. Children are frequently transferred to these facilities to relieve a widow of the stress of providing for her family on a low income or where there is no person or group capable of parenting them. These institutions provide a safe environment to grow, a place to sleep, regular meals, and education, as well as frequent visits from their families for the orphans. Most importantly, here children are provided with a steady lifestyle that they might not receive at home. The facility has many objects and related task that needs a formal structure to be meaningful.

The system consists of four entities and each of them having four attributes. Starting from the Owner, The owner group consists of several owners where the attributes here are [owner_id , owner_name , mail, phone]. Owner group is identified uniquely due to the unique ID. Each of them can have multiple phone numbers. Each owner can assign multiple Supervisors and at least one for the job of supervision. The entity Supervisor consists of 4 attributes as [supervisor_id , supervisor_name , supervisor_phone , salary]. Here the multiple orphans are handled by at least one Supervisor. And so as the owner group handles several supervisors to least one. Supervisors also have unique identification. The orphan entity also has four attributes like any other. Which are [orphan_id, orphan_name, date_of_birth, age]. Age is definable from DOB. Adopters can have several orphans to least one. Adopter entity has four attributes. These are [adopter-id, adopter_name, adopter_number, address]. Here the relationships are perfectly symbolized where the structure reflects the scenario of the real world.

2. ER Diagram



3. Normalization

Adoption

All the Attributes are

Adoption (orphan_ID , orphan_name, date_of_birth , age,
adopter_id, adopter_name, adopter_number, address)

1nf: No Multi Valued Attributes

2nf: orphan-ID, orphan_name, date_of_birth , age
adopter-ID, adopter_name, adopter_number, address

3nf: orphan-ID , orphan_name
orphan age id ,date_of_birth , age
adopter-ID, adopter_name, adopter_number, address

So Tables from Adoption are :

orphan-ID , orphan_name
orphan age id , date_of_birth , age
adopter-ID, adopter_name, adopter_number, address

Supervised_By

All the Attributes are

Supervised_By (orphan-ID, orphan_Name, date_of_Birth , age,
supervisor_ID, supervisor _name, supervisor _phone, salary)

1nf: No Multi Valued Attributes

2nf: orphan-ID , orphan_name, date_of_birth , age
 supervisor-id, supervisor _name, supervisor _phone, salary

3nf: orphan-ID , orphan_name
 orphan age id, date_of_birth, Age
 supervisor-id, supervisor_Name, supervisor_Phone, salary

So Tables for the relation of Supervised-by are :

orphan-id , orphan_name
 orphan age id, date_of_birth, age
 supervisor-ID, , supervisor _name, , supervisor _phone, salary

Assign

All the Attributes are

Assign (supervisor_id, supervisor_name, supervisor_phone, salary
 owner_id, owner_name, mail, phone)

1nf : Phone is a multivalued Attribute

2nf : supervisor-id, supervisor_name, supervisor_phone, salary
 owner-id, Owner_name, mail, phone

3nf : No transitive dependencies found

So Tables from Assign relation are :

supervisor-id, supervisor_name, supervisor_phone, salary

owner-id, owner_name, mail

owner-id, phone

Total Tables for the Project:

1. [Table title = orphan]
orphan_id, orphan_name, orphan_age_id, adopter_id, supervisor_id
2. [Table title = orphan_age]
orphan_age_id, date_of_birth, age
3. [Table title = Adopter]
adopter_id, adopter_name, phone_number, address
4. [Table title = owner]
owner_id, owner_name, mail
5. [Table title = owner_phone]
owner_id, owner_phone
6. [Table title = supervisor]
supervisor_id, supervisor_name, supervisor_phone, salary, owner_id

* Bold and underline => Primary_Key

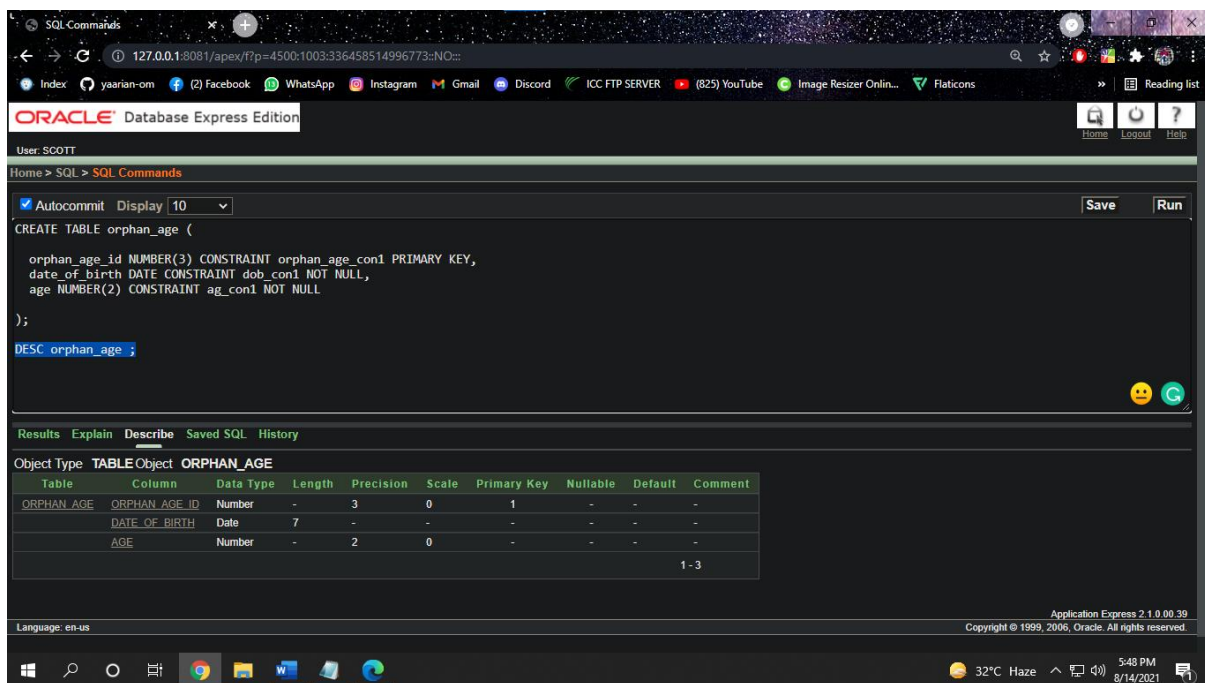
* Red font_color => Composit_Key

* Violet font_color => Foreign_Key

4. Table Creation with constraints

- i. `CREATE TABLE orphan_age`
(
 `orphan_age_id NUMBER(3) CONSTRAINT orphan_age_con1 PRIMARY KEY,`
 `date_of_birth DATE CONSTRAINT dob_con1 NOT NULL,`
 `age NUMBER(2) CONSTRAINT ag_con1 NOT NULL`
);

DESC orphan_age;



- ii. `CREATE TABLE Adopter`
(
 `adopter_id NUMBER(3) CONSTRAINT aid_con1 PRIMARY KEY,`
 `adopter_name VARCHAR2(32) CONSTRAINT anm_con1 NOT NULL,`
 `phone_number VARCHAR(11 CHAR) CONSTRAINT phn_con1 NOT NULL,`
 `address VARCHAR2(50 CHAR) CONSTRAINT adrs_01 NOT NULL,`
 `CONSTRAINT phn_con2 UNIQUE (phone_number)`
);

DESC Adopter;

SQL Commands

127.0.0.1:8081/apex/f?p=4500:1003:336458514996773::NO::

User: SCOTT

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```
CREATE TABLE Adopter (
  adopter_id NUMBER(3) CONSTRAINT aid_con1 PRIMARY KEY,
  adopter_name VARCHAR2(32) CONSTRAINT anm_con1 NOT NULL,
  phone_number VARCHAR(11 CHAR) CONSTRAINT phn_con1 NOT NULL,
  address VARCHAR2(50 CHAR) CONSTRAINT adrs_01 NOT NULL,
  CONSTRAINT phn_con2 UNIQUE (phone_number)
);
```

DESC Adopter ;

Results Explain Describe Saved SQL History

Object Type: TABLE Object: ADOPTER

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ADOPTER	ADOPTER_ID	Number	-	3	0	1	-	-	-
	ADOPTER_NAME	Varchar2	32	-	-	-	-	-	-
	PHONE_NUMBER	Varchar2	11	-	-	-	-	-	-
	ADDRESS	Varchar2	50	-	-	-	-	-	-
									1 - 4

Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

Language: en-us

32°C Haze 5:56 PM 8/14/2021

iii. CREATE TABLE owner

```
(
  owner_id NUMBER(3) CONSTRAINT onwid_con1 PRIMARY KEY,
  owner_name VARCHAR2(10) CONSTRAINT onm_con1 NOT NULL,
  owner_mail VARCHAR(32 CHAR) CONSTRAINT mail_con1 NOT NULL
);
```

DESC owner;

SQL Commands

127.0.0.1:8081/apex/?p=4500:1003:336458514996773:NO::

ORACLE Database Express Edition

User: SCOTT

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```
CREATE TABLE owner (
  owner_id NUMBER(3) CONSTRAINT onw_id_con1 PRIMARY KEY,
  owner_name VARCHAR2(10) CONSTRAINT onm_con1 NOT NULL,
  owner_mail VARCHAR(32 CHAR) CONSTRAINT mail_con1 NOT NULL
);
```

DESC owner;

Results Explain Describe Saved SQL History

Object Type TABLE Object OWNER

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OWNER	OWNER_ID	Number	-	3	0	1	-	-	-
	OWNER_NAME	Varchar2	10	-	-	-	-	-	-
	OWNER_MAIL	Varchar2	32	-	-	-	-	-	-

1 - 3

Language: en-us

Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

31°C Haze 6:30 PM 8/14/2021

iv. CREATE TABLE owner_phone
(
owner_id NUMBER(3) CONSTRAINT FK_owner88 REFERENCES owner ,
owner_phone VARCHAR(11 CHAR) CONSTRAINT phn_con99 NOT NULL,
CONSTRAINT phn_con8 UNIQUE (owner_phone),
CONSTRAINT COMP_KEY09 PRIMARY KEY (owner_id, owner_phone)
);

DESC owner_phone;

SQL Commands

127.0.0.1:8081/apex/?p=4500:1003:336458514996773:NO::

ORACLE Database Express Edition

User: SCOTT

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```
CREATE TABLE owner_phone (
  owner_id NUMBER(3) CONSTRAINT FK_owner88 REFERENCES owner ,
  owner_phone VARCHAR(11 CHAR) CONSTRAINT phn_con99 NOT NULL,
  CONSTRAINT phn_con8 UNIQUE (owner_phone),
  CONSTRAINT COMP_KEY09 PRIMARY KEY (owner_id, owner_phone)
);

DESC owner_phone;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object OWNER_PHONE

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OWNER_PHONE	OWNER_ID	Number	-	3	0	1	-	-	-
OWNER_PHONE	OWNER_PHONE	Varchar2	11	-	-	2	-	-	-

1 - 2

Language: en-us

Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

31°C Haze 6:40 PM 8/14/2021

- v. CREATE TABLE supervisor
(
 supervisor_id NUMBER(3) CONSTRAINT sid_con1 PRIMARY KEY,
 supervisor_name VARCHAR2(32) CONSTRAINT snm_con1 NOT NULL,
 supervisor_phone VARCHAR(11 CHAR) CONSTRAINT phn_con3 NOT NULL,
 salary NUMBER(5) CONSTRAINT sal_con1 NOT NULL,
 owner_id NUMBER(3) CONSTRAINT FK_owner REFERENCES owner,
 CONSTRAINT phn_con4 UNIQUE (supervisor_phone)
);
DESC owner_phone;

SQL Commands

127.0.0.1:8081/apex/?p=4500:1003:336458514996773:NO::

Index yaarian-om Facebook WhatsApp Instagram Gmail Discord ICC FTP SERVER (825) YouTube Image Resizer Onlin... FlatIcons Reading list

ORACLE Database Express Edition

User: SCOTT

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```
CREATE TABLE supervisor (
  supervisor_id NUMBER(3) CONSTRAINT sid_con1 PRIMARY KEY,
  supervisor_name VARCHAR2(32) CONSTRAINT snm_con1 NOT NULL,
  supervisor_phone VARCHAR(11 CHAR) CONSTRAINT phn_con3 NOT NULL,
  salary NUMBER(5) CONSTRAINT sal_con1 NOT NULL,
  owner_id NUMBER(3) CONSTRAINT FK_owner REFERENCES owner,
  CONSTRAINT phn_con4 UNIQUE (supervisor_phone)
);
DESC owner_phone;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object OWNER_PHONE

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OWNER_PHONE	OWNER_ID	Number	-	3	0	1	-	-	-
OWNER_PHONE	OWNER_PHONE	Varchar2	11	-	-	2	-	-	-

1 - 2

Language: en-us

Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

31°C Haze 6:44 PM 8/14/2021

vi. CREATE TABLE orphan
(
 orphan_id NUMBER(5) CONSTRAINT orpid_con1 PRIMARY KEY,
 orphan_name VARCHAR2(32) CONSTRAINT ornm_con1 NOT NULL,
 orphan_age_id NUMBER(3) CONSTRAINT FK_orphan_age REFERENCES orphan_age ,
 adopter_id NUMBER(3) CONSTRAINT FK_adopter REFERENCES adopter ,
 supervisor_id NUMBER(3) CONSTRAINT FK_supervisor REFERENCES supervisor
);
DESC orphan_age ;

SQL Commands

127.0.0.1:8081/apex/?p=4500:1003:336458514996773:NO::

Index yaarian-om (2) Facebook WhatsApp Instagram Gmail Discord ICC FTP SERVER (825) YouTube Image Resizer Onlin... Flaticons

Oracle Database Express Edition

User: SCOTT

Home Logout Help

Home > SQL > SQL Commands

☒ Autocommit Display 10

Save Run

CREATE TABLE orphan (
 orphan_id NUMBER(5) CONSTRAINT orpid_con1 PRIMARY KEY,
 orphan_name VARCHAR2(32) CONSTRAINT ornm_con1 NOT NULL,

 orphan_age_id NUMBER(3) CONSTRAINT FK_orphan_age REFERENCES orphan_age ,
 adopter_id NUMBER(3) CONSTRAINT FK_adopter REFERENCES adopter ,
 supervisor_id NUMBER(3) CONSTRAINT FK_supervisor REFERENCES supervisor
);

DESC orphan_age ;

Results Explain Describe Saved SQL History

Object Type TABLEObject ORPHAN_AGE

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ORPHAN_AGE	ORPHAN_AGE_ID	Number	-	3	0	1	-	-	-
	DATE OF BIRTH	Date	7	-	-	-	-	-	-
	AGE	Number	-	2	0	-	-	-	-
1 - 3									

Language: en-us

Application Express 2.1.0.00.20
Copyright © 1999, 2006, Oracle. All rights reserved.

31°C Haze 6:48 PM 8/14/2021

5. Data insertion:

For orphan_age Table:

```
INSERT INTO orphan_age VALUES
(301,to_date('18-01-2010','dd-mm-yyyy'),11);
INSERT INTO orphan_age VALUES
(302,to_date('19-02-2011','dd-mm-yyyy'),10);
INSERT INTO orphan_age VALUES
(303,to_date('12-03-2012','dd-mm-yyyy'),9);
INSERT INTO orphan_age VALUES
(304,to_date('13-03-2013','dd-mm-yyyy'),8);
INSERT INTO orphan_age VALUES
(305,to_date('14-04-2014','dd-mm-yyyy'),7);
INSERT INTO orphan_age VALUES
(306,to_date('18-05-2009','dd-mm-yyyy'),12);
INSERT INTO orphan_age VALUES
(307,to_date('24-07-2014','dd-mm-yyyy'),7);
INSERT INTO orphan_age VALUES
(308,to_date('14-09-2015','dd-mm-yyyy'),6);
INSERT INTO orphan_age VALUES
(309,to_date('29-10-2016','dd-mm-yyyy'),5);
INSERT INTO orphan_age VALUES
(310,to_date('20-11-2016','dd-mm-yyyy'),5);
select * from orphan_age;
```

Results	Explain	Describe	Saved SQL	History
ORPHAN_AGE_ID	DATE_OF_BIRTH	AGE		
301	18-JAN-10	11		
302	19-FEB-11	10		
303	12-MAR-12	9		
304	13-MAR-13	8		
305	14-APR-14	7		
306	18-MAY-09	12		
307	24-JUL-14	7		
308	14-SEP-15	6		
309	29-OCT-16	5		
310	20-NOV-16	5		
10 rows returned in 0.02 seconds				CSV Export

For Adopter Table:

INSERT INTO Adopter VALUES

(501,'Joey Tribbiany','01866853354','Agrabad, Chittagong');

INSERT INTO Adopter VALUES

(502,'Monica Geller','01588853354','Chowmuhoni, Chittagong');

INSERT INTO Adopter VALUES

(503,'Ross Geller','01599953354','Mogbazar, Dhaka');

INSERT INTO Adopter VALUES

(504,'Pheobe Buffay','01599954454','Uttara, Dhaka');

INSERT INTO Adopter VALUES

(505,'Chandler Bing','01599957654','Bashundhara, Dhaka');

select * from Adopter;

ADOPTER_ID	ADOPTER_NAME	PHONE_NUMBER	ADDRESS
501	Joey Tribbiany	01866853354	Agrabad, Chittagong
502	Monica Geller	01588853354	Chowmuhoni, Chittagong
503	Ross Geller	01599953354	Mogbazar, Dhaka
504	Pheobe Buffay	01599954454	Uttara, Dhaka
505	Chandler Bing	01599957654	Bashundhara, Dhaka

5 rows returned in 0.00 seconds [CSV Export](#)

For Owner Table:

```

INSERT INTO owner VALUES
(101,'Tuba Azad','tubaazad@gmail.com');

INSERT INTO owner VALUES
(102,'Nitu Hasan','nituhasan@gmail.com');

INSERT INTO owner VALUES
(103,'Riad Islam','riadislam@gmail.com');

select * from owner;
```

OWNER_ID	OWNER_NAME	OWNER_MAIL
101	Tuba Azad	tubaazad@gmail.com
102	Nitu Hasan	nituhasan@gmail.com
103	Riad Islam	riadislam@gmail.com

3 rows returned in 0.00 seconds [CSV Export](#)

For Owner_phone Table:

```

INSERT INTO owner_phone VALUES
(101,'01848484848');

INSERT INTO owner_phone VALUES
(101,'01948575747');

INSERT INTO owner_phone VALUES
(102,'01586881818');

INSERT INTO owner_phone VALUES
```

OWNER_ID	OWNER_PHONE
101	01848484848
101	01948575747
102	01586881818
103	01686666666

4 rows returned in 0.05 seconds

```
(103,'01686666666');  
select * from owner_phone;
```

For Supervisor Table:

```
INSERT INTO supervisor VALUES  
(201,'Bashar Kazi','01588594853',12000,101);  
INSERT INTO supervisor VALUES  
(202,'Arafat Islam','01590984950',12000,101);  
INSERT INTO supervisor VALUES  
(203,'Naim Islam','01586881234',13000,102);  
INSERT INTO supervisor VALUES  
(204,'Emon Hasan','01586878987',16000,103);  
INSERT INTO supervisor VALUES  
(205,'karim rahman','01580008354',11000,103);  
select * from supervisor;
```

SUPERVISOR_ID	SUPERVISOR_NAME	SUPERVISOR_PHONE	SALARY	OWNER_ID
201	Bashar Kazi	01588594853	12000	101
202	Arafat Islam	01590984950	12000	101
203	Naim Islam	01586881234	13000	102
204	Emon Hasan	01586878987	16000	103
205	karim rahman	01580008354	11000	103

5 rows returned in 0.00 seconds [CSV Export](#)

For Orphan Table:

```
INSERT INTO orphan VALUES  
(601,'Ria Hasan',301,501,205);  
INSERT INTO orphan VALUES  
(602,'Joya Ahsan',302,501,205);
```

```

INSERT INTO orphan VALUES
(603,'Joy Islam',303,502,203);
INSERT INTO orphan VALUES
(604,'Rakib Hasan',304,502,203);
INSERT INTO orphan VALUES
(605,'Emon pasha',305,503,201);
INSERT INTO orphan VALUES
(606,'Musa Aman',306,503,201);
INSERT INTO orphan VALUES
(607,'Robin Rogers',307,503,202);
INSERT INTO orphan VALUES
(608,'Kishor Dutt',308,504,202);
INSERT INTO orphan VALUES
(609,'Kishor Kumar',309,505,203);
INSERT INTO orphan VALUES
(610,'Anupam Roy',310,505,204);
select * from orphan;

```

ORPHAN_ID	ORPHAN_NAME	ORPHAN_AGE_ID	ADOPTER_ID	SUPERVISOR_ID
601	Ria Hasan	301	501	205
602	Joya Ahsan	302	501	205
603	Joy Islam	303	502	203
604	Rakib Hasan	304	502	203
605	Emon pasha	305	503	201
606	Musa Aman	306	503	201
607	Robin Rogers	307	503	202
608	Kishor Dutt	308	504	202
609	Kishor Kumar	309	505	203
610	Anupam Roy	310	505	204

10 rows returned in 0.00 seconds [CSV Export](#)

6. Query writings based on the tables

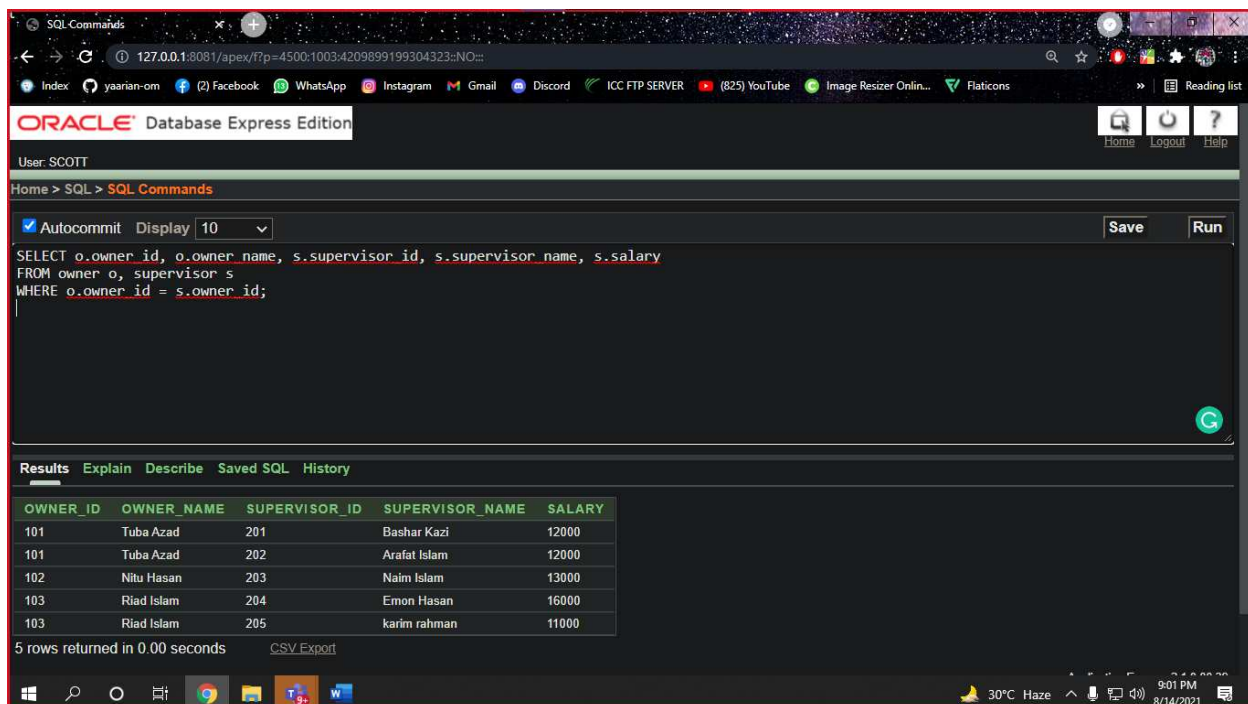
▪ Joining:

Equi-Join

1. Show owner_id, owner_name, supervisor_id, s.supervisor_name, salary

→

```
SELECT o.owner_id, o.owner_name, s.supervisor_id, s.supervisor_name, s.salary
FROM owner o, supervisor s
WHERE o.owner_id = s.owner_id;
```



The screenshot shows the Oracle Database Express Edition interface. The SQL Command window contains the following query:

```
SELECT o.owner_id, o.owner_name, s.supervisor_id, s.supervisor_name, s.salary
FROM owner o, supervisor s
WHERE o.owner_id = s.owner_id;
```

The Results tab shows the following data:

OWNER_ID	OWNER_NAME	SUPERVISOR_ID	SUPERVISOR_NAME	SALARY
101	Tuba Azad	201	Bashar Kazi	12000
101	Tuba Azad	202	Arafat Islam	12000
102	Nitu Hasan	203	Naim Islam	13000
103	Riad Islam	204	Emon Hasan	16000
103	Riad Islam	205	karim rahman	11000

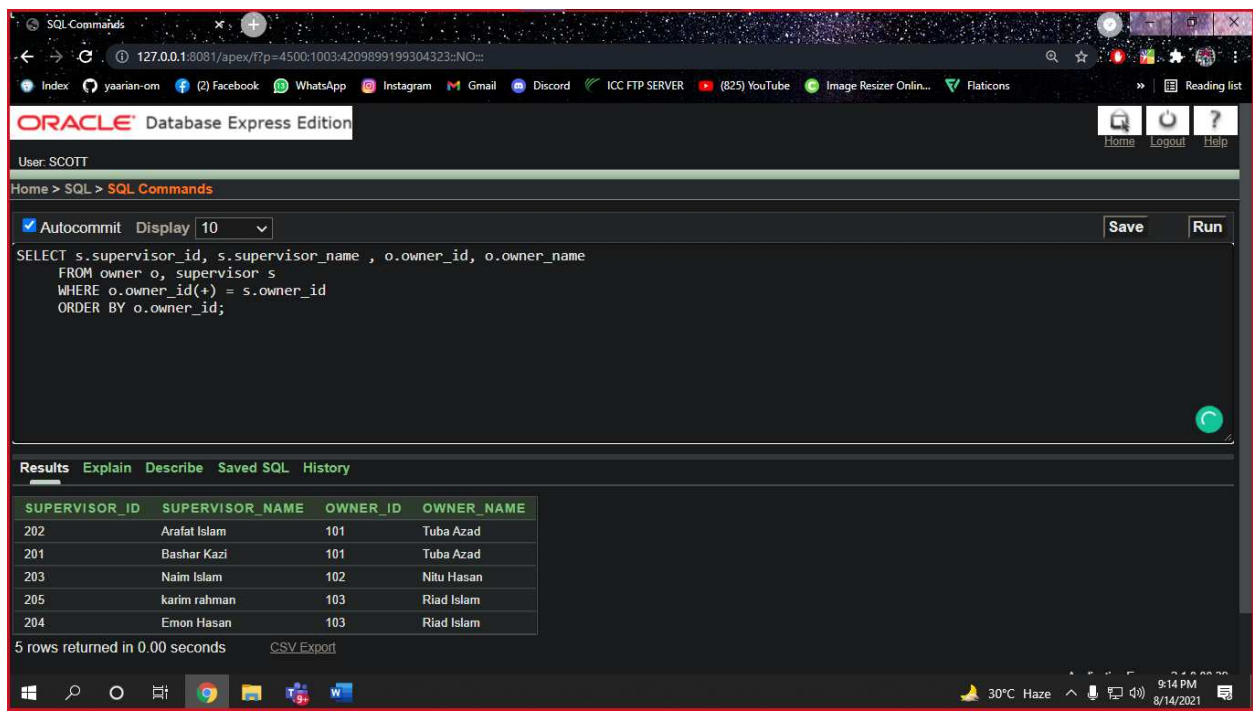
5 rows returned in 0.00 seconds

Outer Join

2. Show all the supervisor_name and salary if null it also be showed

→

```
SELECT s.supervisor_id, s.supervisor_name , o.owner_id, o.owner_name
FROM owner o, supervisor s
WHERE o.owner_id(+) = s.owner_id
ORDER BY o.owner_id;
```



The screenshot shows the Oracle Database Express Edition interface. The SQL Commands window displays the following query:

```
SELECT s.supervisor_id, s.supervisor_name , o.owner_id, o.owner_name
FROM owner o, supervisor s
WHERE o.owner_id(+) = s.owner_id
ORDER BY o.owner_id;
```

The query has been executed, and the Results tab shows the following data:

SUPERVISOR_ID	SUPERVISOR_NAME	OWNER_ID	OWNER_NAME
202	Arafat Islam	101	Tuba Azad
201	Bashar Kazi	101	Tuba Azad
203	Naim Islam	102	Nitu Hasan
205	karim rahman	103	Riad Islam
204	Emon Hasan	103	Riad Islam

5 rows returned in 0.00 seconds. A CSV Export link is available.

Self-Join

3. Display supervisor_id, supervisor_name and salary of those who have same salary .

```
SELECT a.supervisor_id, a.supervisor_name, b.salary
FROM supervisor a, supervisor b
WHERE a.salary = b.salary
AND a.supervisor_name <> b.supervisor_name;
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The query entered is a self-join on the supervisor table, filtering for rows where the salary is the same but the supervisor name is different. The results show two rows: one for supervisor_id 202 (Arafat Islam) and one for supervisor_id 201 (Bashar Kazi), both with a salary of 12000.

Oracle Database Express Edition

User: SCOTT

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```
SELECT a.supervisor_id, a.supervisor_name, b.salary
FROM supervisor a, supervisor b
WHERE a.salary = b.salary
AND a.supervisor_name <> b.supervisor_name;
```

Results Explain Describe Saved SQL History

SUPERVISOR_ID	SUPERVISOR_NAME	SALARY
202	Arafat Islam	12000
201	Bashar Kazi	12000

2 rows returned in 0.00 seconds [CSV Export](#)

Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

Language: en-us

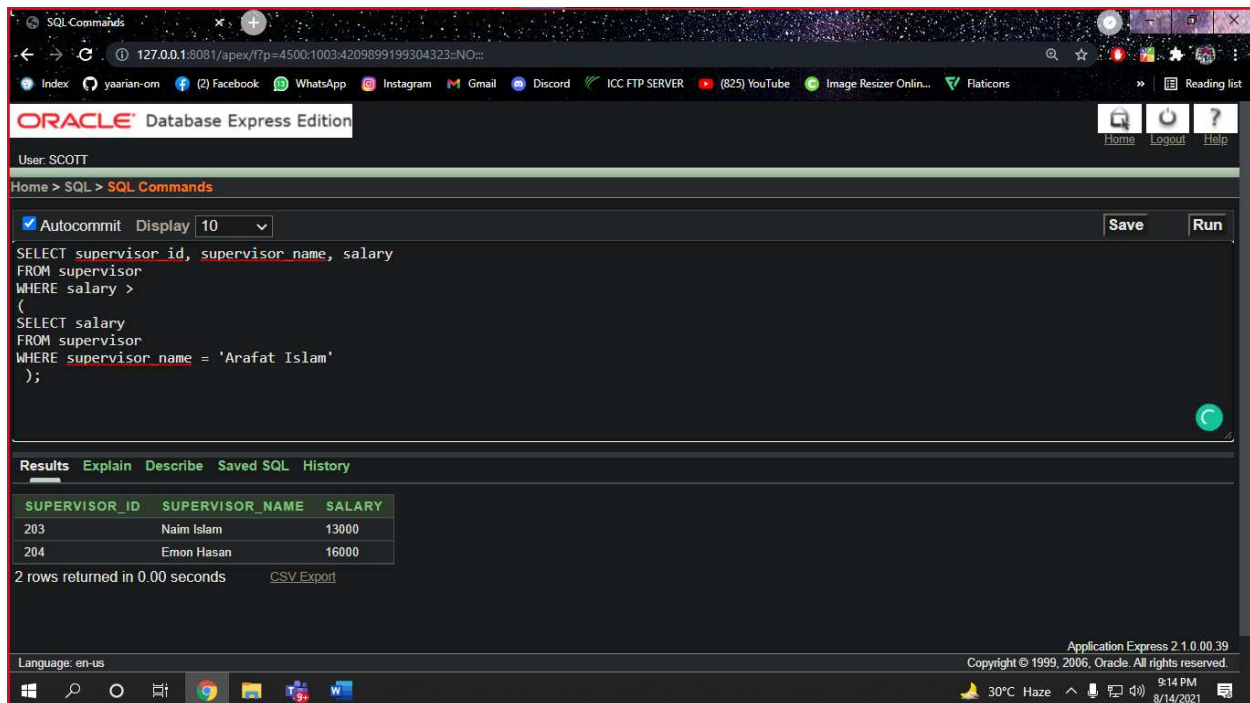
30°C Light rain 2:27 AM 8/16/2021

■ Sub-Query

1. Show all the supervisor_id, supervisor_name, salary who's salary is more than "Arafat Islam".

→

```
SELECT supervisor_id, supervisor_name, salary
FROM supervisor
WHERE salary >
(
  SELECT salary
  FROM supervisor
  WHERE supervisor_name = 'Arafat Islam'
);
```



The screenshot shows the Oracle Database Express Edition interface. The SQL Commands window contains the following query:

```
SELECT supervisor_id, supervisor_name, salary
FROM supervisor
WHERE salary >
(
  SELECT salary
  FROM supervisor
  WHERE supervisor_name = 'Arafat Islam'
);
```

The Results tab shows the following data:

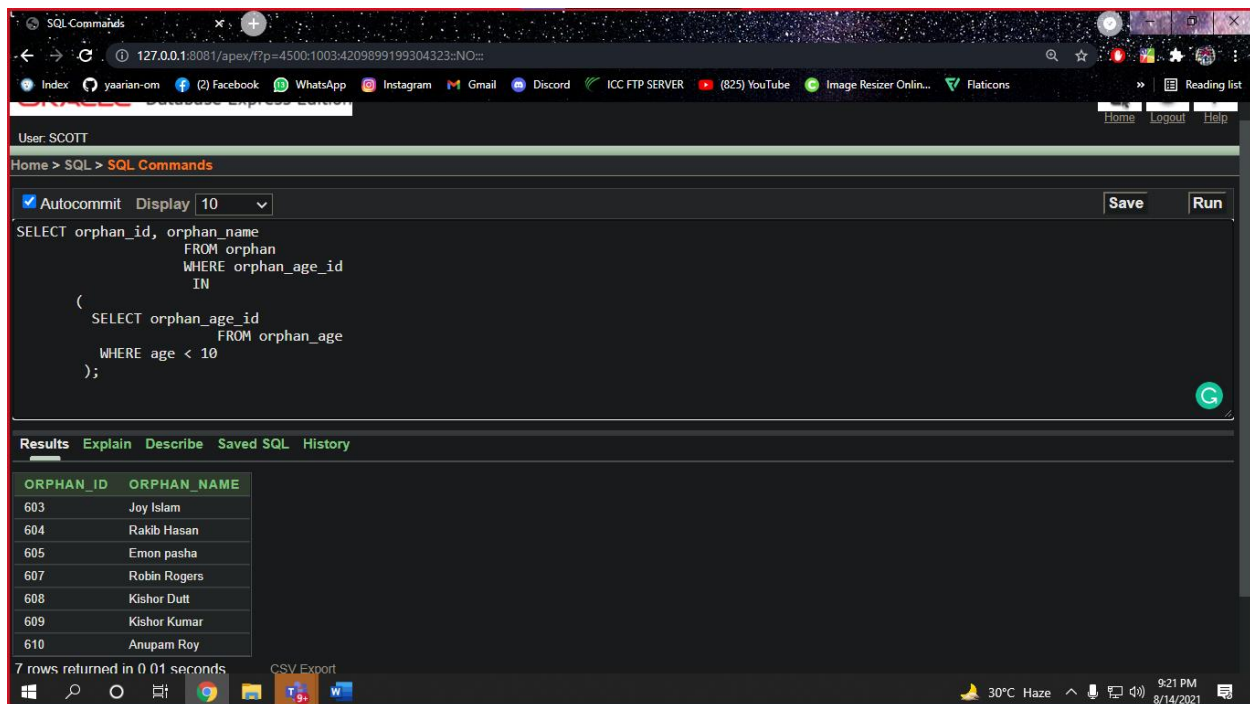
SUPERVISOR_ID	SUPERVISOR_NAME	SALARY
203	Naim Islam	13000
204	Emon Hasan	16000

2 rows returned in 0.00 seconds

Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.
9:14 PM 8/14/2021

2. Show orphan_id, orphan_name who's age is less than 10.

```
→  
SELECT orphan_id, orphan_name  
FROM orphan  
WHERE orphan_age_id  
IN  
(  
    SELECT orphan_age_id  
    FROM orphan_age  
    WHERE age < 10  
);
```



The screenshot shows a web-based SQL Command Line interface. The browser address bar displays the URL `127.0.0.1:8081/apex/?p=4500:1003:4209899199304323:NO::`. The interface includes a navigation bar with links for Home, Logout, and Help. Below the navigation bar, the user is identified as "User: SCOTT". The main area is titled "SQL Commands" and contains a text input field with the following SQL query:

```
SELECT orphan_id, orphan_name  
FROM orphan  
WHERE orphan_age_id  
IN  
(  
    SELECT orphan_age_id  
    FROM orphan_age  
    WHERE age < 10  
);
```

Buttons for "Autocommit", "Display", "10", "Save", and "Run" are visible above the query input. Below the query input, there are tabs for "Results", "Explain", "Describe", "Saved SQL", and "History". The "Results" tab is active, showing a table with two columns: "ORPHAN_ID" and "ORPHAN_NAME". The table contains seven rows of data:

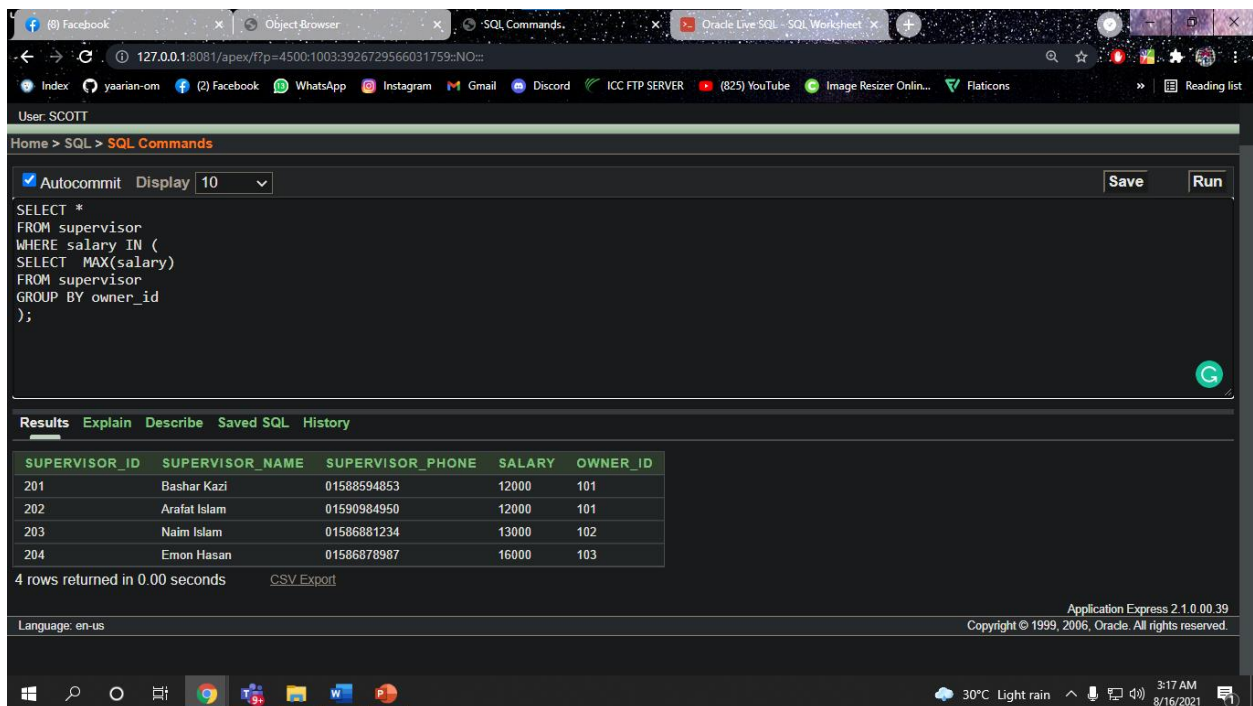
ORPHAN_ID	ORPHAN_NAME
603	Joy Islam
604	Rakib Hasan
605	Emon pasha
607	Robin Rogers
608	Kishor Dutt
609	Kishor Kumar
610	Anupam Roy

At the bottom of the results section, it states "7 rows returned in 0.01 seconds". A "CSV Export" button is also present. The Windows taskbar at the bottom shows the system time as 9:21 PM on 8/14/2021, with a weather widget indicating 30°C Haze.

Sub-query (Group function):

3. Write a query to display highest salary owner wise and show name of supervisor who's getting the highest salary.

```
→ SELECT *  
FROM supervisor  
WHERE salary IN  
(  
SELECT MAX(salary)  
FROM supervisor  
GROUP BY owner_id  
);
```



The screenshot shows a web browser window with multiple tabs. The active tab is 'Oracle Live SQL - SQL Worksheet'. The URL bar shows '127.0.0.1:8081/apex/f?p=4500:1003:3926729566031759::NO::'. The browser's address bar and tabs are visible at the top. Below the browser window, there is a dark-themed interface for the 'SQL Commands' window. The 'Autocommit' checkbox is checked, and the 'Display' dropdown is set to '10'. The 'Save' and 'Run' buttons are visible. The SQL command entered is:

```
SELECT *  
FROM supervisor  
WHERE salary IN (  
SELECT MAX(salary)  
FROM supervisor  
GROUP BY owner_id  
);
```

 The 'Results' tab is selected, showing a table with 4 rows and 5 columns: SUPERVISOR_ID, SUPERVISOR_NAME, SUPERVISOR_PHONE, SALARY, and OWNER_ID. The data is as follows:

SUPERVISOR_ID	SUPERVISOR_NAME	SUPERVISOR_PHONE	SALARY	OWNER_ID
201	Bashar Kazi	01588594853	12000	101
202	Arafat Islam	01590984950	12000	101
203	Naim Islam	01586881234	13000	102
204	Eman Hasan	01586878987	16000	103

 Below the table, it says '4 rows returned in 0.00 seconds' and 'CSV Export'. The bottom of the window shows the 'Application Express 2.1.0.00.39' version and 'Copyright © 1999, 2006, Oracle. All rights reserved.' The Windows taskbar is visible at the bottom with the date '8/16/2021' and time '3:17 AM'.

4. Write a query to display average salary owner wise and show name of supervisor who's getting the average salary.

```
→ SELECT *  
FROM supervisor  
WHERE salary > all  
(  
SELECT avg(salary)  
FROM supervisor GROUP BY owner_id  
);
```

User: SCOTT

Home > SQL > SQL Commands

☒ Autocommit Display 10 Save Run

```
SELECT *
FROM supervisor
WHERE salary > all
(
SELECT avg(salary)
FROM supervisor GROUP BY owner_id
);
```

Results Explain Describe Saved SQL History

SUPERVISOR_ID	SUPERVISOR_NAME	SUPERVISOR_PHONE	SALARY	OWNER_ID
204	Emon Hasan	01586878987	16000	103

1 rows returned in 0.00 seconds [CSV Export](#)

Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

■ View

Create a complex view with name Complex_View and it should contain owner_id, owner_name, supervisor_id, supervisor_name.

→

```
CREATE VIEW Complex_View(Owner_ID, Owner_Name, Supervisor_ID, Supervisor_Name)
AS SELECT o.owner_id, o.owner_name, s.supervisor_id, s.supervisor_name
FROM owner o, supervisor s
WHERE o.owner_id = s.owner_id;

SELECT * FROM Complex_View;
```

Oracle Live SQL - SQL Worksheet

livesql.oracle.com/apex/?p=590:1-9073437677253:NO:RP::

Index yaarian-om (2) Facebook WhatsApp Instagram Gmail Discord ICC FTP SERVER (825) YouTube Image Resizer Online... FlatIcons Reading list

Live SQL Feedback Help dip.kumar020@gmail.com

SQL Worksheet Clear Find Actions Save Run

```

1 CREATE VIEW Complex_View(Owner_ID, Owner_Name, Supervisor_ID, Supervisor_Name)
2 AS SELECT o.owner_id, o.owner_name, s.supervisor_id, s.supervisor_name
3 FROM owner o, supervisor s
4 WHERE o.owner_id = s.supervisor_id;
5 SELECT * FROM Complex_View;
6

```

OWNER_ID	OWNER_NAME	SUPERVISOR_ID	SUPERVISOR_NAME
101	Tuba Azad	201	Bashar Kazi
101	Tuba Azad	202	Arafat Islam
102	Nitu Hasan	203	Naim Islam
103	Riad Islam	204	Emon Hasan
103	Riad Islam	205	karim rahman

© 2021 Oracle Corporation · Privacy · Terms of Use
 Oracle Learning Library · Ask Tom · Dev Gym · Database Documentation · Follow on Twitter
 Live SQL 21.2.1, running Oracle Database 19c Enterprise Edition - 19.8.0.0.0
 Built with using Oracle APEX running on Oracle Cloud Infrastructure and Oracle Kubernetes Engine

■ Constraints

A Constraint is added to the supervisor Table using “alter” statement.

ALTER TABLE supervisor

ADD CHECK (salary>=10000);

SQL Commands

127.0.0.1:8081/apex/?p=4500:1003:63537172:10049061:NO::

ORACLE Database Express Edition

User SCOTT

Home > SQL > SQL Commands

Autocommit Display 10 Save Run

```

ALTER TABLE supervisor
ADD CHECK (salary>=10000);

```

Results Explain Describe Saved SQL History

Table altered.

0.07 seconds

Language: en-us Application Express 2.1.0.0.39 Copyright © 1999, 2006, Oracle. All rights reserved.

28°C Light rain 5:42 PM 8/17/2021

7. MySQL code part:

```
/*  
CREATE TABLE orphan_age (  
  
    orphan_age_id varchar(3) ,  
    date_of_birth varchar(15),  
    age int(2),  
    PRIMARY KEY (orphan_age_id)  
  
);
```

```
INSERT INTO orphan_age VALUES  
(301,'18-01-2010',11),  
(302,'19-02-2011',10),  
(303,'12-03-2012',9),  
(304,'13-03-2013',8),  
(305,'14-04-2014',7),  
(306,'18-05-2009',12),  
(307,'24-07-2014',7),  
(308,'14-09-2015',6),  
(309,'29-10-2016',5),  
(310,'20-11-2016',5);  
  
select * from orphan_age;
```

```
CREATE TABLE Adopter (  
  
    adopter_id varchar(3) ,  
    adopter_name VARCHAR(32),  
    phone_number VARCHAR(13) ,
```

```
address VARCHAR(50) ,  
PRIMARY KEY(adopter_id)
```

```
);
```

```
INSERT INTO Adopter VALUES
```

```
(501,'Joey Tribbiany','01866853354','Agrabad, Chittagong');
```

```
INSERT INTO Adopter VALUES
```

```
(502,'Monica Geller','01588853354','Chowmuhoni, Chittagong');
```

```
INSERT INTO Adopter VALUES
```

```
(503,'Ross Geller','01599953354','Mogbazar, Dhaka');
```

```
INSERT INTO Adopter VALUES
```

```
(504,'Pheobe Buffay','01599954454','Uttara, Dhaka');
```

```
INSERT INTO Adopter VALUES
```

```
(505,'Chandler Bing','01599957654','Bashundhara, Dhaka');
```

```
select * from Adopter;
```

```
CREATE TABLE owner (
```

```
owner_id varchar(3),
```

```
owner_name VARCHAR(10) ,
```

```
owner_mail VARCHAR(32),
```

```
PRIMARY KEY(owner_id)
```

```
);
```

```
INSERT INTO owner VALUES
```

```
(101,'Tuba Azad','tubaazad@gmail.com');
```

```
INSERT INTO owner VALUES
```

```
(102,'Nitu Hasan','nituhasan@gmail.com');
```

```
INSERT INTO owner VALUES
```



```
(103,'Riad Islam','riadislam@gmail.com');
```

```
select * from owner;
```

```
CREATE TABLE supervisor (  
  supervisor_id VARCHAR(3) ,  
  supervisor_name VARCHAR(32) ,  
  supervisor_phone VARCHAR(11 ) ,  
  salary VARCHAR(5) ,  
  owner_id VARCHAR(3),  
  PRIMARY KEY(supervisor_id)  
);
```

```
INSERT INTO supervisor VALUES  
(201,'Bashar Kazi','01588594853',12000,101);  
INSERT INTO supervisor VALUES  
(202,'Arafat Islam','01590984950',12000,101);  
INSERT INTO supervisor VALUES  
(203,'Naim Islam','01586881234',13000,102);  
INSERT INTO supervisor VALUES  
(204,'Emon Hasan','01586878987',16000,103);  
INSERT INTO supervisor VALUES  
(205,'karim rahman','01580008354',11000,103);
```

```
select * from supervisor;
```

```
CREATE TABLE orphan (  
  
  orphan_id VARCHAR(5) ,
```

```
orphan_name VARCHAR(32) ,  
PRIMARY KEY( orphan_id )  
);
```

```
INSERT INTO orphan VALUES  
(601,'Ria Hasan');
```

```
INSERT INTO orphan VALUES  
(602,'Joya Ahsan');
```

```
INSERT INTO orphan VALUES  
(603,'Joy Islam');
```

```
INSERT INTO orphan VALUES  
(604,'Rakib Hasan');
```

```
INSERT INTO orphan VALUES  
(605,'Emon pasha');
```

```
INSERT INTO orphan VALUES  
(606,'Musa Aman');
```

```
INSERT INTO orphan VALUES  
(607,'Robin Rogers');
```

```
INSERT INTO orphan VALUES  
(608,'Kishor Dutt');
```

```
INSERT INTO orphan VALUES  
(609,'Kishor Kumar');
```

```
INSERT INTO orphan VALUES  
(610,'Anupam Roy');
```

```
select * from orphan;
```

```
CREATE TABLE owner_phone (
```

```
owner_id VARCHAR(3) ,
```

```
owner_phone VARCHAR(11 )  
);
```

```
INSERT INTO owner_phone VALUES  
(101,'01848484848');
```

```
INSERT INTO owner_phone VALUES  
(101,'01948575747');
```

```
INSERT INTO owner_phone VALUES  
(102,'01586881818');
```

```
INSERT INTO owner_phone VALUES  
(103,'01686666666');
```

```
select * from owner_phone;  
*/
```

EXAMPLE Queries:

Database ▾ Queries ▾ Export ▾ Run ▶

```
1 SELECT orphan.orphan_id,orphan.orphan_name,orphan_age.date_of_birth,orphan_age.age
2 FROM orphan_age
3 JOIN orphan
4 ON orphan.orphan_id=orphan.orphan_id;
5
6 /* SELF JOIN */
```

100 rows

orphan_id	orphan_name	date_of_birth	age
601	Ria Hasan	20-11-2016	5
601	Ria Hasan	29-10-2016	5
601	Ria Hasan	14-09-2015	6
601	Ria Hasan	24-07-2014	7
601	Ria Hasan	18-05-2009	12

/*

SELECT

orphan.orphan_id,orphan.orphan_name,orphan_age.date_of_birth,orphan_age.age

FROM orphan_age

JOIN orphan

ON orphan.orphan_id=orphan.orphan_id;

*/

```
1 SELECT *
2 FROM orphan_age, orphan, supervisor, owner, Adopter
3
4 ;
```

7500 rows

orphan_age_id	date_of_birth	age	orphan_id	orphan_name	supervisor_id	supervisor_name	supervisor_phone	salary	owner_id
310	20-11-2016	5	601	Ria Hasan	205	karim rahman	01580008354	11000	101

[Simply attaching tables]

/*

SELECT *

FROM orphan_age, orphan, supervisor, owner, Adopter

;

*/

[MySQL part was done by Patrick]

[Oracle SQL part is evaluated perfectly; MySQL part is not. But tests show its working state.]

THANK YOU