Practice 4

Practice name	DDL2
Academic Program	Software Engineering
Subject name	Laboratory of Database Systems II
Unit	I. SQL.
Professor	Aldonso Becerra Sánchez
Due date	September 8, 2022
Due date with penalty	September 9, 2022
Elaboration date	September 6, 2022

Practice objective	Review the notion of Oracle DDL statements creating other types of object.
Estimated time of completion	5 hours
Introduction	The Oracle DDL language is transcendental in the handling of SQL statements at the level of both administrator and database programmer, since it allows the definition of database schemes regardless of the platform used to generate it. Sequences, synonymous and indexes are salient objects in Oracle, since they can help you in several tasks during daily programmer's days.

Reference 1:

1. Oracle Database 11g: SQL Fundamentals.

Reference 2:

2. Oracle Database SQL Language Reference 11g.

Reference 3:

Initial Activity:

Write the corresponding report. Start with the **Introduction** section.

Activity 1:

Write the section that describes the **work developed** in the following activities.

Read all the choices carefully because there might be more than one correct answer. Choose all the correct answers for each question.

Explain the reason for your answer.

CREATE PRIVATE AND PUBLIC SYNONYMS

1. What are distinguishing characteristics of a public synonym rather than a private synonym? Challenge question.

(Choose two correct answers.)

A. Public synonyms are always visible to all users.

B. Public synonyms can be accessed by name without a schema name qualifier.

C. Public synonyms can be selected from without needing any permission.

D. Public synonyms can have the same names as tables or views.

2. Consider these three statements:

create synonym s1 for staff;

create public synonym s1 for warehouse;

select * from s1;

Which of the following statements is correct? (Choose the best answer.)

A. The second statement will fail because an object S1 already exists.

B. The third statement will show the contents of warehouse.

C. The third statement will show the contents of staff.

D. The third statement will show the contents of the table S1, if such a table exists in the current schema.

3. A view and a synonym are created as follows:

create view dept_v as select * from dept;

create synonym dept_s for dept_v;

Subsequently the table DEPT is dropped. What will happen if you query the synonym DEPT_S? (Choose the best answer.)

A. There will not be an error because the synonym addresses the view, which still exists, but there will be no rows returned.

B. There will not be an error if you first recompile the view with the command ALTER VIEW DEPT_V COMPILE FORCE;

C. There will be an error because the synonym will be invalid.

D. There will be an error because the view will be invalid.

E. There will be an error because the view will have been dropped implicitly when the table was dropped.

CREATE, MAINTAIN, AND USE SEQUENCES

4. A sequence is created as follows:

create sequence seq1 maxvalue 100;

If the current value is already 100, when you attempt to select SEQ1.NEXTVAL what will happen? (Choose the best answer.)

A. The sequence will cycle and issue 0.

B. The sequence will cycle and issue 1.

C. The sequence will reissue 100.

D. There will be an error.

5. You create a sequence as follows:

create sequence seq1 start with 5;

After selecting from it a few times, you want to reinitialize it to reissue the numbers already generated. How can you do this? (Choose the best answer.)

A. You must drop and re-create the sequence.

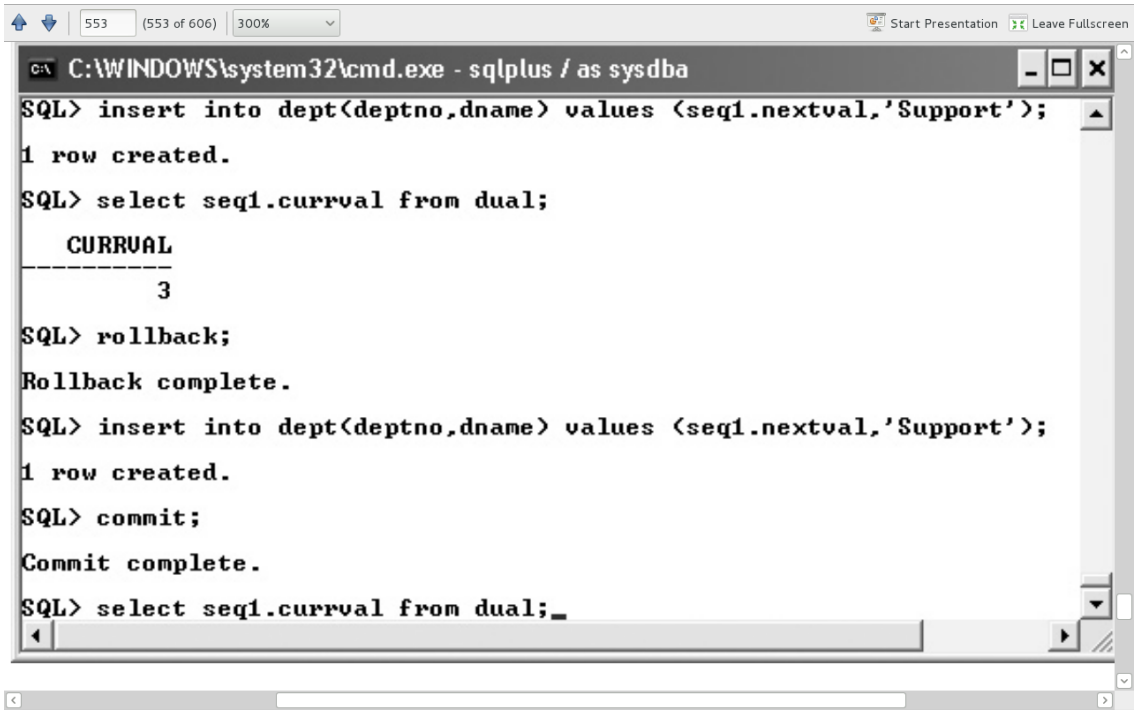
B. You can't. Under no circumstances can numbers from a sequence be reissued once they have been used.

C. Use the command ALTER SEQUENCE SEQ1 START WITH 5; to reset the next value to 5.

D. Use the command ALTER SEQUENCE SEQ1 CYCLE; to reset the sequence to its

starting value.

6. Study the following exhibit:



Assuming that the sequence SEQ1 was created with the option ORDER and INCREMENT BY set to 1, what value will be returned by the final SELECT statement? (Choose the best answer.)

A. 2

B. 3

C. 4

D. It will depend on whether any other sessions are selecting from the sequence while the statements in the exhibit are being run.

CREATE AND MAINTAIN INDEXES

7. A UNIQUE constraint on a column requires an index. Which of the following scenarios is correct? (Choose one or more correct answers.)

A. If a UNIQUE index already exists on the column, it will be used.

B. If a NONUNIQUE index already exists it will be used.

C. If a NONUNIQUE index already exists on the column, a UNIQUE index will be created implicitly.

D. If any index exists on the column, there will be an error as Oracle attempts to create another index implicitly.

8. This statement will fail:

create unique bitmap index on employees(department_id,hire_date);

Why? (Choose the best answer.)

A. Bitmap indexes cannot be unique.

B. The two columns are of different data types.

C. A bitmap index can be on only one column.

D. There is already a B*Tree index on DEPARTMENT_ID.

9. You have created an index with this statement:

create index ename_i on employees(last_name,first_name);

How can you adjust the index to include the employees' birthdays, which is a date type column called DOB? (Choose the best answer.)

A. Use ALTER INDEX ENAME_I ADD COLUMN DOB;

B. You can't do this because of the data type mismatch.

C. You must drop the index and re-create it.

D. This can only be done if the column DOB is NULL in all existing rows.

Activity 2:

Consider the following context issue:

SHOP (shop_id, address, manager).

PRODUCT (product_id, pname, sale_price, purchase_price, provider).

CHANNEL (channel_id, cname).

EMPLOYEE (emp_id, emp_name, emp_lastn, boss_id, address, date_of_birth, gender, beneficiaries)

Figure 1 shows an entity-relationship diagram for a simple system designed to store and analyze sales. The columns for the fact table SALES are as follows:

- SALE_ID System-generated primary key
- CHANNEL_ID Foreign key to CHANNELS.
- PRODUCT_ID Foreign key to PRODUCTS.
- SHOP_ID Foreign key to SHOPS.
- QUANTITY The quantity of the product sold.
- EMP_ID The id of the employee that sold a product. Foreign Key to EMPLOYEE.
- SALE_DATE The date of the sale.

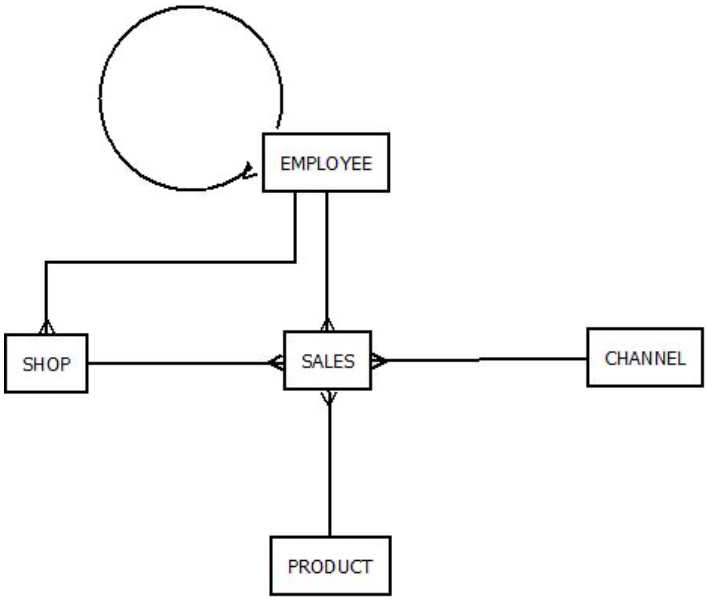


Figure 1. Sales relational diagram

It is expected that there will be several million SALES rows per year. The dimension tables are as follows:

PRODUCT A list of all products, including price (a few hundred).

EMPLOYEE A list of employees in the stores. Hundreds of employees.

CHANNEL Possible sales methods, such as walk-in, Internet, and telephone.

SHOP Details of all the shops (no more than a couple of dozen).

SALES Details of the sales in the different shops. Thousands of sales per day.

- Write code to create the tables (only columns data type specifications);
- Create indexes (choose appropriate type);
- Create constraints (all that it needs).
- Create sequences to be used for primary keys where necessary with the best options.
- Create short name synonymous for each table.

Activity 3:

Consider these tables:

```
create table vehicle(id_veh number, lic_plates varchar2(10),
owner_id number);
create table owner(owner_id number, surname varchar2(10),
forename varchar2(10), dateobirth date);
```

Reports often require information from surname and forename of owners, reports from vehicle license plates, vehicle license plates of specific people.

1. Define required indexes (choose adequate type).
2. Add required constraints.

Activity 4:

Propose a response to the following issues:

- You are involved in designing a database to be used for online order entry and offline financial reporting. What should you consider with regard to data consulting, synonyms, and indexes?
- Should sequences always be used for primary keys?

Activity 5:

In this exercise, you will create some sequences and use them. You will need two concurrent sessions.

1. Log on to your database twice, as YOURSCHEME in separate sessions. Consider one to be your A session and the other to be your B session.
2. In your A session, create a sequence as follows:
   ```
   create sequence seq1 start with 10 nocache maxvalue
   15 cycle;
   ```
   The use of NOCACHE is deleterious for performance. If MAXVALUE is specified, then CYCLE will be necessary to prevent errors when MAXVALUE is reached.
3. Execute the following commands in the appropriate session in the correct order to observe the use of NEXTVAL and CURRVAL and the cycling of the sequence:

	In Your A Session	In Your B Session
1st	select seq1.nextval from dual;	
2nd		select seq1.nextval from dual;
3rd	select seq1.nextval from dual;	
4th		select seq1.nextval from dual;
5th	select seq1.currval from dual;	
6th		select seq1.nextval from dual;
7th	select seq1.nextval from dual;	
8th		select seq1.currval from dual;
9th	select seq1.nextval from dual;	
10th		select seq1.nextval from dual;

4. Create a table with a primary key:
   ```
   create table seqtest(c1 number,c2 varchar2(10));
   alter table seqtest add constraint seqtest_pk
   primary key (c1);
   ```
5. Create a sequence to generate primary key values:
   ```
   create sequence seqtest_pk_s;
   ```
6. In your A session, insert a row into the new table and commit:
   ```
   insert into seqtest
   values(seqtest_pk_s.nextval,'first');
   commit;
   ```
7. In your B session, insert a row into the new table and do not commit it:
   ```
   insert into seqtest
   values(seqtest_pk_s.nextval,'second');
   ```
8. In your A session, insert a third row and commit:

```
insert into seqtest
values(seqtest_pk_s.nextval,'third');
commit;
```

9. In your B session, roll back the second insertion:
```
rollback;
```
10. In your B session, see the contents of the table:
```
select * from seqtest;
```
This demonstrates that sequences are incremented and the next value published immediately, outside the transaction control mechanism.
11. Tidy up:
```
drop table seqtest;
drop sequence seqtest_pk_s;
drop sequence seq1;
```

NOTE: Capture an image for each statement output.

Activity 6:

Execute the following sentences and include an image for each one:

Creating Indexes

In this exercise, create indexes on a copy of the EMPLOYEES table in the HR schema.

1. Connect to your database as your user.

2. Create a table that is a copy of HR.EMPLOYEES:

create table emps as select * from hr.employees;

This table will have neither indexes nor primary, unique, or foreign key constraints, because these are not copied by a CREATE TABLE AS command.

The NOT NULL constraints will have been copied. Confirm this by describing the table:

describe emps;

3. Create an index to be used for the primary key constraint:

create unique index emps_empid_i on emps(employee_id);

4. Demonstrate that a unique index cannot accept duplicates, even before a constraint is defined:

insert into emps(employee_id,last_name,email,hire_date,job_id)

values(198,'Watson','jw@bplc.co.za',sysdate,'IT_PROG');

This will return an error because the index cannot insert a second employee_id 198. Index uniqueness is an attribute of the index that can exist without a constraint but should not be relied upon to enforce data integrity.

5. Create additional indexes on columns that are likely to be used in WHERE clauses, using B*Tree for columns of high cardinality and bitmap for columns of low cardinality:

create index emps_name_i on emps(last_name,first_name);

create index emps_tel_i on emps(phone_number);

create bitmap index emps_mgr_i on emps(manage_id);

create bitmap index emps_dept_i on emps(department_id);

6. Define some constraints:

alter table emps add constraint emps_pk primary key (employee_id);

alter table emps add constraint emps_email_uk unique(email);

alter table emps add constraint emps_tel_uk unique(phone_number);

7. Display the index names and their type:

select index_name,index_type,uniqueness from user_indexes

where table_name='EMPS';

The view USER_INDEXES shows details of all indexes in your current schema. Note that in addition to the five indexes explicitly created in steps 3 and 5, there is also an index created implicitly with the name of the constraint defined on EMAIL. Note also that the unique constraint on PHONE_NUMBER is being enforced with a nonunique index; this is perfectly possible, because although the constraint mechanism uses indexes, it is independent of the structure of the index.

8. Tidy up by dropping the EMPS table, and confirm that all the indexes have also gone:

drop table emps;

select index_name from user_indexes where table_name='EMPS';

Activity 7:

Write the section that describes the **work developed** in the following activities.

Consider the following context issues employed in past practices. For each one:

- Analyze the use of "sequences", "indexes", and "synonyms" objects for a new formulation of the solution according to the user requirements appreciated in each case.
- Write code to create the tables (only the create table statements with basic constrains: default, check, not null, …, and column data type specifications).
- Create indexes (choose appropriate type).
- Create constraints (unique, primary key, foreign keys, …).
- Create sequences to be used for primary keys where necessary with the best options.
- Create short name synonymous for each table.

User case 1. Bookstore. The establishment of activity 1/practice 1.

There is an establishment that provides book sales. Such books are always interested in knowing their author (s) and their publisher, as well as the warehouse where they are stored and the sales establishment (around the world) where they are sold. Suppose the bookstore adds music cassettes, CDs, Blue-ray and electronic cards for online purchases (a code for digital format) to its collection.

1) Generate an entity-relationship model that allows obtaining information such as the one indicated (this information is only a set of requirements):

- Customers who buy the most items.
- Products that customers currently have in the shopping cart.
- Know the best-selling product.
- Know what items are in stock on a specific date.
- Obtain the lists of the books with their complete data.
- Know the author of the best-selling book.
- Know the best-selling musical medium.
- Know the least sold type of musical medium.
- Know today's sales.
- Kwon the best-selling electronic files.

User Case 2. Accounts and loans. The Bank scenario of activity 8/practice 2.

- Check the primary key columns.
- Then correct the necessary attributes.
- User requirements:
    - List of clients order by name and account.
    - List of loans and balances.

- List of employees and their boss.
- List of loans and their payments.
- List of branches and their corresponding loans and accounts.

User case 3. Sales. The patients scenario of activity 3/practice 1.

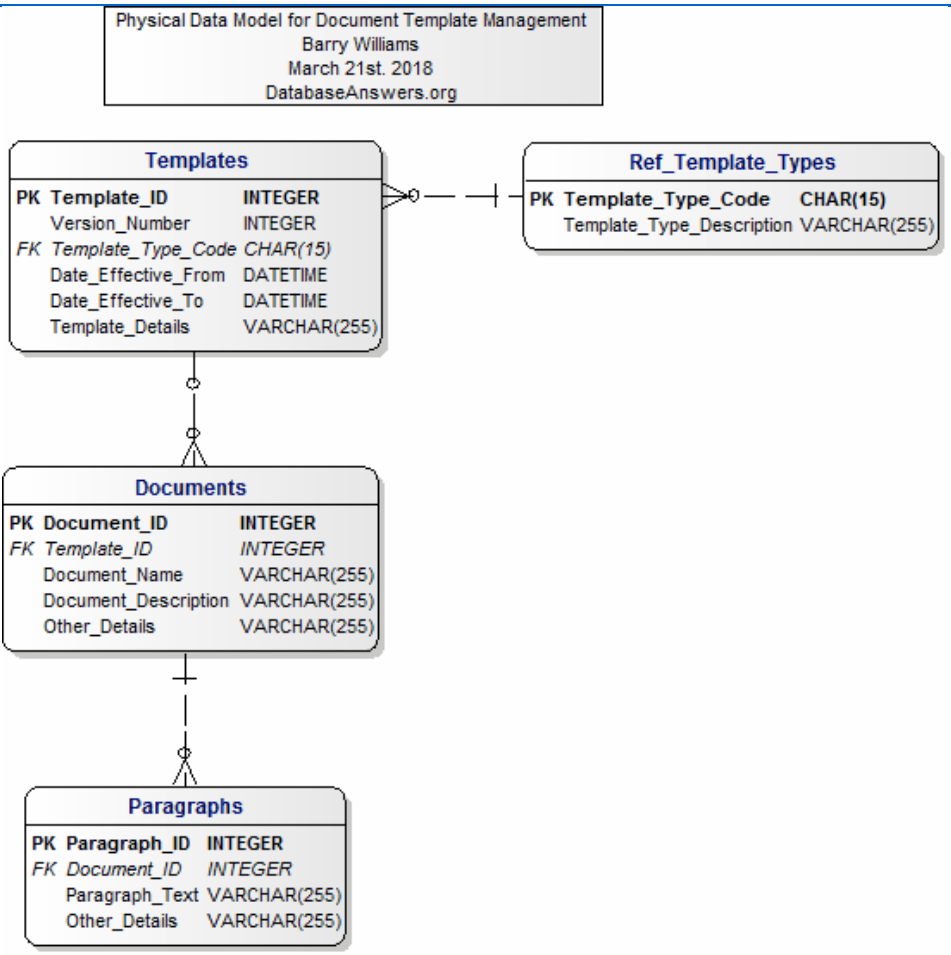
User requirements:

- List of patients.
- List of staffs.
- The most used id room.
- The most common medical condition.
- The list of payment types.
- The list of patient records.
- The list of patients addresses.
- The list of the patients records of the month.
- The place with the largest number of patients.

User case 4. Documents.

User requirements:

- How many paragraphs has a document?
- What is the most used template?
- What type of template is the least used?
- List the name of the templates and documents.



User case 5. Jobs. The jobs scenario of activity 7/practice 2.

User requirements:

- List of employees.
- List of jobs.
- List of branches.
- List of managers.
- Which jobs are being done?
- The jobs dates are being managed.

Activity 8:

Write the **Pre-assessment** section.

Final activity:

Write the **Conclusion** section.

Attached file that is required for this task (optional):

e-mail: a7donso@gmail.com