

SDEV 350 Lab 2

Student: Patrick Walsh

Date: 3/23/2021

Course: SDEV 350 7380

Professor: Reginald Haseltine

1. Write and test a set of SQL statements that will drop the following tables:
 - a. Engineers
 - b. Faculty
 - c. Classes
 - d. ClassEnrollments

Oracle SQL Developer : C:\Users\pwals\h\Desktop\School\UMGC\SDEV 350\Week 2\Lab 2.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

Oracle Connections

SDEV350 Connection

Tables (Filtered)

- CLASSENROLLMENTS
- CLASSES
- ENGINEERS
- FACULTY

Views

- COOL_VIEW

Indexes

Packages

Procedures

Functions

Operators

Queues

Queues Tables

Triggers

Types

Sequences

Materialized Views

Reports

All Reports

- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

Welcome Page x SDEV350 Connection x Lab 2.sql x

SQL Worksheet History

Worksheet Query Builder

```
1 /*
2 1. Write and test a set of SQL statements that will drop the following tables:
3 a. Engineers
4 b. Faculty
5 c. Classes
6 d. ClassEnrollments
7
8 Use CASCADE CONSTRAINTS to ensure table is dropped even if there are constraints
9 */
10 DROP TABLE Engineers CASCADE CONSTRAINTS;
11 DROP TABLE Faculty CASCADE CONSTRAINTS;
12 DROP TABLE Classes CASCADE CONSTRAINTS;
13 DROP TABLE ClassEnrollments CASCADE CONSTRAINTS;
14 DROP SEQUENCE seq_eng;
15 DROP SEQUENCE seq_fac;
16 DROP SEQUENCE seq_class;
17 DROP SEQUENCE seq_enroll;
18 DROP VIEW cool_view;
19
```

Script Output x Query Result x

Task completed in 2.585 seconds

Table CLASSENROLLMENTS dropped.

Sequence SEQ_ENG dropped.

Sequence SEQ_FAC dropped.

Sequence SEQ_CLASS dropped.

Sequence SEQ_ENROLL dropped.

2. Write and test a set of SQL statements that will create the following tables:

a. Engineers

Primary Key: EID

Columns: Lastname, Firstname, Email, Graddate

b. Faculty

Primary Key: FID

Columns: Lastname, Firstname, Email, Hiredate

c. Classes

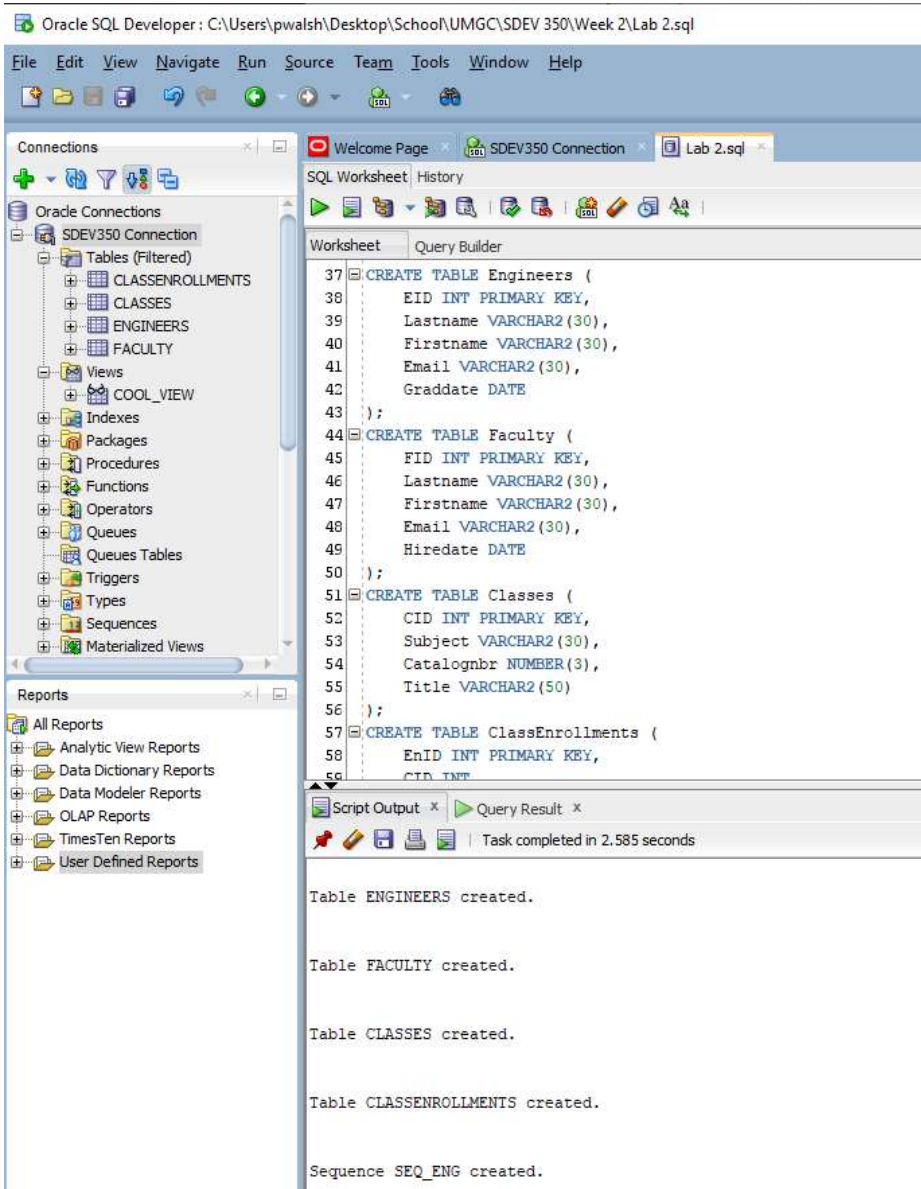
Primary Key: CID

Columns: Subject (e.g. SDEV), Catalognbr (e.g. 350), Title (e.g Database Security)

d. ClassEnrollments

Primary Key: EnID

Foreign Keys: CID (from Classes), FID (from Faculty), EID (from Engineers)



3. Write and test a set of SQL statements that will insert the following quantity of records into each table
- 15 Engineers
 - 3 Faculty

- 3 Classes
- 15 ClassEnrollments

Oracle SQL Developer: C:\Users\pwalsh\Desktop\Schoo\UMGC\SDEV 350\Week 2\Lab 2.sql

The screenshot displays the Oracle SQL Developer interface. The left pane shows the 'Connections' tree with 'SDEV350 Connection' selected. The main window shows a SQL worksheet with the following code:

```
79 Primary Key ID is Okay for this exercise.
80 */
81 CREATE SEQUENCE seq_eng; -- allows FK to be auto-incremented
82 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Halpert', 'Jim', 'jhalpert@test.com', '12 Mar 2012');
83 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Scott', 'Michael', 'mscott@test.com', '15 May 2010');
84 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Beesley', 'Pam', 'pbeesley@test.com', '1 Jun 2018');
85 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Packer', 'Todd', 'tpacker@test.com', '6 Sep 2011');
86 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Kapoor', 'Kelly', 'kkapoor@test.com', '8 Dec 2002');
87 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Flenderson', 'Toby', 'tflenderson@test.com', '10 Nov 1997');
88 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Philbin', 'Darryl', 'dphilbin@test.com', '11 Dec 2020');
89 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Shrute', 'Dwight', 'dshrute@test.com', '11 Jun 2003');
90 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Howard', 'Ryan', 'rhoward@test.com', '3 Nov 2009');
91 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Hudson', 'Stanley', 'shudson@test.com', '19 Mar 2010');
92 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Bernard', 'Andy', 'abernard@test.com', '26 Sep 2002');
93 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Malone', 'Kevin', 'kmalone@test.com', '21 Mar 1986');
94 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Martinez', 'Oscar', 'omartinez@test.com', '2 Aug 1995');
95 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Martin', 'Angela', 'amartin@test.com', '4 Jun 1999');
96 INSERT INTO Engineers(EID, Lastname, Firstname, Email, Graddate) VALUES(seq_eng.NEXTVAL, 'Lapin', 'Phyllis', 'plapin@test.com', '7 Feb 2017');
97 --SELECT * FROM Engineers;
98
99 CREATE SEQUENCE seq_fac; -- allows FK to be auto-incremented
100 INSERT INTO Faculty(FID, Lastname, Firstname, Email, Hiredate) VALUES(seq_fac.NEXTVAL, 'Wallace', 'David', 'dwallace@test.com', '2 Jul 2013');
101 INSERT INTO Faculty(FID, Lastname, Firstname, Email, Hiredate) VALUES(seq_fac.NEXTVAL, 'Jan', 'Laloussan', 'ljaloussan@test.com', '15 Oct 2007');
```

The bottom pane shows the 'Script Output' window with the following results:

```
Sequence SEQ_ENG created.

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.
```


4. Write and test a set of SQL statements that will select **all** records from **each** table. The output should display the records in **descending** order by Primary key.

Oracle SQL Developer : C:\Users\pwalsh\Desktop\School\UMGC\SDEV 350\Week 2\Lab 2.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

Oracle Connections

SDEV350 Connection

Tables (Filtered)

- CLASSENROLLMENTS
- CLASSES
- ENGINEERS
- FACULTY

Views

- COOL_VIEW

Indexes

Packages

Procedures

Functions

Operators

Queues

Queues Tables

Triggers

Types

Sequences

Materialized Views

Reports

- All Reports
- Analytic View Reports
- Data Dictionary Reports
- Data Modeler Reports
- OLAP Reports
- TimesTen Reports
- User Defined Reports

SQL Worksheet History

Worksheet Query Builder

```
133 4. Write and test a set of SQL statements that will select all records from
134 each table. The output should display the records in descending order
135 by Primary key.
136 */
137 SELECT * FROM Engineers ORDER BY EID DESC;
138 SELECT * FROM Faculty ORDER BY FID DESC;
139 SELECT * FROM Classes ORDER BY CID DESC;
140 SELECT * FROM ClassEnrollments ORDER BY EnID DESC;
141
```

Script Output x Query Result x

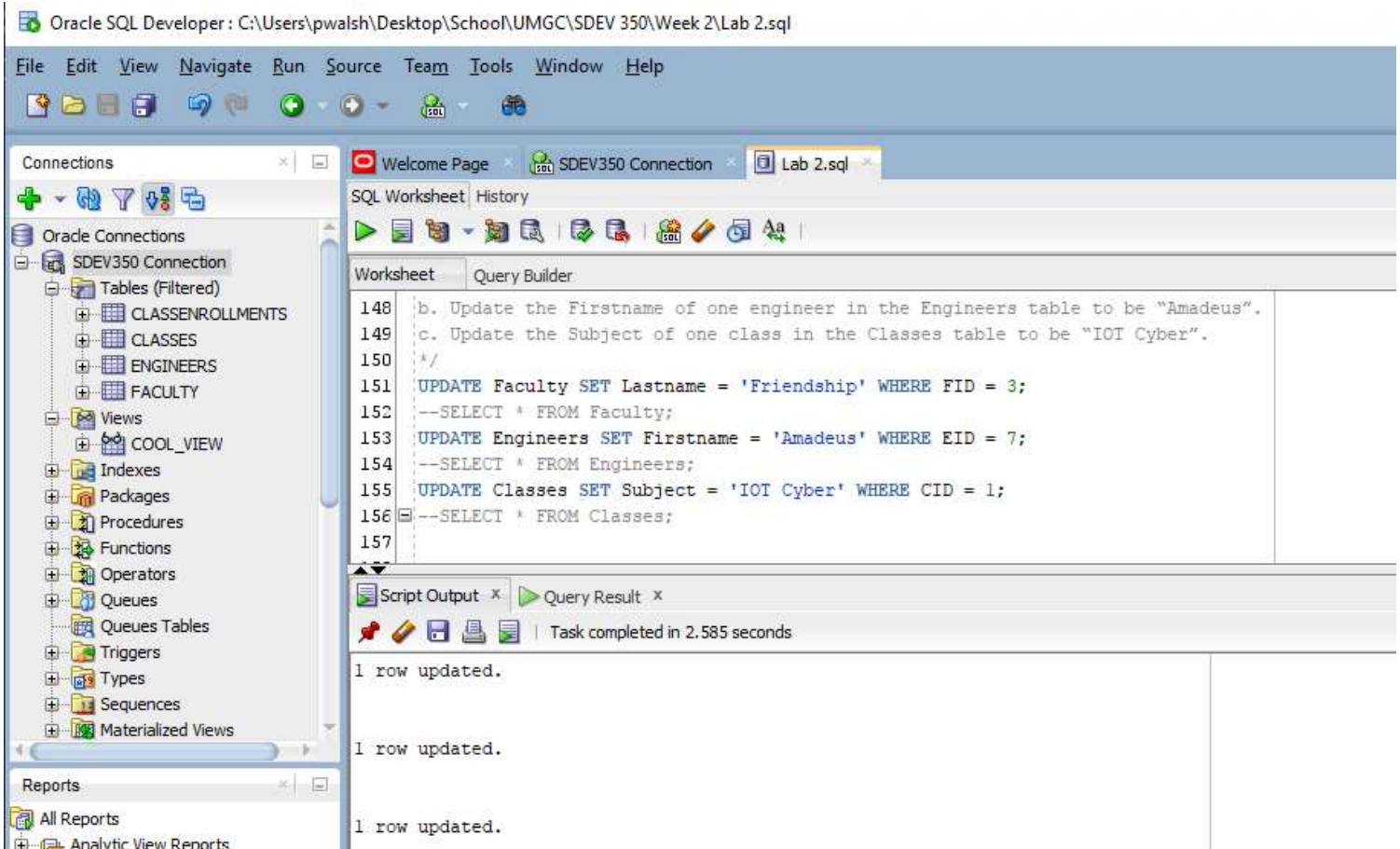
Task completed in 2.585 seconds

EID	LASTNAME	FIRSTNAME	EMAIL	GRADDATE
15	Lapin	Phyllis	plapin@test.com	07-FEB-17
14	Martin	Angela	amartin@test.com	04-JUN-99
13	Martinez	Oscar	omartinez@test.com	02-AUG-95
12	Malone	Kevin	kmalone@test.com	21-MAR-86
11	Bernard	Andy	abernard@test.com	26-SEP-02
10	Hudson	Stanley	shudson@test.com	19-MAR-10
9	Howard	Ryan	rhoward@test.com	03-NOV-09
8	Shrute	Dwight	dshrute@test.com	11-JUN-03
7	Philbin	Darryl	dphilbin@test.com	11-DEC-20
6	Flenderson	Toby	tflenderson@test.com	10-NOV-97
5	Kapoor	Kelly	kkapoor@test.com	08-DEC-02

15 rows selected.

FID	LASTNAME	FIRSTNAME	EMAIL	HIREDATE
3	California	Robert	rcalifornia@test.com	06-JUL-08
2	Levensen	Jan	jlevonsen@test.com	15-OCT-07

5. Write and test a set of SQL statements that will Update records with the following specifications
- Update the Lastname of one faculty in the Faculty table to be "Friendship".
 - Update the Firstname of one engineer in the Engineers table to be "Amadeus".
 - Update the Subject of one class in the Classes table to be "IOT Cyber".



6. Write and test a SQL statement that will Delete the ClassEnrollments record with the lowest EnID

Oracle SQL Developer : C:\Users\pwalsh\Desktop\School\UMGC\SDEV 350\Week 2\Lab 2.sql

The screenshot shows the Oracle SQL Developer interface. The left pane displays the 'Connections' tree with 'SDEV350 Connection' selected. The main workspace shows a SQL worksheet with the following code:

```
160 /*
161 6. Write and test a SQL statement that will Delete the ClassEnrollments
162 record with the lowest EnID
163 */
164 DELETE FROM ClassEnrollments WHERE EnID = (SELECT MIN(EnID) FROM ClassEnrollments);
165 --SELECT * FROM ClassEnrollments;
166
167
168
169
```

The bottom pane shows the 'Script Output' tab with the message: 'Task completed in 2.585 seconds' and '1 row deleted.'

7. Write and test a SQL statement that creates a **view** joining the required tables such that a user can retrieve the Engineer's Lastname and Firstname, the Faculty Lastname and Email and the Classes's Subject and Title for each Course enrollment.

Oracle SQL Developer: C:\Users\pwalsh\Desktop\School\UMGC\SDEV 350\Week 2\Lab 2.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections

Oracle Connections

SDEV350 Connection

Tables (Filtered)

CLASSENROLLMENTS

CLASSES

ENGINEERS

FACULTY

Views

COOL_VIEW

Indexes

Packages

Procedures

Functions

Operators

Queues

Queues Tables

Triggers

Types

Sequences

Materialized Views

Reports

All Reports

Analytic View Reports

Data Dictionary Reports

Data Modeler Reports

OLAP Reports

TimesTen Reports

User Defined Reports

SQL Worksheet: History

Worksheet Query Builder

```
175 /*
176 CREATE VIEW cool_view AS SELECT
177   EnID, eng.Lastname AS Eng_Last, eng.Firstname AS Eng_First, fac.Lastname AS Fac_Lastname, fac.Email AS Fac_Email, Subject, Title
178 FROM ClassEnrollments, Engineers eng, Faculty fac, Classes
179 WHERE ClassEnrollments.EID=eng.EID
180 AND ClassEnrollments.FID=fac.FID
181 AND ClassEnrollments.CID=Classes.CID
182 ORDER BY EnID;
183 SELECT * FROM cool_view;
184
```

Script Output x Query Result x

Task completed in 2.585 seconds

1 row deleted.

View COOL_VIEW created.

ENID	ENG_LAST	ENG_FIRST	FAC_LASTNAME	FAC_EMAIL	SUBJECT	TITLE
2	Philbin	Amadeus	Levensen	jlevonsen@test.com	CMIS	Intermediate Programming
3	Kapoor	Kelly	Wallace	dwallace@test.com	CMSC	Game Design
4	Scott	Michael	Levensen	jlevonsen@test.com	IOT Cyber	Database Security
5	Shrute	Dwight	Wallace	dwallace@test.com	CMIS	Intermediate Programming
6	Malone	Kevin	Friendship	rcalifornia@test.com	IOT Cyber	Database Security
7	Bernard	Andy	Wallace	dwallace@test.com	IOT Cyber	Database Security
8	Howard	Ryan	Levensen	jlevonsen@test.com	CMSC	Game Design
9	Flenderson	Toby	Wallace	dwallace@test.com	CMSC	Game Design
10	Packer	Todd	Levensen	jlevonsen@test.com	CMSC	Game Design
11	Halpert	Jim	Levensen	jlevonsen@test.com	IOT Cyber	Database Security
12	Martinez	Oscar	Levensen	jlevonsen@test.com	CMIS	Intermediate Programming
13	Hudson	Stanley	Wallace	dwallace@test.com	CMIS	Intermediate Programming
14	Philbin	Amadeus	Friendship	rcalifornia@test.com	CMSC	Game Design
15	Lapin	Phyllis	Wallace	dwallace@test.com	IOT Cyber	Database Security

14 rows selected.