[www.oracle.com/academy](http://www.oracle.com/academy)

Database Programming with SQL 16-1: Working With Sequences Practice Solutions

# Vocabulary

Directions: Identify the vocabulary word for each definition below.

|  |  |
| --- | --- |
| **CREATE SE-**  **QUENCE** | Command that automatically generates sequential num-  Ber |
| **Sequences** | Generates a numeric value |
| **NEXTVAL** | Returns the next available sequence value |
| **INCREMENT BY** | Specifies the interval between sequence numbers |
| **NO MAXVALUE** | Specifies a maximum value of 10^27 for an ascending  sequence and -1 for a descending sequence (default) |
| **CURRVAL** | Returns the current sequence value |
| **MINVALUE** | Specifies the minimum sequence value |
| **CYCLE/ NOCYCLE** | Specifies whether the sequence continues to generate  values after reaching its maximum or minimum values |
| **NO MINVALUE** | Specifies a minimum value of 1 for an ascending se- quence and – (10^26) for a descending sequence (de- fault) |
| **MAXVALUE** | Specifies a maximum or default value the sequence can  Generate |
| **STARTS WITH** | Specifies the first sequence number to be generated |
| **CACHE/ NOCACHE** | Specifies how many values the Server pre-allocates and keeps in memory |

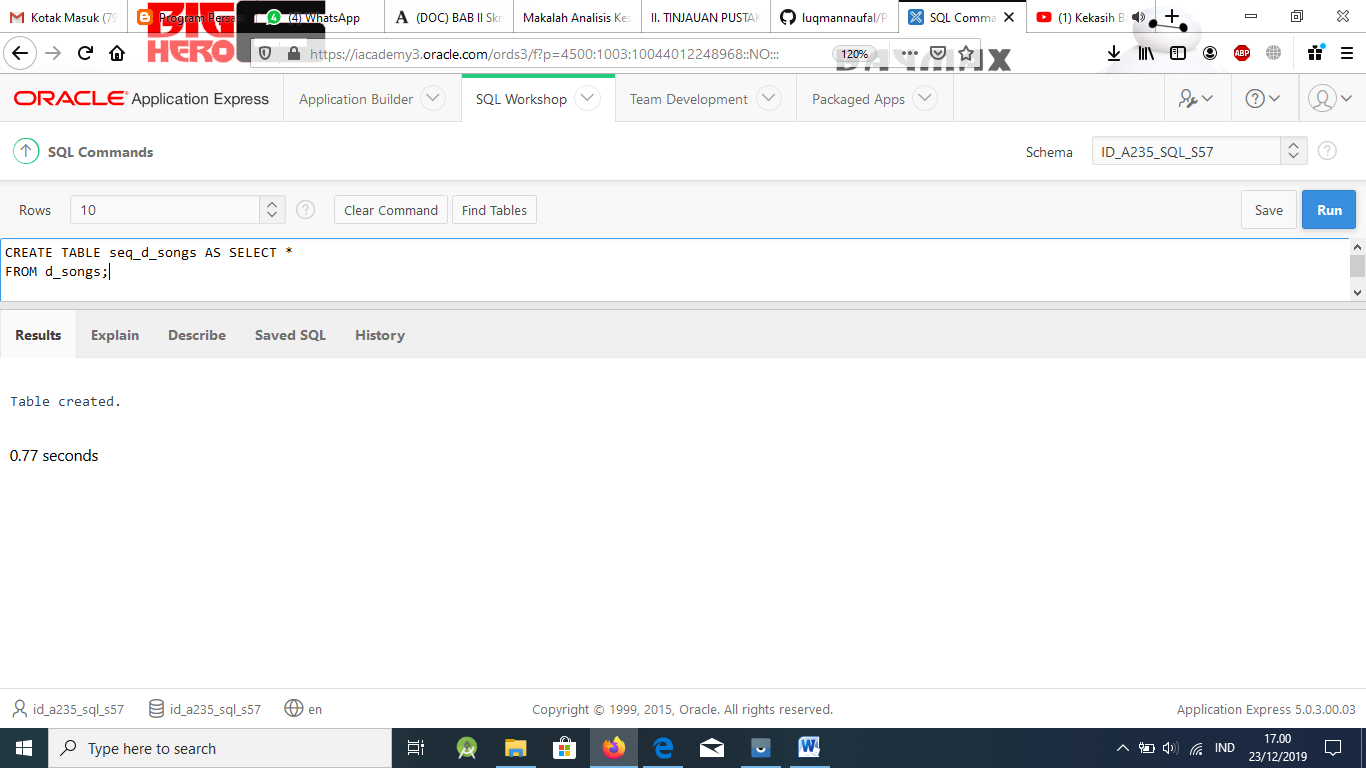
# Try It / Solve It

1. Using CREATE TABLE AS subquery syntax, create a seq\_d\_songs table of all the col- umns in the DJs on Demand database table d\_songs. Use the SELECT \* in the subquery to make sure that you have copied all of the columns.

## Solution:

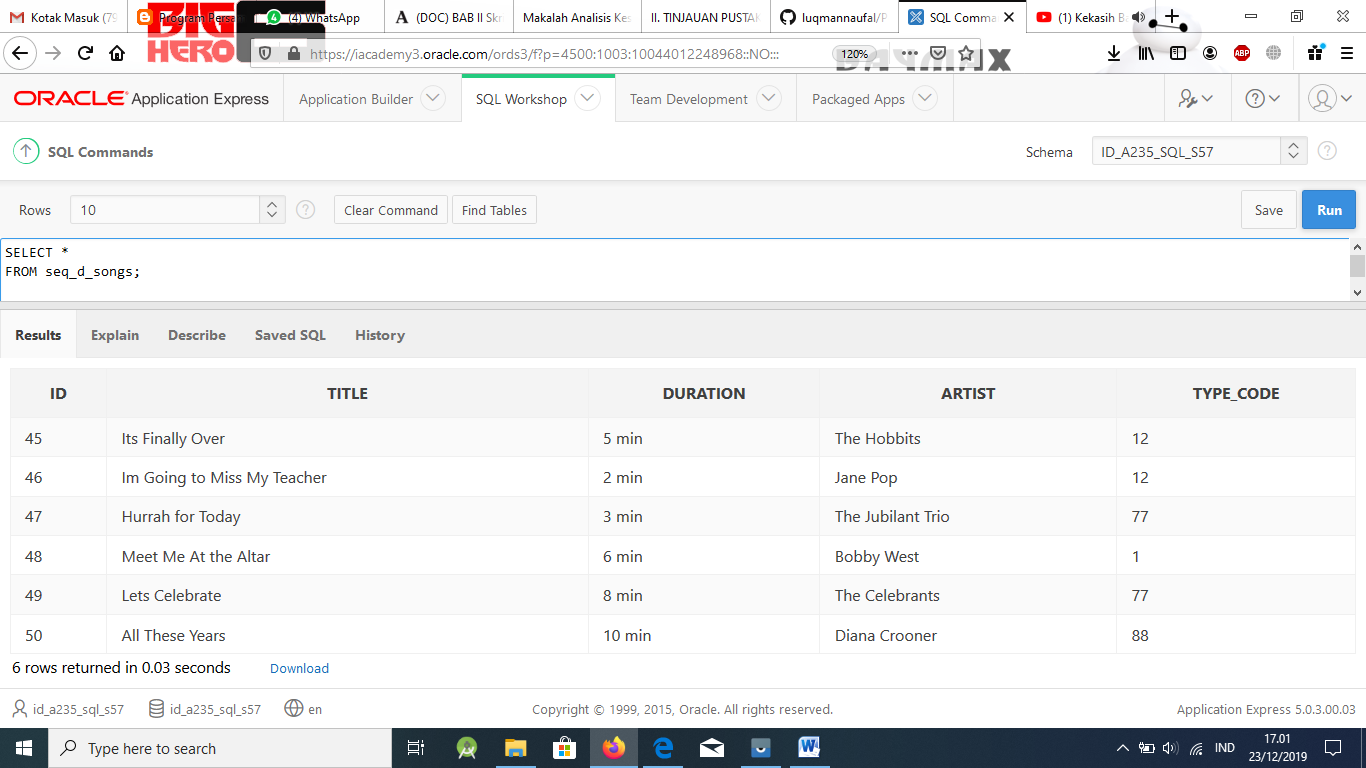
CREATE TABLE seq\_d\_songs AS SELECT \*

FROM d\_songs;



SELECT \*

FROM seq\_d\_songs;



1. Because you are using copies of the original tables, the only constraints that were carried over were the NOT NULL constraints. Create a sequence to be used with the primary-key column of the seq\_d\_songs table. To avoid assigning primary-key numbers to these tables

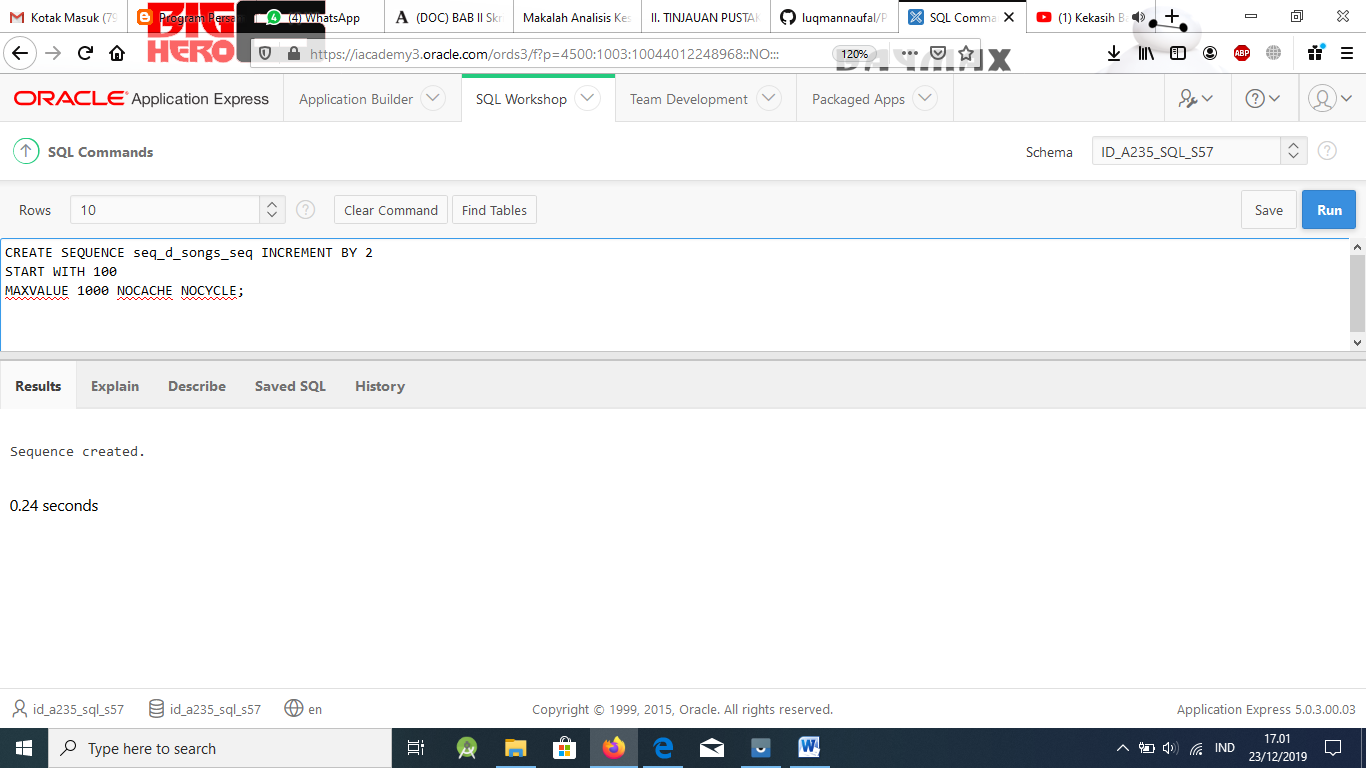
that already exist, the sequence should start at 100 and have a maximum value of 1000. Have your sequence increment by 2 and have NOCACHE and NOCYCLE. Name the se- quence seq\_d\_songs\_seq.

## Solution:

CREATE SEQUENCE seq\_d\_songs\_seq INCREMENT BY 2

START WITH 100

MAXVALUE 1000 NOCACHE NOCYCLE;

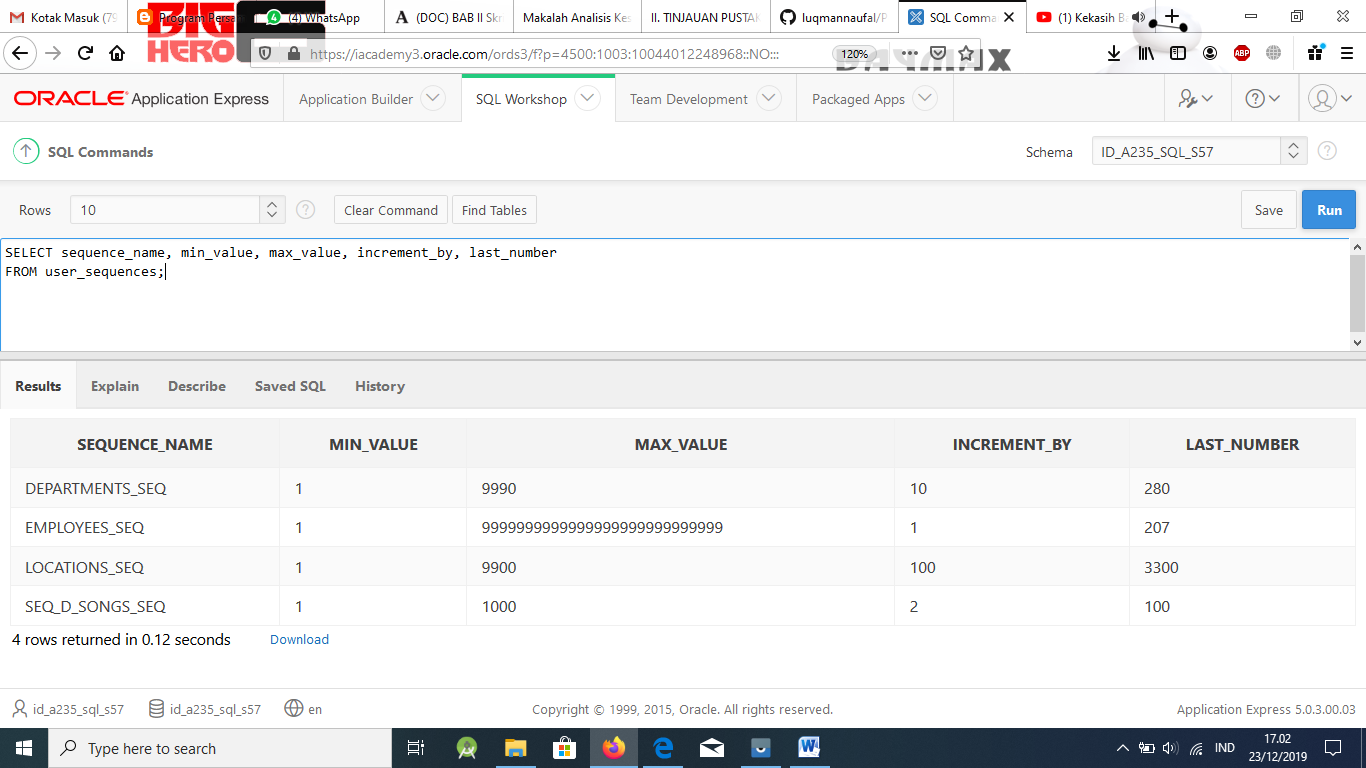


1. Query the USER\_SEQUENCES data dictionary to verify the seq\_d\_songs\_seq SE- QUENCE settings.

## Solution:

SELECT sequence\_name, min\_value, max\_value, increment\_by, last\_number

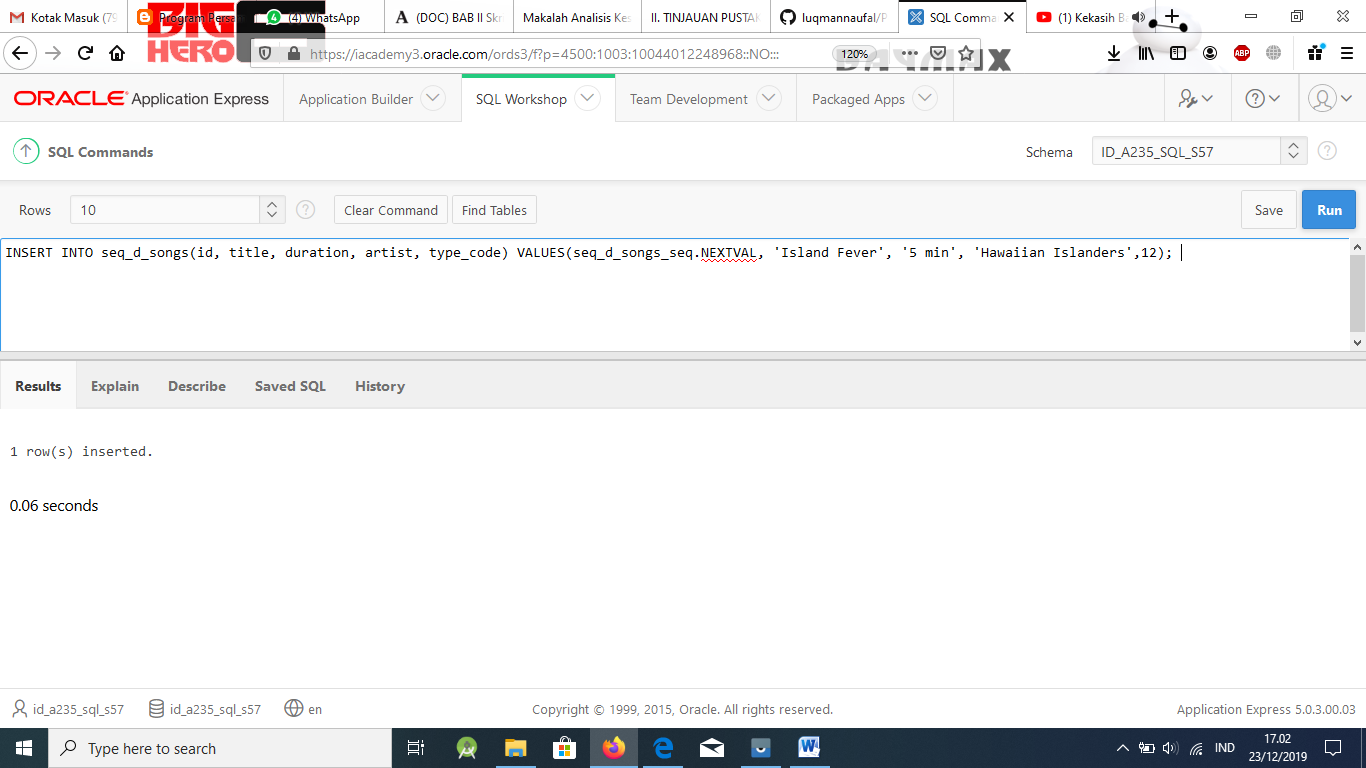
FROM user\_sequences;



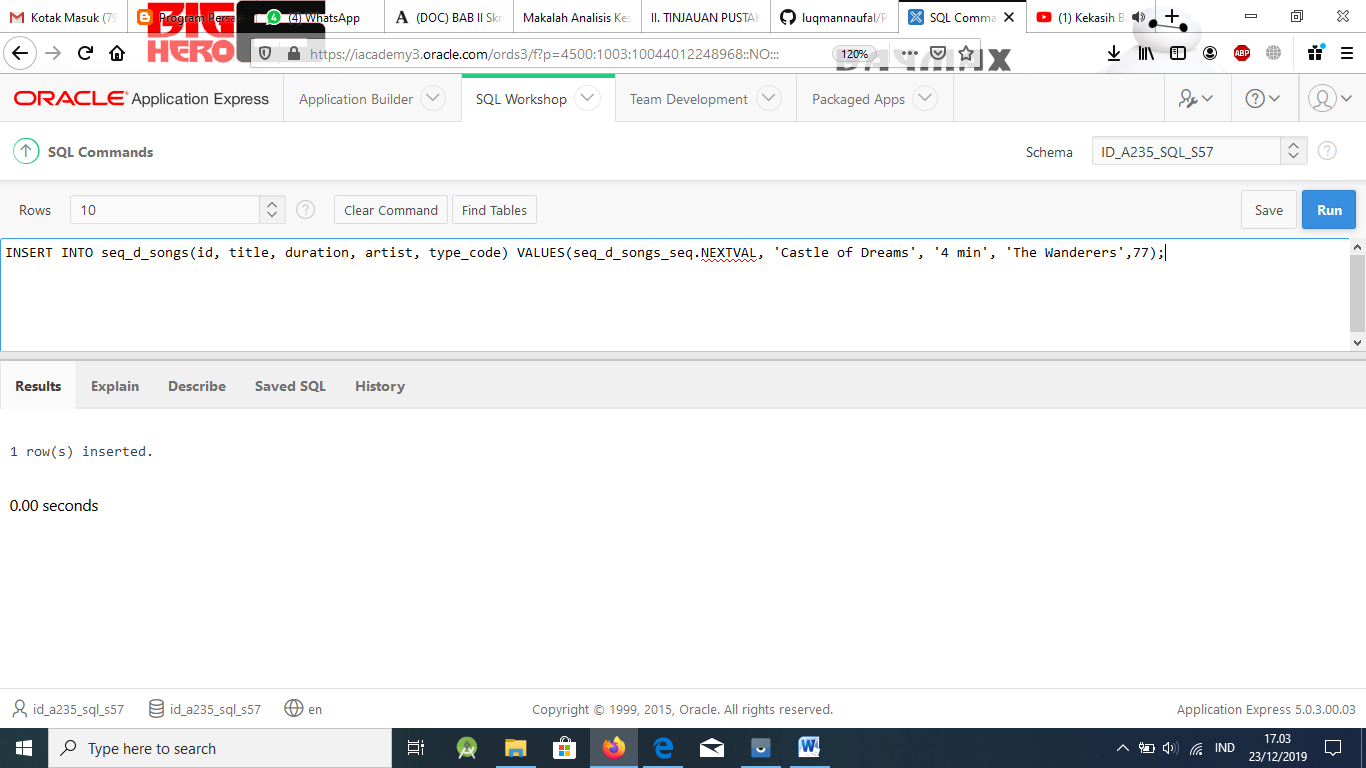
1. Insert two rows into the seq\_d\_songs table. Be sure to use the sequence that you created for the ID column. Add the two songs shown in the graphic.

## Solution:

INSERT INTO seq\_d\_songs(id, title, duration, artist, type\_code) VALUES(seq\_d\_songs\_seq.NEXTVAL, 'Island Fever', '5 min', 'Hawaiian Islanders',12);



INSERT INTO seq\_d\_songs(id, title, duration, artist, type\_code) VALUES(seq\_d\_songs\_seq.NEXTVAL, 'Castle of Dreams', '4 min', 'The Wanderers',77);



1. Write out the syntax for seq\_d\_songs\_seq to view the current value for the sequence. Use the DUAL table. (Oracle Application Express will not run this query.)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | TITLE | DURATION | ARTIST | TYPE\_CODE |
|  | Island Fever | 5 min | Hawaiian Islanders | 12 |
|  | Castle of Dreams | 4 min | The Wanderers | 77 |

## Solution:

SELECT seq\_d\_songs\_seq.CURRVAL FROM dual;

1. What are three benefits of using SEQUENCEs?

## Solution:

* + Reduces amount of application code so faster
  + Creates unique numbers
  + Shareable (this can cause problems, so is “useful” with care)

1. What are the advantages of caching sequence values?

## Solution:

Cache sequences in memory provide faster access to sequence values

1. Name three reasons why gaps may occur in a sequence?

## Solution:

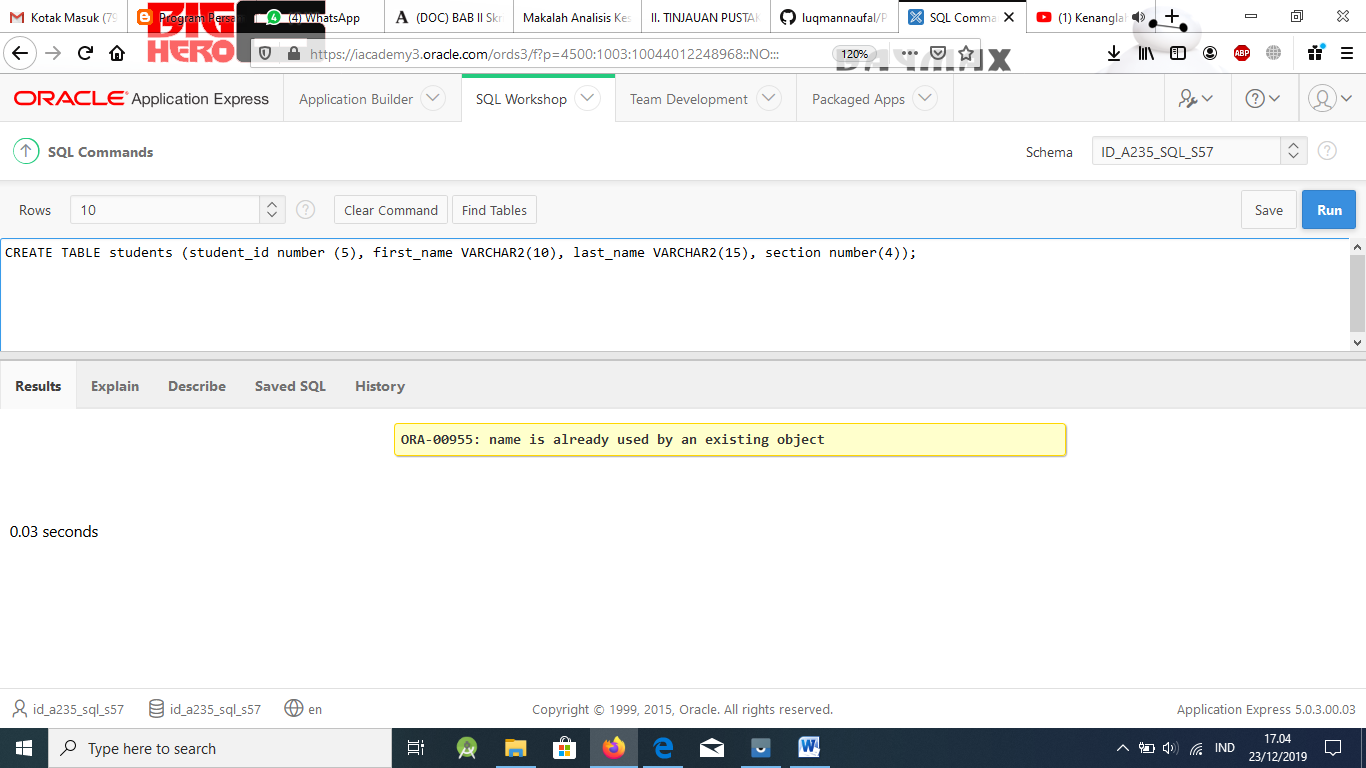
* + Rolling back a statement containing a sequence, the number is lost.
  + A system crash. If the sequence caches values into the memory and the system crashes, those values are lost.
  + The same sequence being used for multiple tables. If you do so, each table can con- tain gaps in the sequential numbers.

## NOTE: Use the following as an extension activity or as an additional assignment.

*Extension Exercise*

1. Create a table called “students.” You can decide which columns belong in that table and what datatypes these columns require. (The students may create a table with dif- ferent columns; however, the important piece that must be there is the student\_id col- umn with a numeric datatype. This column length must allow the sequence to fit, e.g. a column length of 4 with a sequence that starts with 1 and goes to 10000000 will not work after student #9999 is entered.)

CREATE TABLE students (student\_id number (5), first\_name VARCHAR2(10), last\_name VARCHAR2(15), section number(4));



1. Create a sequence called student\_id\_seq so that you can assign unique student\_id numbers for all students that you add to your table.

CREATE SEQUENCE student\_id\_seq INCREMENT BY 1

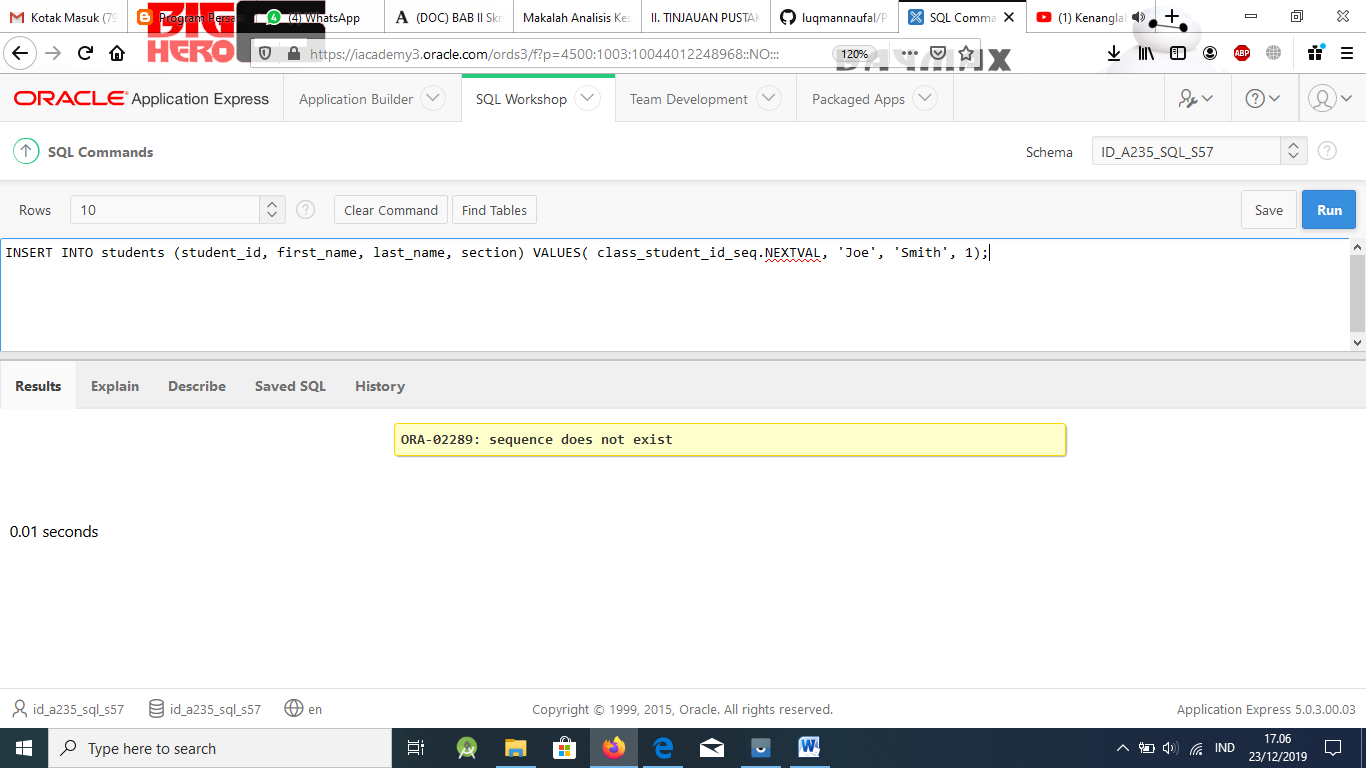
START WITH n

MAXVALUE nnnnn NOCACHE NOCYCLE ;

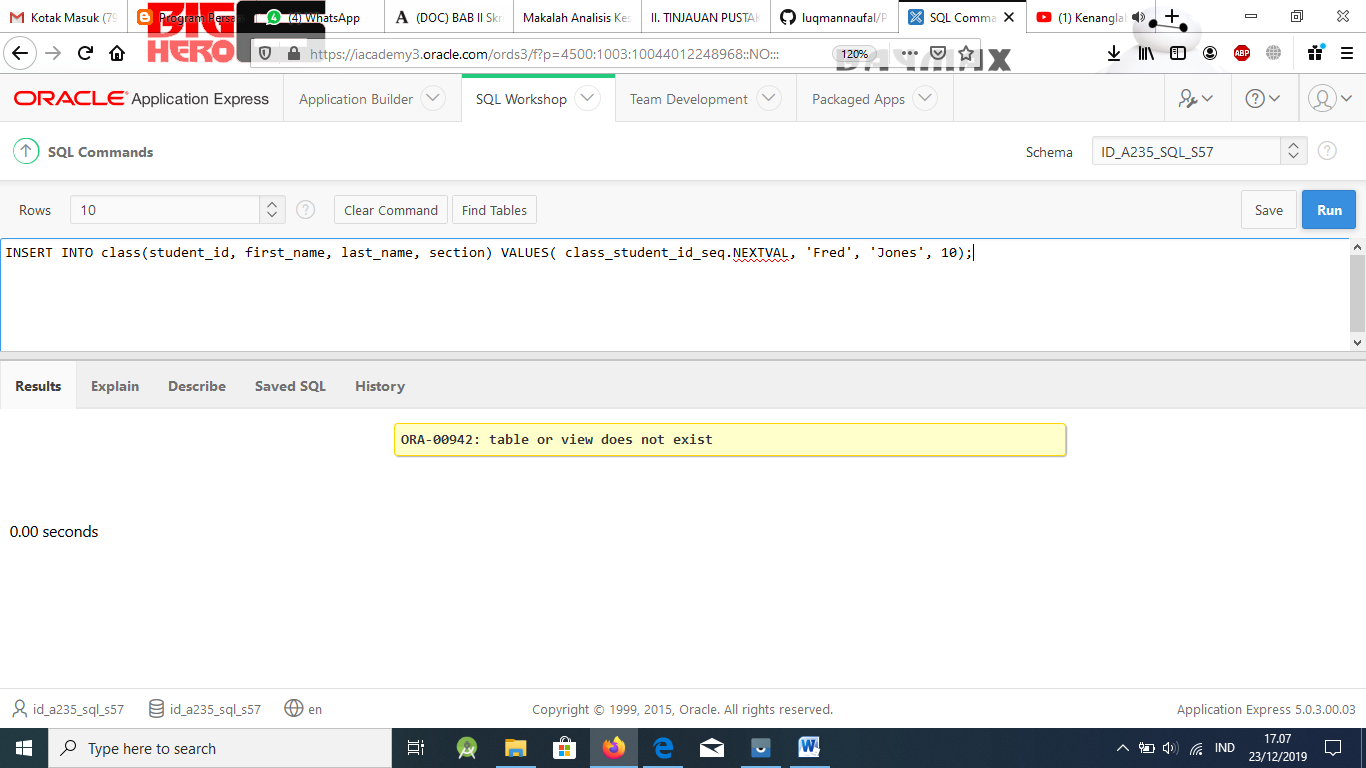
Ask students what it means if they specified NOCYCLE when they defined their se- quence. It means numbers will not be reused. Ask why that might be important. (Per- haps the school wants to keep track of how many different students have ever attend- ed the school. Perhaps students may leave the school, and then come back to the school, but still keep their original student id. This example is great for explaining NOCYCLE – can you use a similar example in the lecture content?)

1. Now write the code to add students to your STUDENTS table, using your sequence “database object.”

INSERT INTO students (student\_id, first\_name, last\_name, section) VALUES( class\_student\_id\_seq.NEXTVAL, 'Joe', 'Smith', 1);



INSERT INTO class(student\_id, first\_name, last\_name, section) VALUES( class\_student\_id\_seq.NEXTVAL, 'Fred', 'Jones', 10);



Again, their code could be different depending on what columns they used to create their tables. The important part for students to include in this code is the use of NEXTVAL. You may want to reiterate the CURRVAL issue with Oracle Application Ex- press here.

[www.oracle.com/academy](http://www.oracle.com/academy)

Database Programming with SQL 16-2: Indexes and Synonyms Practice Solutions

# Vocabulary

Directions: Identify the vocabulary word for each definition below.

|  |  |
| --- | --- |
| **Confirming index** | Confirms the existence of indexes from the  USER\_INDEXES data dictionary view |
| **Non-unique index** | Schema object that speeds up retrieval of rows |
| **CREATE PUBLIC SYNONYM** | To refer to a table by another name to simpli-  fy access |
| **Composite index** | An index that you create on multiple columns  in a table |
| **Unique index** | The Oracle Server automatically creates this index when you define a column in a table to have a PRIMARY KEY or a UNIQUE KEY  constraint |
| **Function-based index** | Stores the indexed values and uses the in- dex based on a SELECT statement to re- trieve the data |
| **DROP INDEX** | Removes an index |
| **Synonym** | Gives alternative names to objects |

# Try It / Solve It

1. What is an index and what is it used for?

## Solution:

Schema object used to speed up retrieval of rows by using a pointer.

1. What is a ROWID, and how is it used?

## Solution:

A ROWID is a base 64 string representation of the row address containing block identifier, row location in the block, and the database file identifier. The fastest way to access any particular row is by referencing its ROWID.

1. When will an index be created automatically?

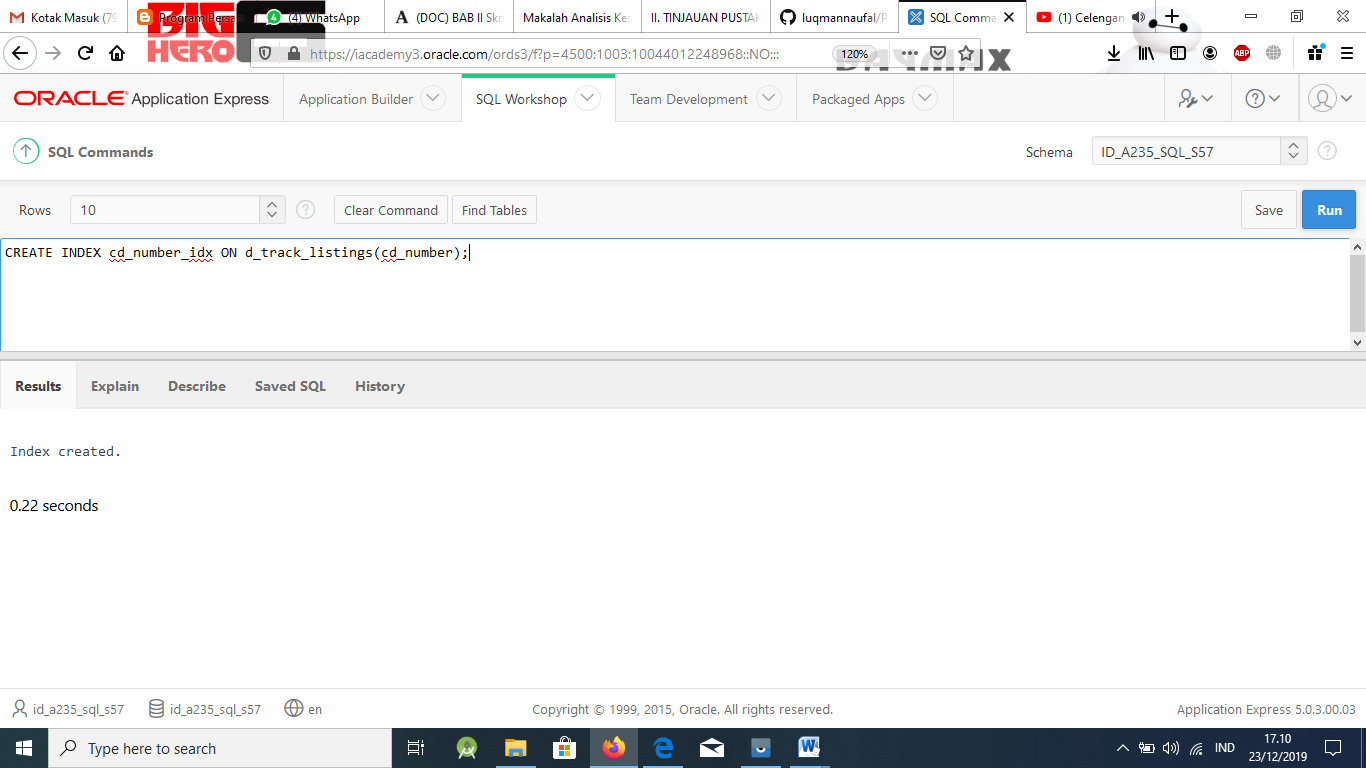
## Solution:

An index is automatically created for a PRIMARY or UNIQUE KEY constraint.

1. Create a nonunique index (foreign key) for the DJs on Demand column (cd\_number) in the D\_TRACK\_LISTINGS table. Use the Oracle Application Express SQL Workshop Data Browser to confirm that the index was created.

## Solution:

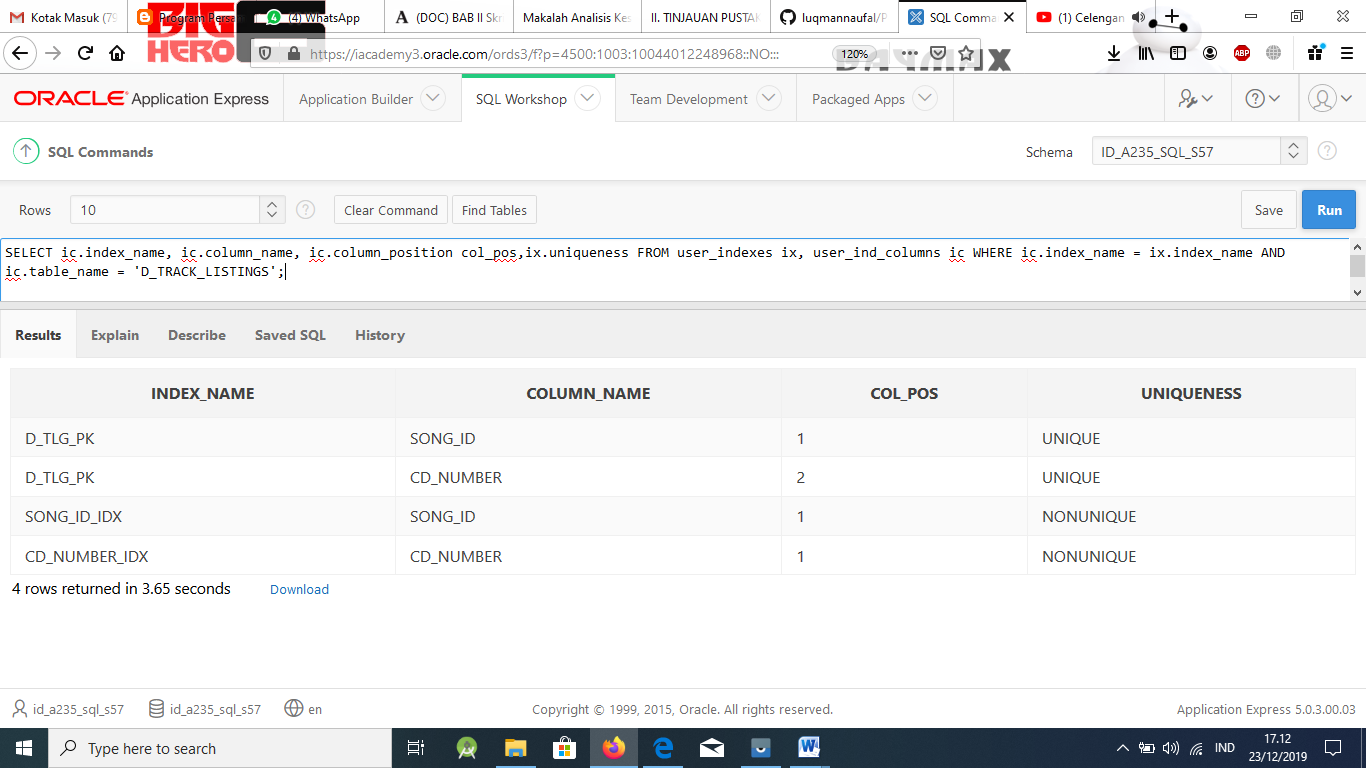
CREATE INDEX cd\_number\_idx ON d\_track\_listings(cd\_number);



There are two ways to confirm that the index was created.

* + The first way is to use the Data Browser to find a database object with type=INDEX and name=CD\_NUMBER\_IDX.
  + The second way is to query the data dictionary tables. Below is a sample query us- ing the tables, USER\_INDEXES and USER\_IND\_COLUMNS.

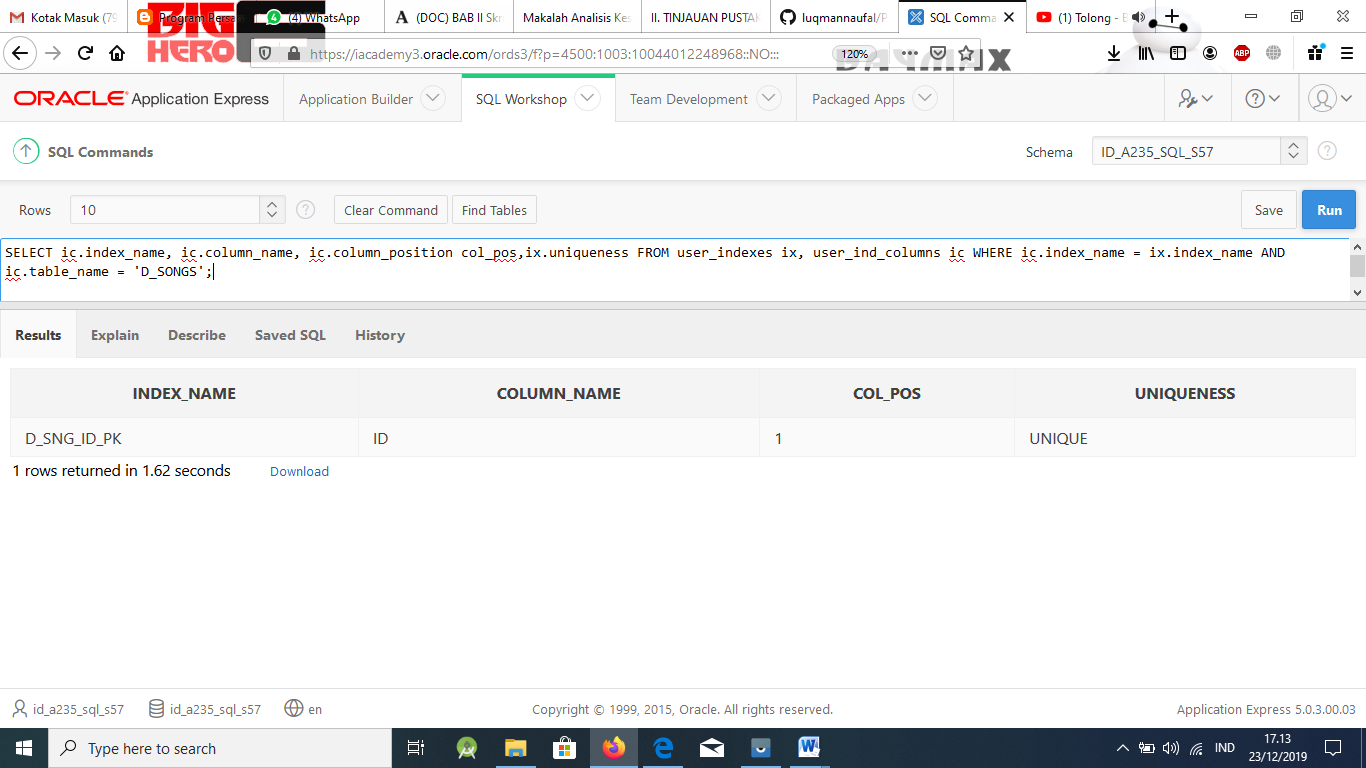
SELECT ic.index\_name, ic.column\_name, ic.column\_position col\_pos,ix.uniqueness FROM user\_indexes ix, user\_ind\_columns ic WHERE ic.index\_name = ix.index\_name AND ic.table\_name = 'D\_TRACK\_LISTINGS';



1. Use the join statement to display the indexes and uniqueness that exist in the data dic- tionary for the DJs on Demand D\_SONGS table.

## Solution:

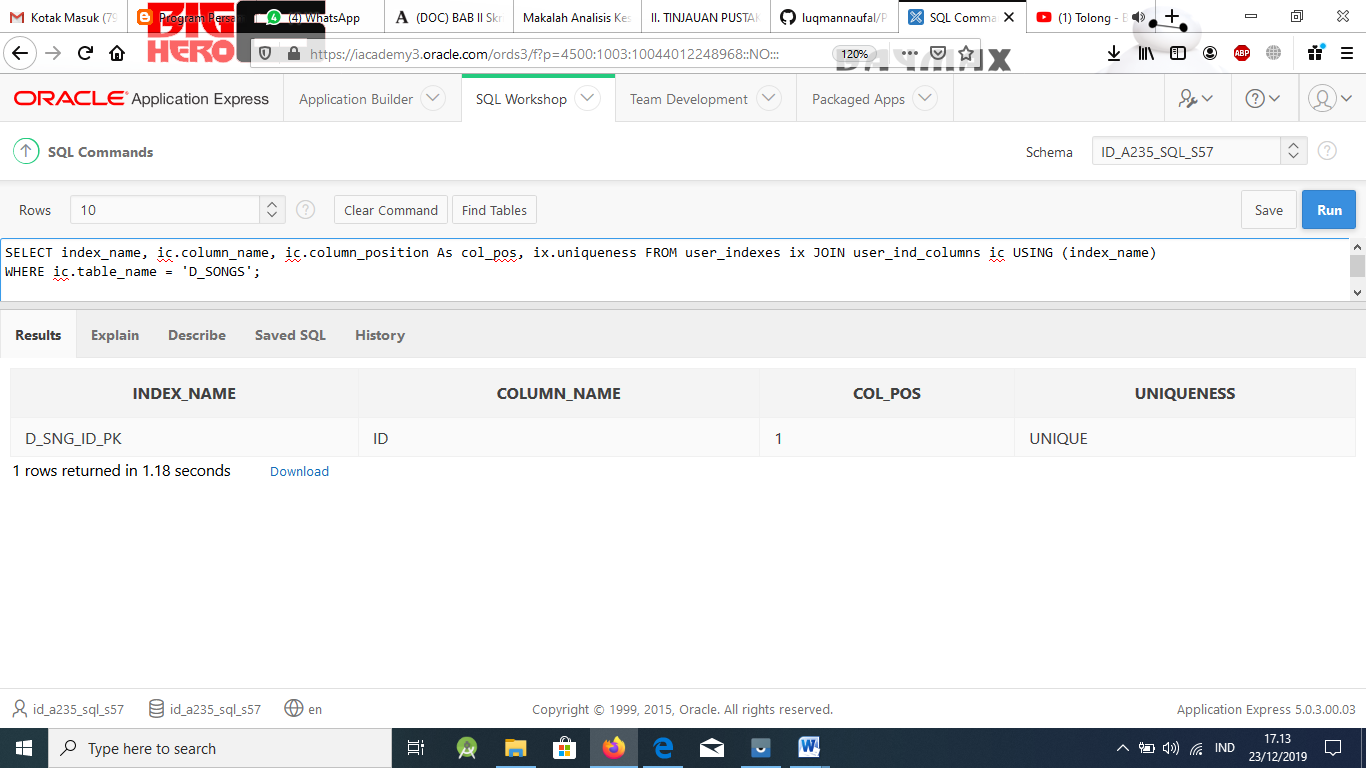
SELECT ic.index\_name, ic.column\_name, ic.column\_position col\_pos,ix.uniqueness FROM user\_indexes ix, user\_ind\_columns ic WHERE ic.index\_name = ix.index\_name AND ic.table\_name = 'D\_SONGS';



An alternative solution using the join statement is:

SELECT index\_name, ic.column\_name, ic.column\_position As col\_pos, ix.uniqueness FROM user\_indexes ix JOIN user\_ind\_columns ic USING (index\_name)

WHERE ic.table\_name = 'D\_SONGS';

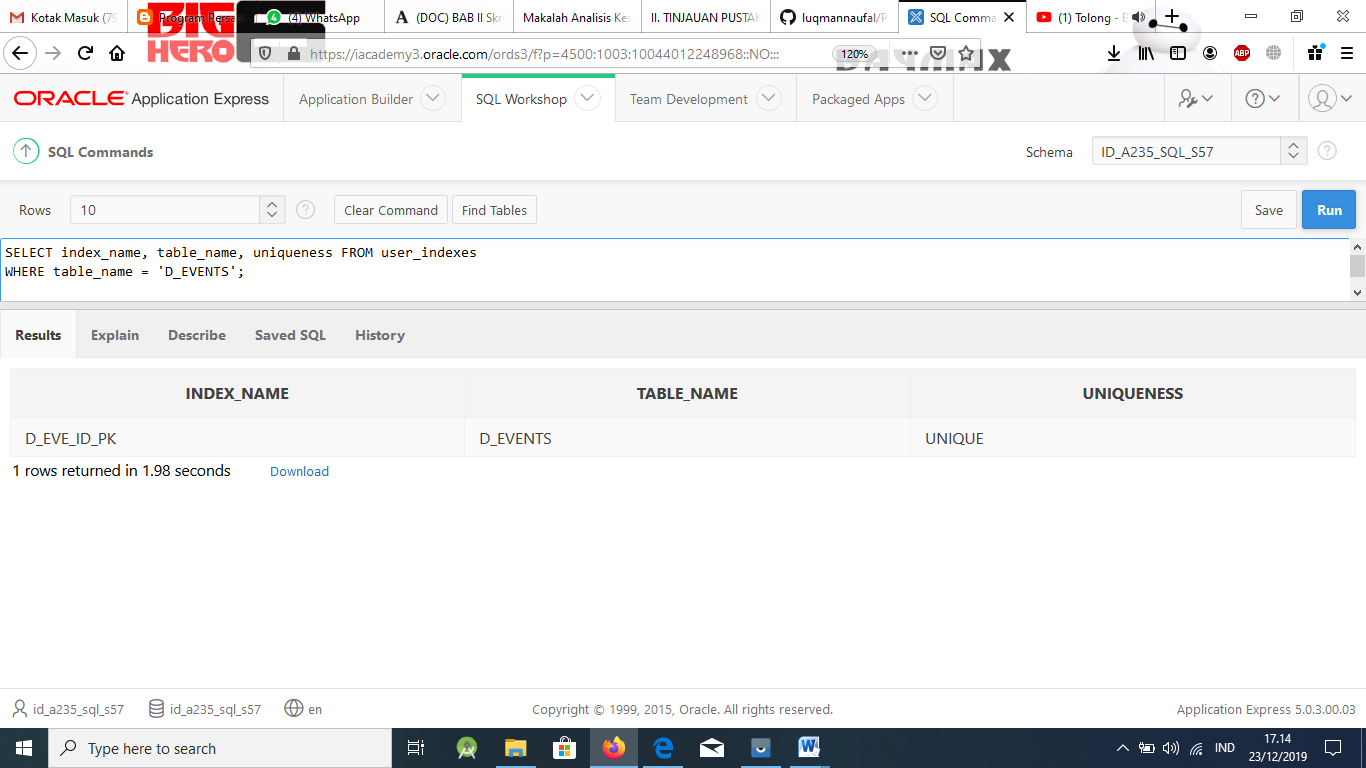


1. Use a SELECT statement to display the index\_name, table\_name, and uniqueness from the data dictionary USER\_INDEXES for the DJs on Demand D\_EVENTS table.

## Solution:

SELECT index\_name, table\_name, uniqueness FROM user\_indexes

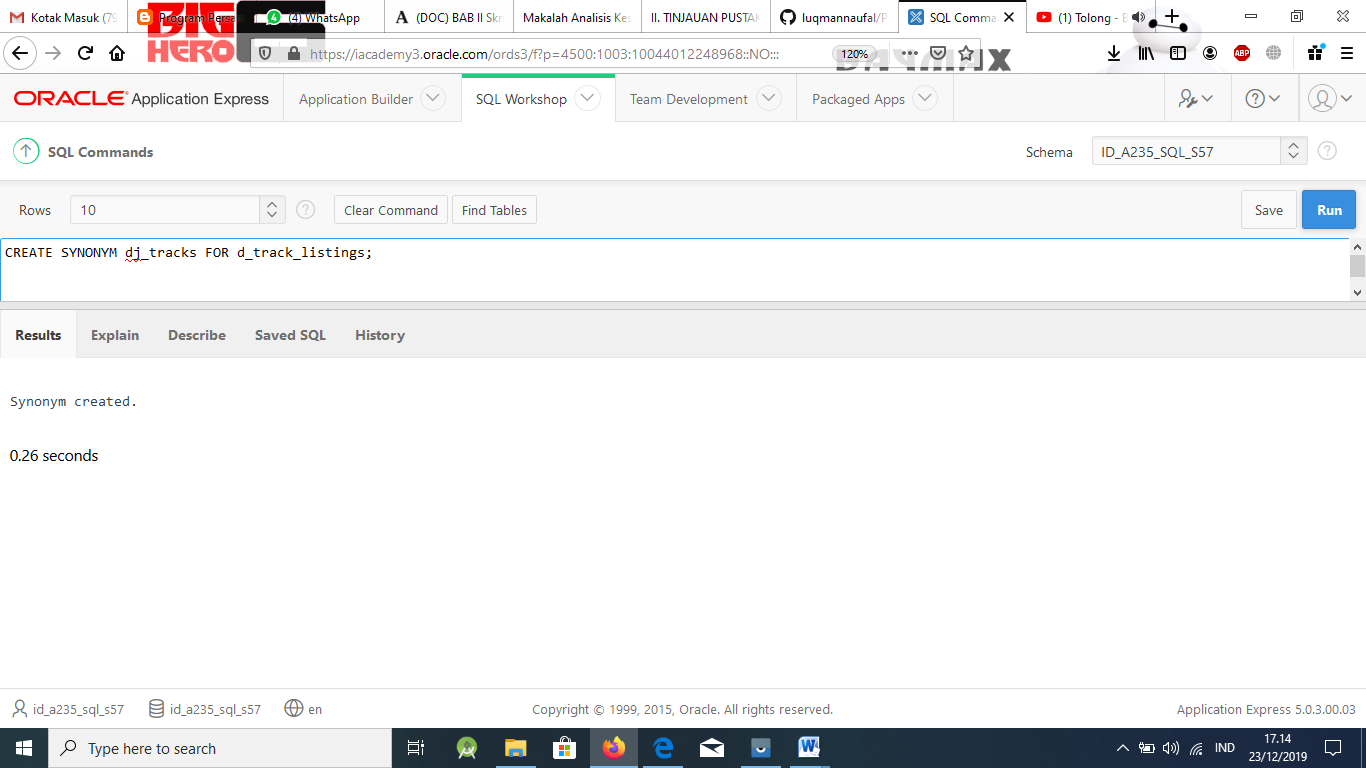
WHERE table\_name = 'D\_EVENTS';



1. Write a query to create a synonym called dj\_tracks for the DJs on Demand d\_track\_listings table.

## Solution:

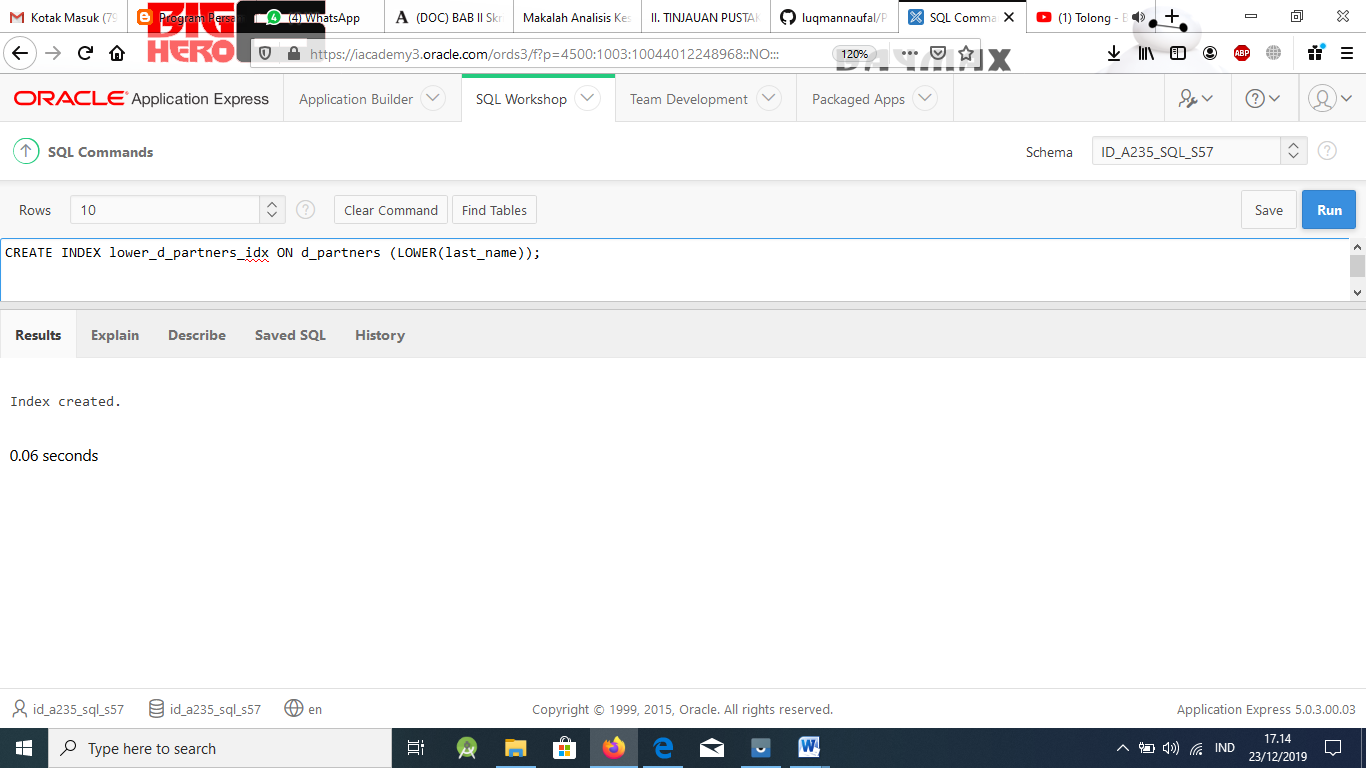
CREATE SYNONYM dj\_tracks FOR d\_track\_listings;



1. Create a function-based index for the last\_name column in DJs on Demand D\_PARTNERS table that makes it possible not to have to capitalize the table name for searches. Write a SELECT statement that would use this index.

## Solution:

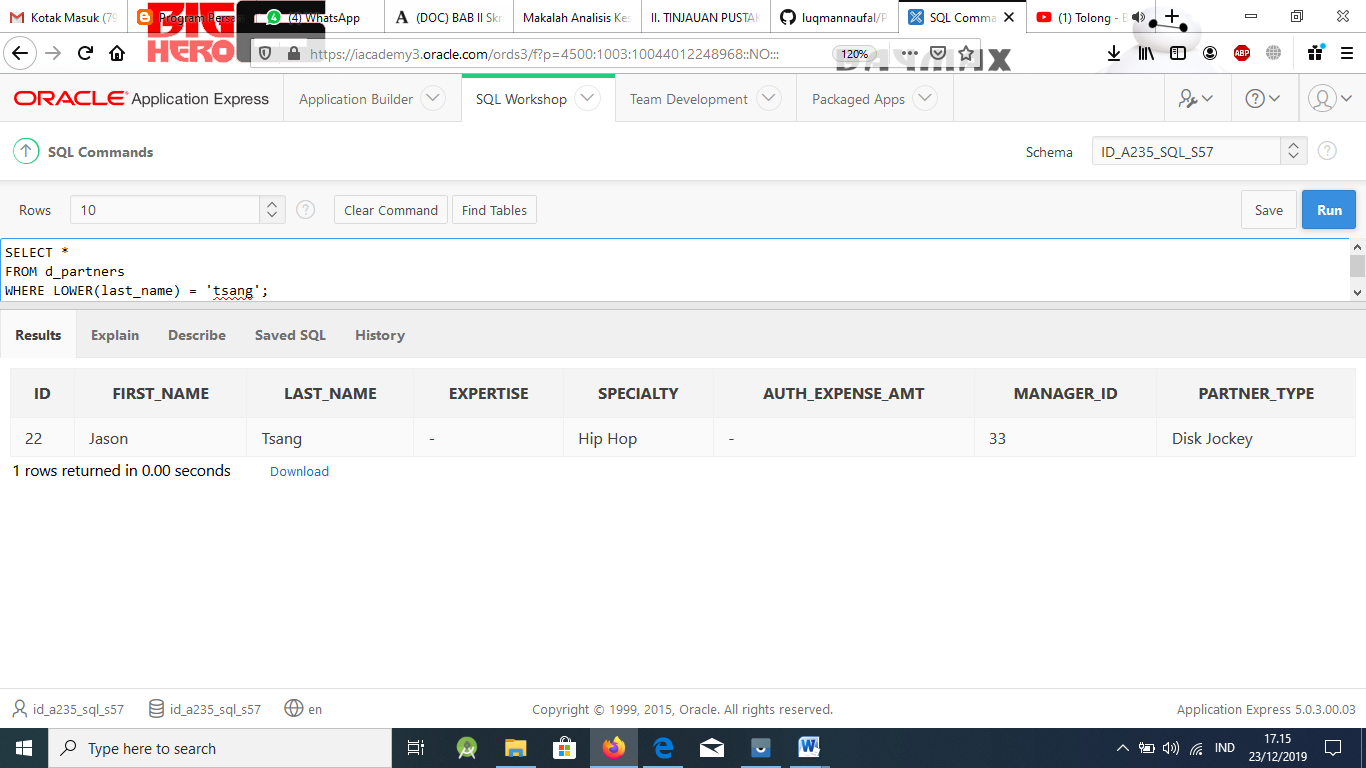
CREATE INDEX lower\_d\_partners\_idx ON d\_partners (LOWER(last\_name));



SELECT \*

FROM d\_partners

WHERE LOWER(last\_name) = 'tsang';



1. Create a synonym for the D\_TRACK\_LISTINGS table. Confirm that it has been created by querying the data dictionary.

## Solution:

CREATE SYNONYM *name* FOR d\_track\_listings

1. Drop the synonym that you created in question 9.

## Solution:

DROP SYNONYM *name*