PRACTICE SOLUTIONS

# SECTION 19 LESSON 1 – Testing

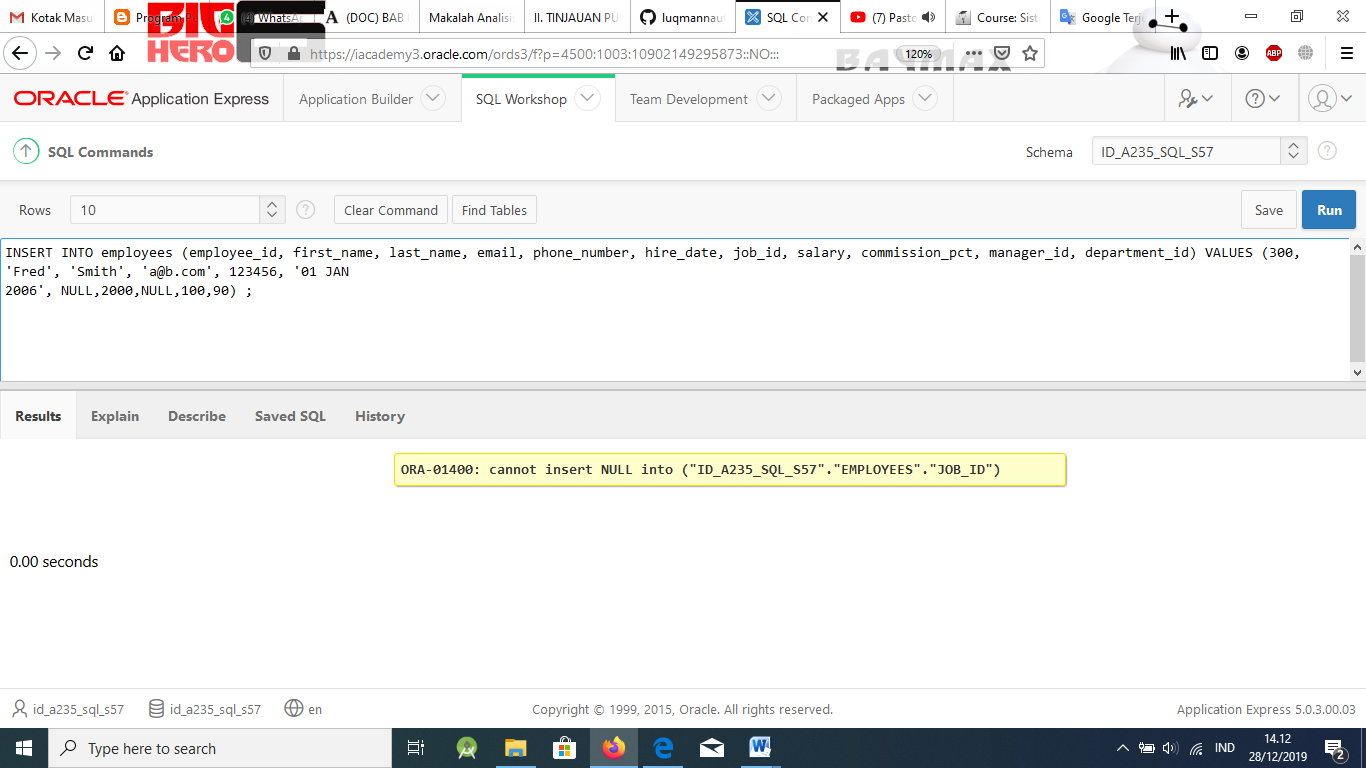
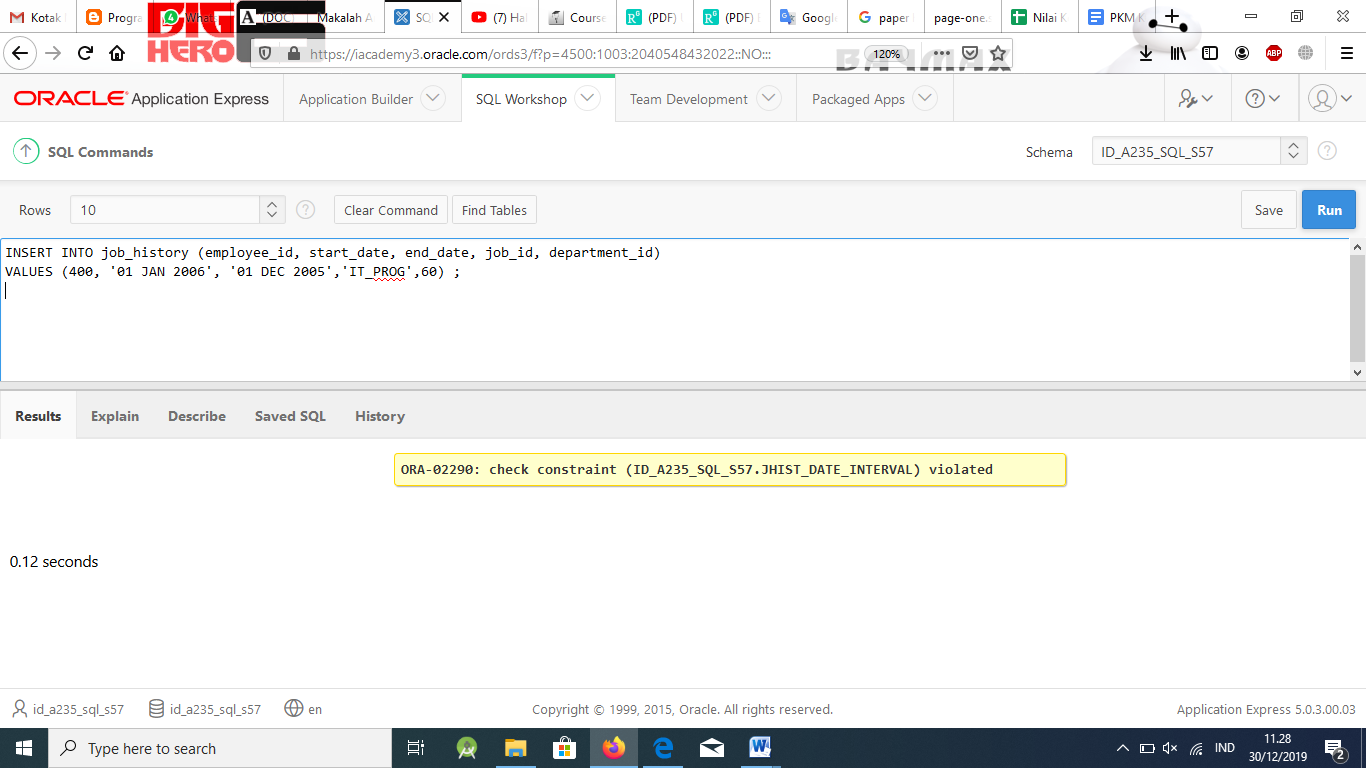
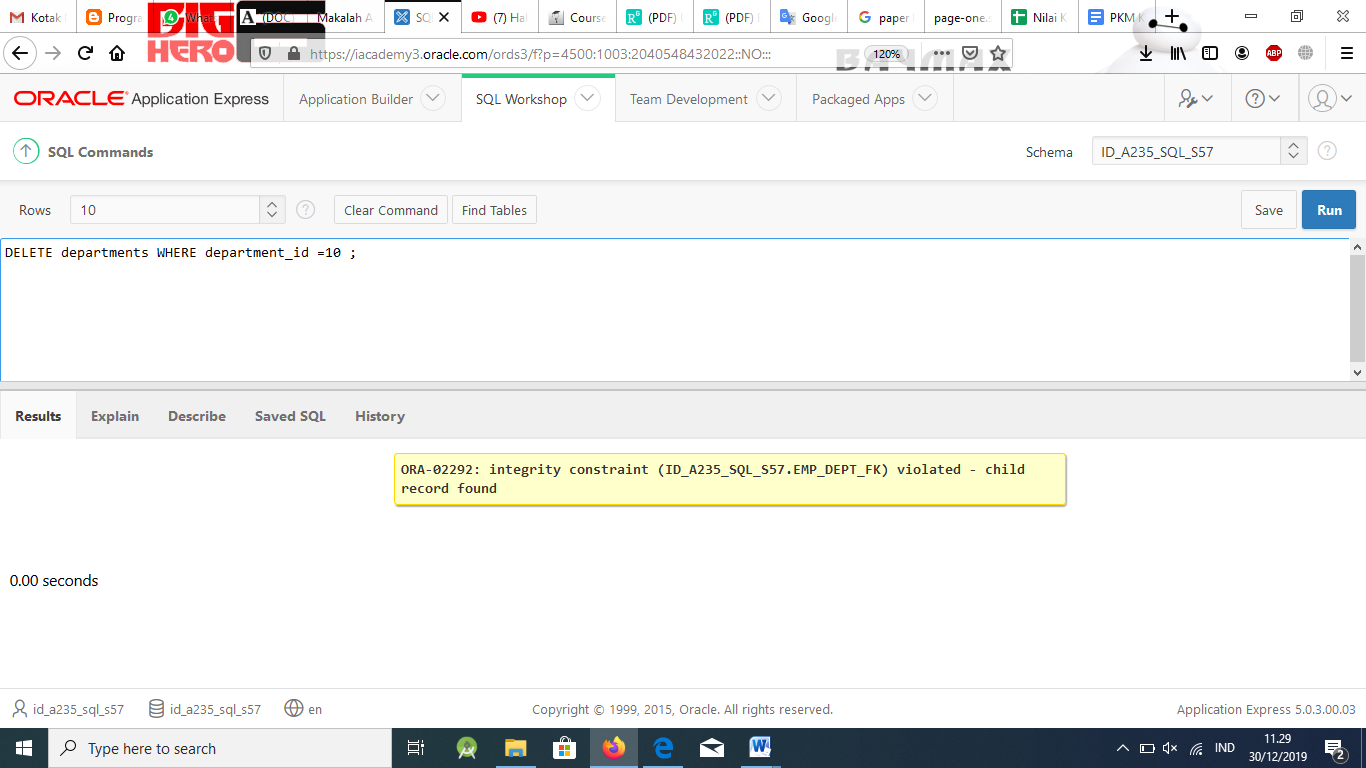
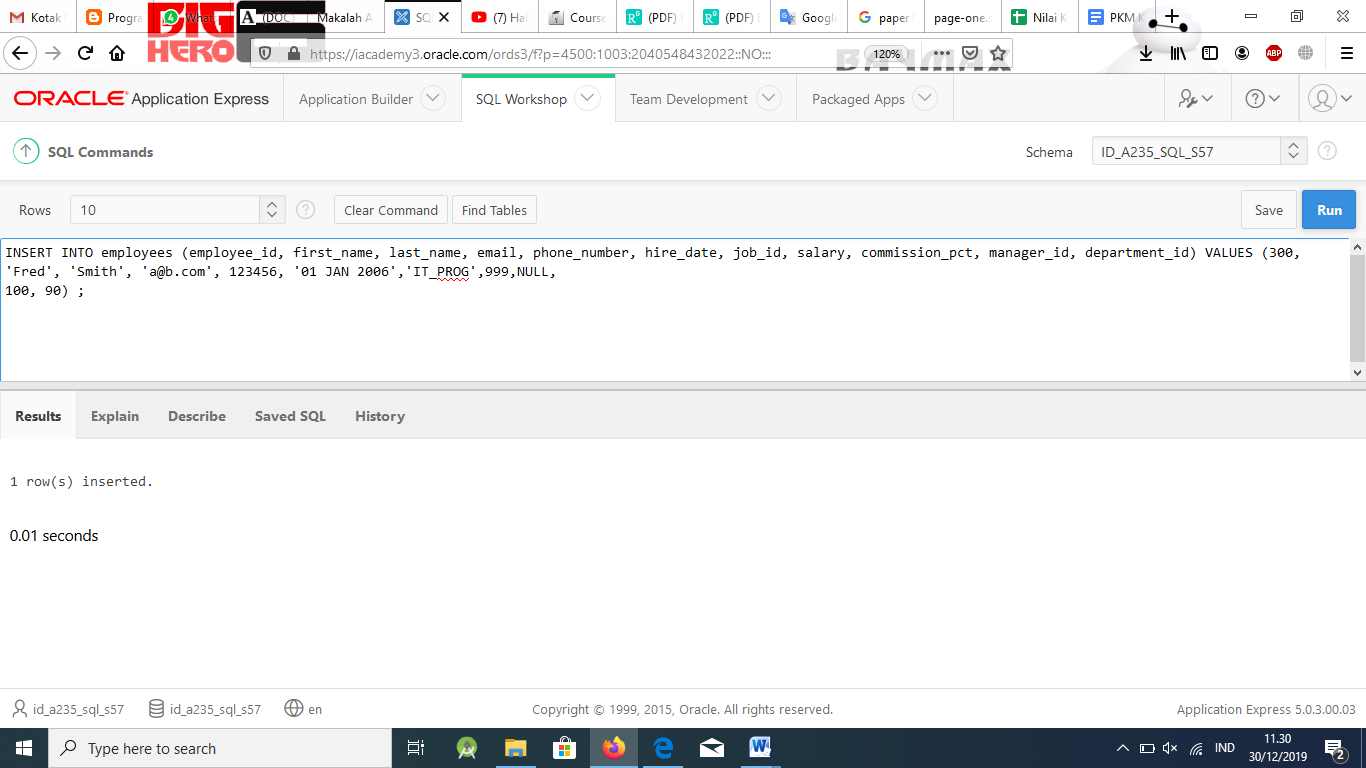
*Try It / Solve It*

1. Design and carry out tests to check the following:
   1. The business rule that requires that employees have a job\_id
   2. The business rule that requires that the end date of an employment is after a start date in the job history table.
   3. The business rule that states that departments can be closed down with employees in that department (resulting in the department\_id becoming unknown).
   4. The minimum salary of an employee is 1000.

# Solution:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Number | Date | Test Description | Input | Expected Output | Result/ Discrepancy | Action |
| 1 |  | a | INSERT INTO employees (employee\_id, first\_name, last\_name, email, phone\_number, hire\_date, job\_id, salary, commission\_pct, manager\_id, department\_id) VALUES (300, 'Fred', 'Smith', 'a@b.com', 123456, '01 JAN  2006', NULL,2000,NULL,100,90) ; | cannot insert NULL into JOB\_ID | cannot insert NULL into JOB\_ID | None |
| 2 |  | b | INSERT INTO job\_history (employee\_id, start\_date, end\_date, job\_id, department\_id)  VALUES (400, '01 JAN 2006', '01 DEC 2005','IT\_PROG',60) ; | Check constraint violated | Check constraint violated | None |
| 3 |  | c | DELETE departments WHERE department\_id =10 ; | 1 row deleted | integrity constraint violated - child record found | Alter referential  integrity constraint |

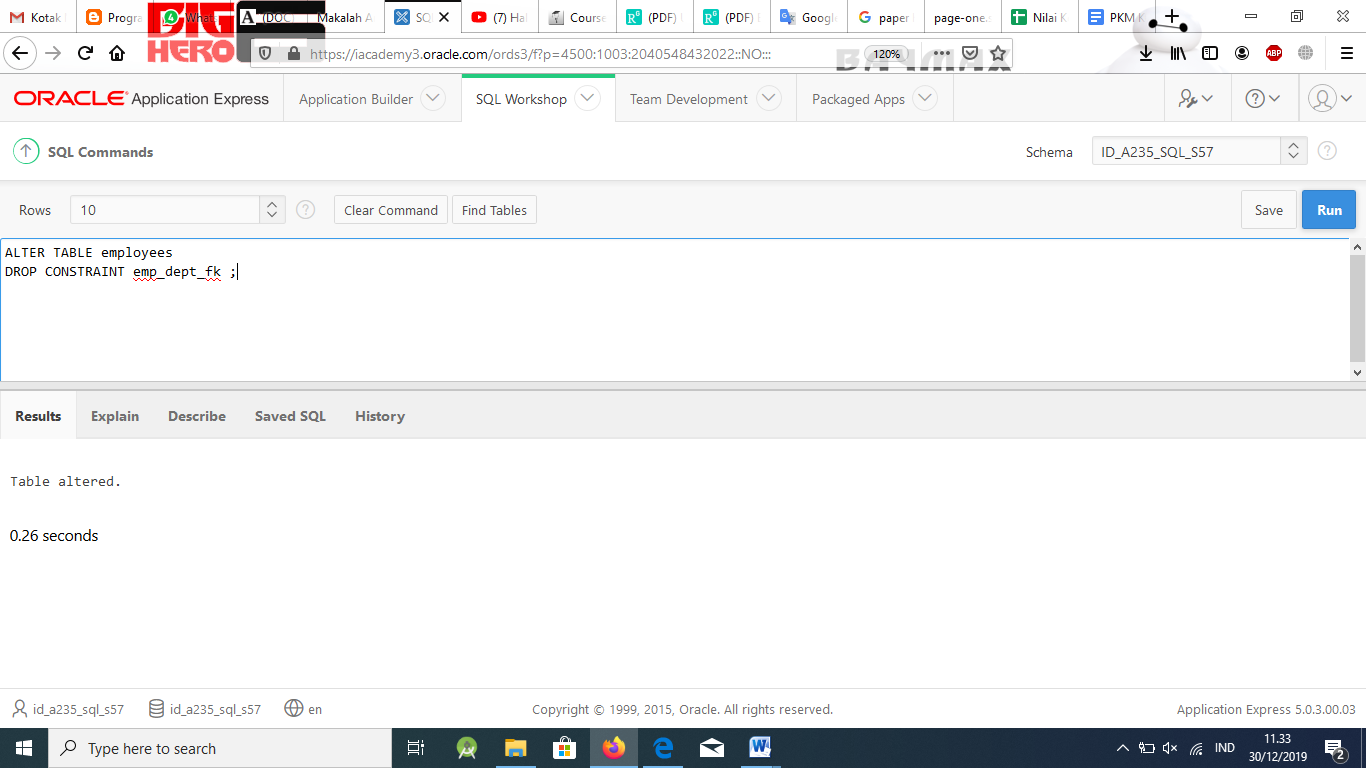
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 4 |  | d | INSERT INTO employees (employee\_id, first\_name, last\_name, email, phone\_number, hire\_date, job\_id, salary, commission\_pct, manager\_id, department\_id) VALUES (300, 'Fred', 'Smith', 'a@b.com', 123456, '01 JAN 2006','IT\_PROG',999,NULL,  100, 90) ; | Check constraint violated | 1 row inserted | Alter check constraint |

1. 
2. 
3. 
4. 
5. If one of the above tests fails, write out the SQL statement(s) that would be needed to correct the test. With the permission of your teacher, implement the change and then rerun the test with the same input and confirm that it works.

# Solution:

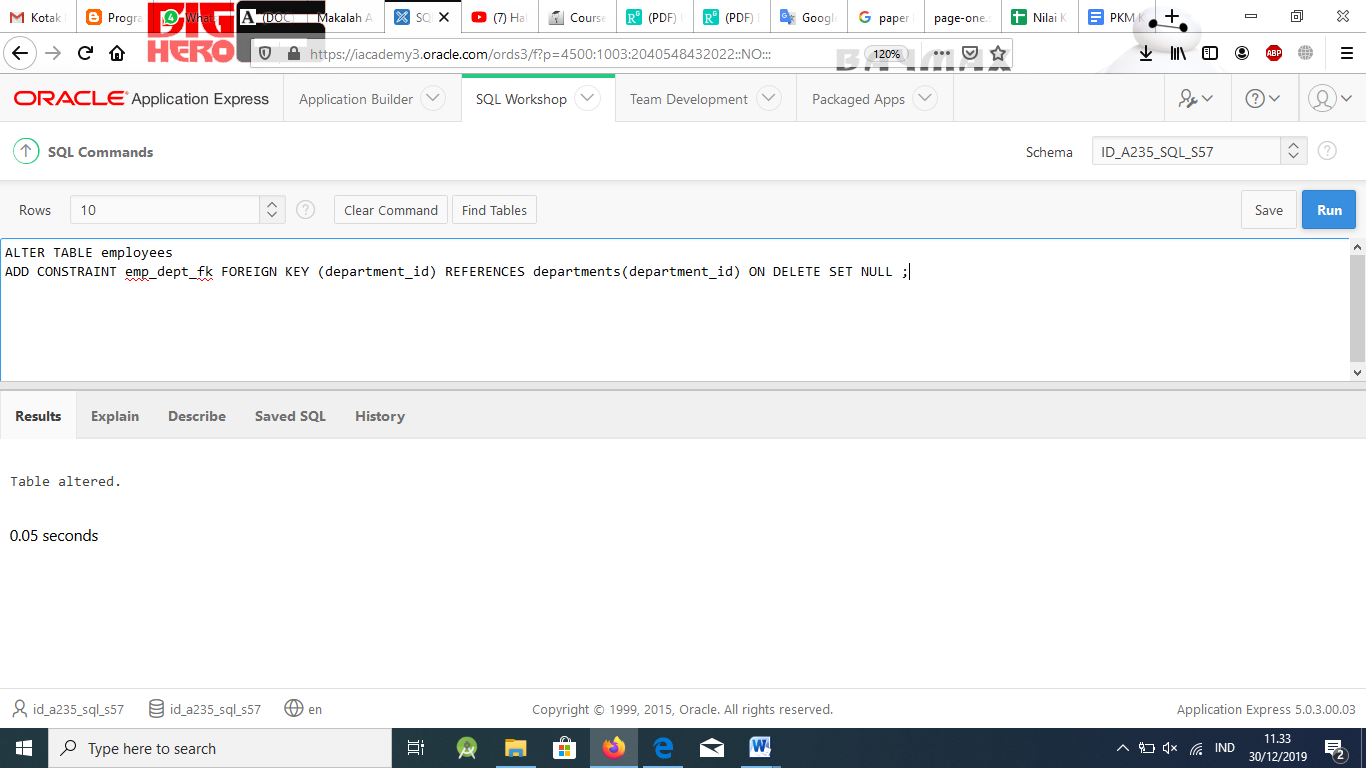
c. ALTER TABLE employees

DROP CONSTRAINT emp\_dept\_fk ;



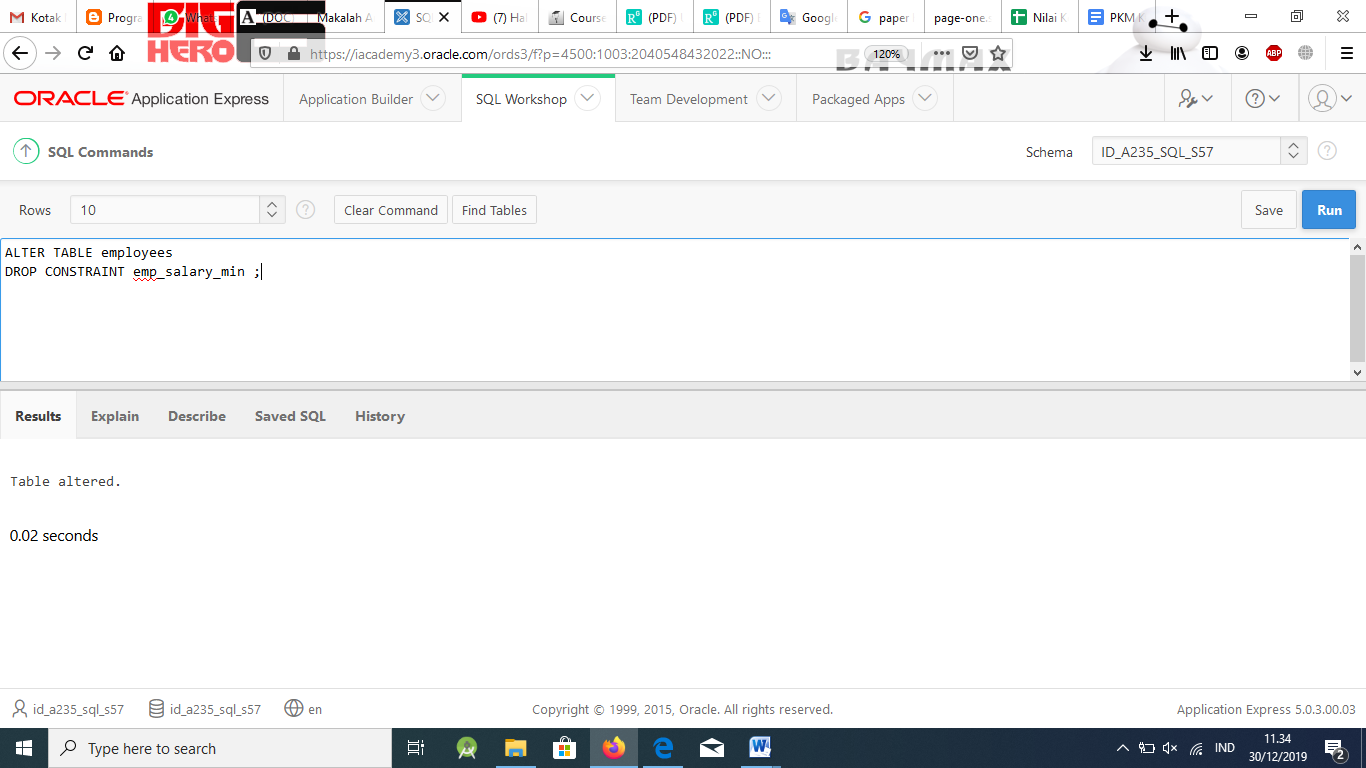
ALTER TABLE employees

ADD CONSTRAINT emp\_dept\_fk FOREIGN KEY (department\_id) REFERENCES departments(department\_id) ON DELETE SET NULL ;



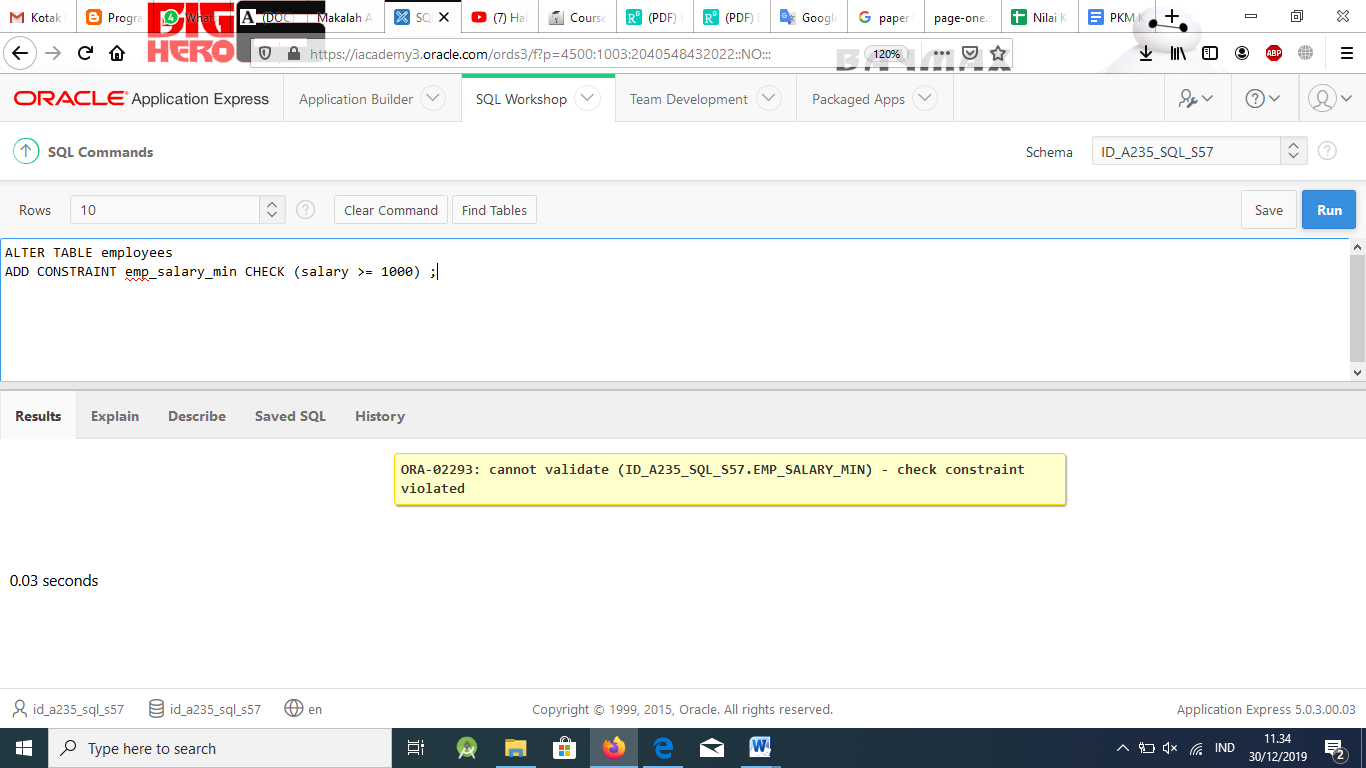
d. ALTER TABLE employees

DROP CONSTRAINT emp\_salary\_min ;



ALTER TABLE employees

ADD CONSTRAINT emp\_salary\_min CHECK (salary >= 1000) ;



PRACTICE SOLUTIONS

# SECTION 19 LESSON 3 – Final Exam Review

*Try It / Solve It*

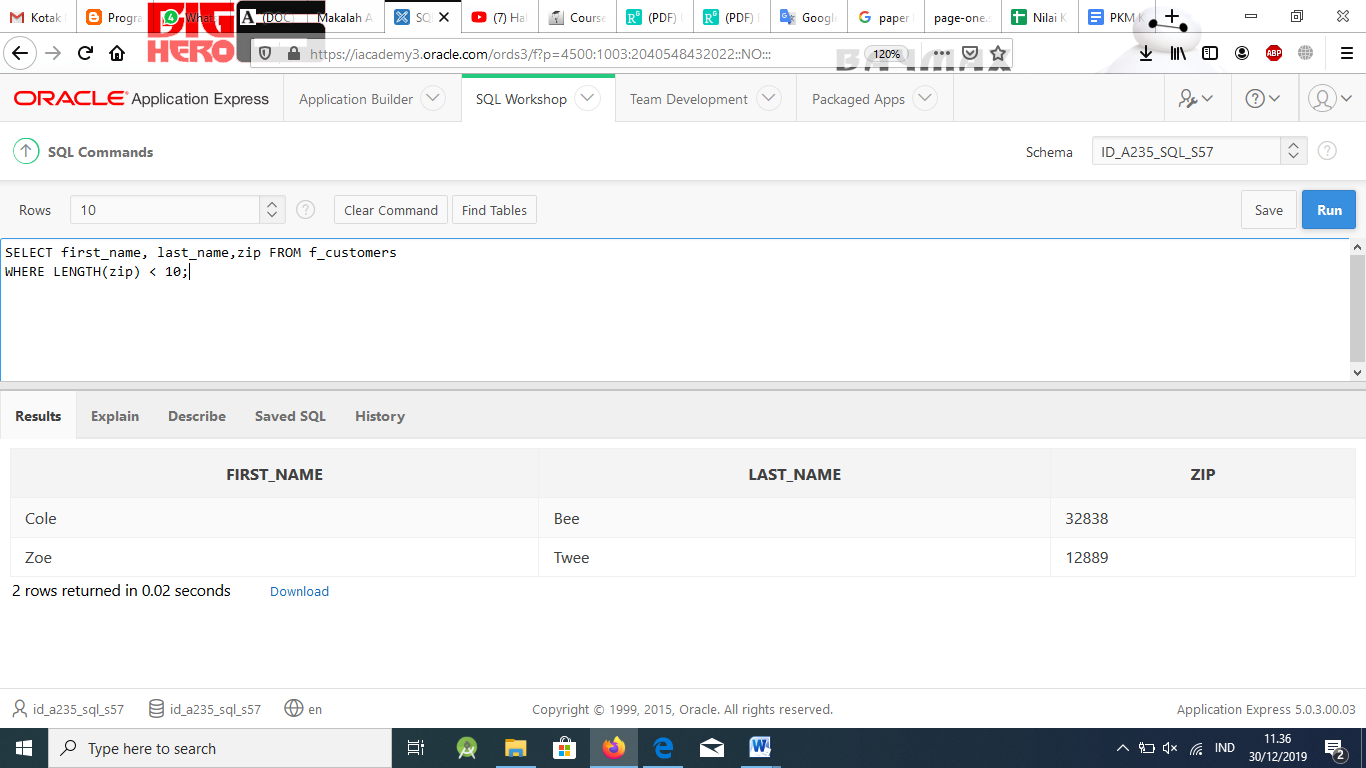
**Part 1**

1. The business manager of Global Fast Foods needs to update the customer list. She wants to find any zip/postal code that has fewer than 10 digits in order to identify those codes without the new format postfix 87392-8723. Create a query to identify those customers.

# Solution:

SELECT first\_name, last\_name,zip FROM f\_customers

WHERE LENGTH(zip) < 10;



1. In the following list of functions, mark with N those that can be used with numbers, mark with C those that can operate on character data, and mark with D those that can be used with dates.
   1. LPAD
   2. ROUND
   3. TRUNC
   4. LENGTH
   5. LAST\_DAY
   6. INSTR
   7. CONCAT

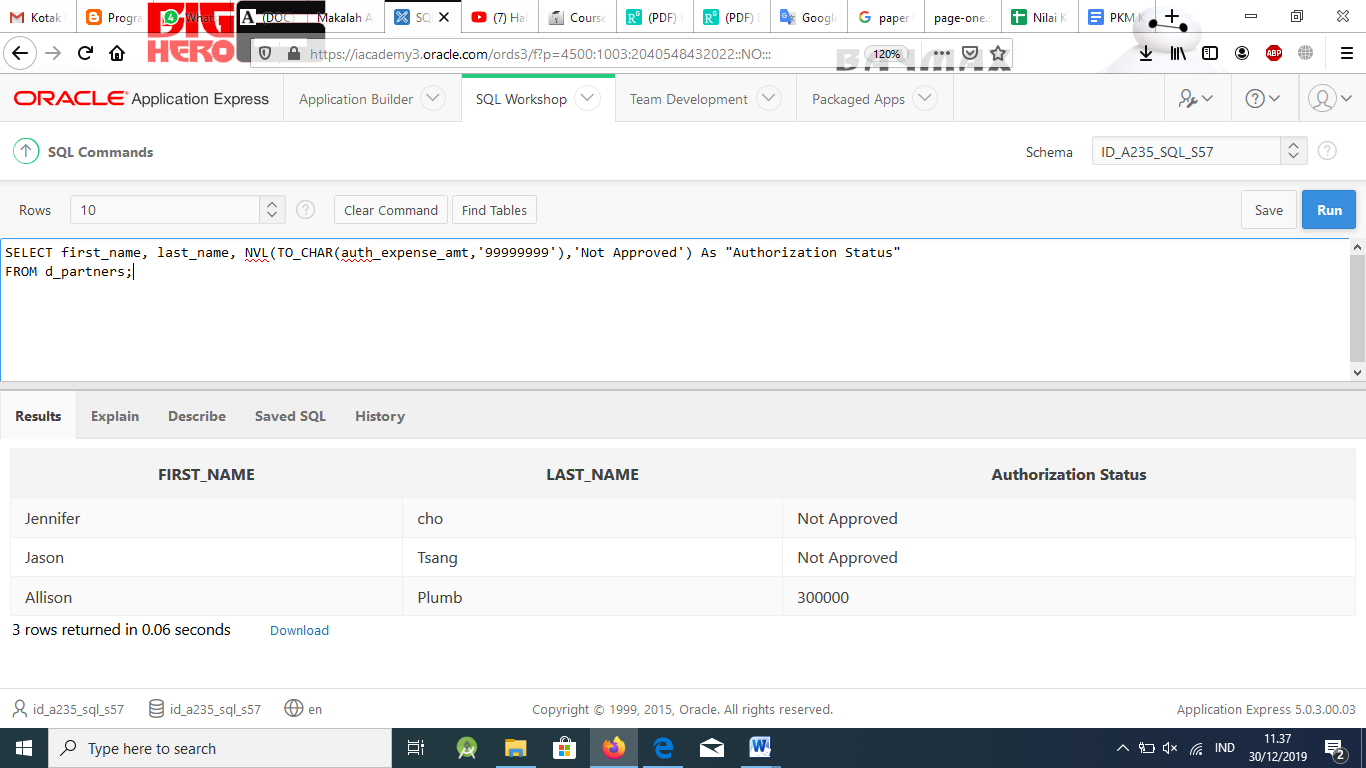
# Solution:

1. LPAD (CDN)
2. ROUND (DN)
3. TRUNC (DN)
4. LENGTH (C)
5. LAST\_DAY (D)
6. INSTR (CDN)
7. CONCAT (CDN)
8. You need to display the auth\_expense\_amt for each DJs on Demand partner. For those partners who do not have an expense account, the output should display “Not Approved.”

# Solution:

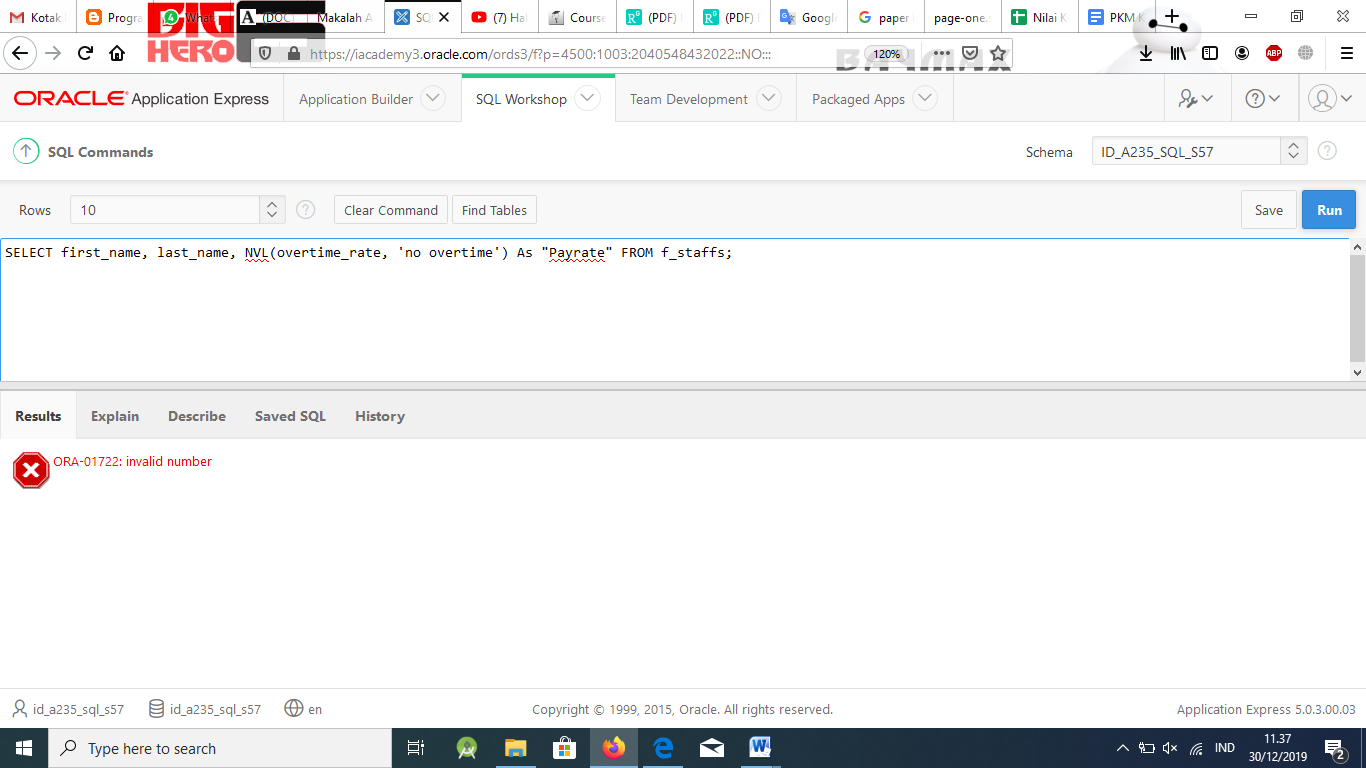
SELECT first\_name, last\_name, NVL(TO\_CHAR(auth\_expense\_amt,'99999999'),'Not Approved') As "Authorization Status"

FROM d\_partners;



1. Jason and Jamie tried to run a report displaying the Global Fast Foods staff members who do not have an overtime rate. They wrote the following SQL query and received a report error: ORA-01722: invalid number.

SELECT first\_name, last\_name, NVL(overtime\_rate, 'no overtime') As "Payrate" FROM f\_staffs;



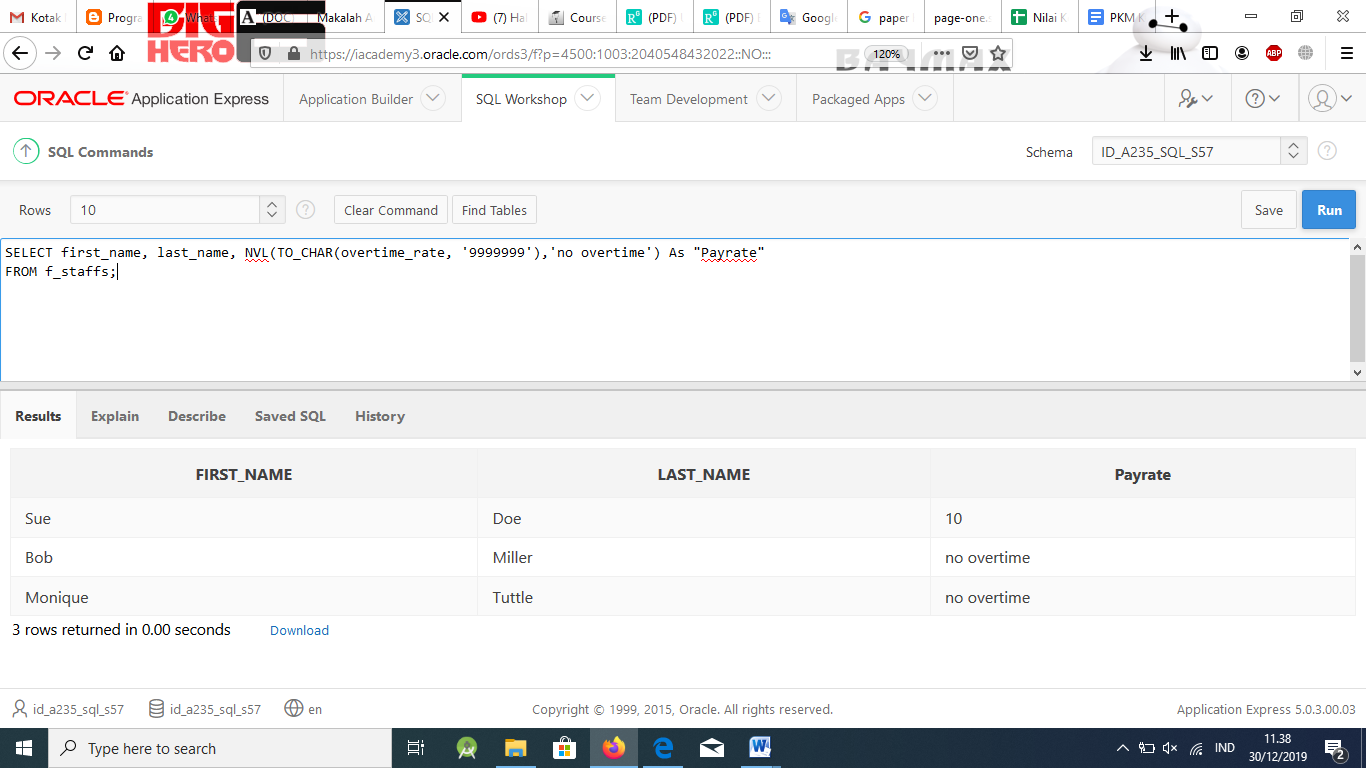
What is wrong with their query and how can it be fixed?

# Solution:

Jason and Jamie forgot that the ‘s of the NVL elements must be the same’! A TO\_CHAR function is needed to convert the number overtime\_rate to a character value.

SELECT first\_name, last\_name, NVL(TO\_CHAR(overtime\_rate, '9999999'),'no overtime') As "Payrate"

FROM f\_staffs;



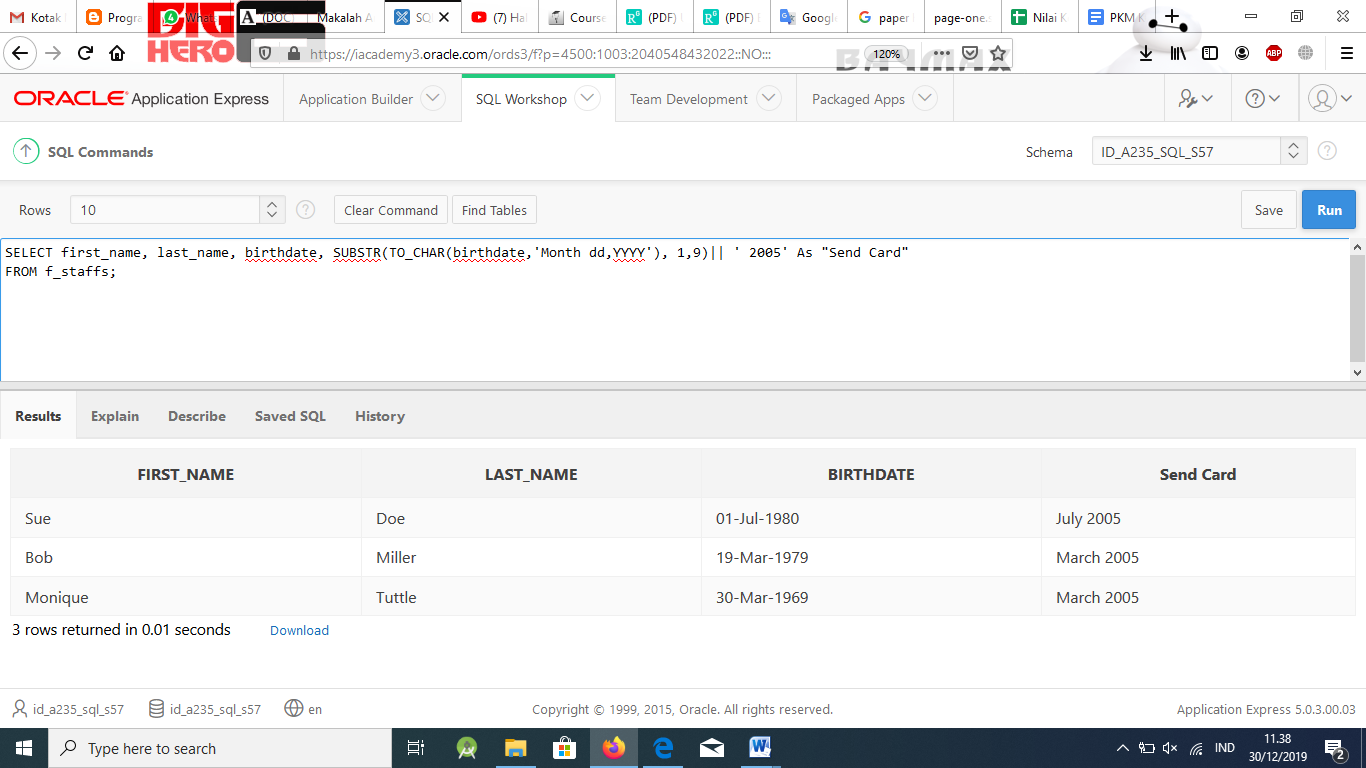
1. The president of Global Fast Foods likes to send a birthday card to all employees. He has asked you to send a reminder to him on the month of the employee’s birthday so the card can be sent with that month’s paycheck. Prepare a query to produce the table shown.

|  |  |  |  |
| --- | --- | --- | --- |
| FIRST\_NAME | LAST\_NAME | BIRTHDATE | SEND CARD |
| Sue | Doe | 01-JUL-1980 | July 2005 |
| Bob | Miller | 19-MAR-1979 | March 2005 |
| Monique | Tuttle | 30-MAR-1969 | March 2005 |

# Solution:

SELECT first\_name, last\_name, birthdate, SUBSTR(TO\_CHAR(birthdate,'Month dd,YYYY'), 1,9)|| ' 2005' As "Send Card"

FROM f\_staffs;



1. For each statement, mark T if the statement is true or F if the statement is false.
   1. TO\_CHAR is required to convert the date '03-JUN-04' to June 3, 2004.
   2. TO\_NUMBER will convert '23-NOV-02' to use with ADD\_MONTHS.
   3. TO\_DATE will convert SYSDATE to today’s date.
   4. TO\_NUMBER('101', '$99999') will convert 101 to a number.
   5. TO\_CHAR(salary, '$9999.99') will convert number to character format.
   6. TO\_NUM(varchar2 column) will convert character data to a number.
   7. TO\_CHAR(SYSDATE, 'Month fmdd, yyyy') will format the date.

# Solution:

T a. TO\_CHAR is required to convert the date '03-JUN-04' to June 3, 2004

F b. TO\_NUMBER will convert '23-NOV-02' to use with ADD\_MONTHS

F c. TO\_DATE will convert SYSDATE to today's date

F d. TO\_NUMBER('101', '$99999') will convert 101 to a number

T e. TO\_CHAR(salary, '$9999.99') will convert number to character format

T f. TO\_NUM(varchar2 column) will convert character data to a number

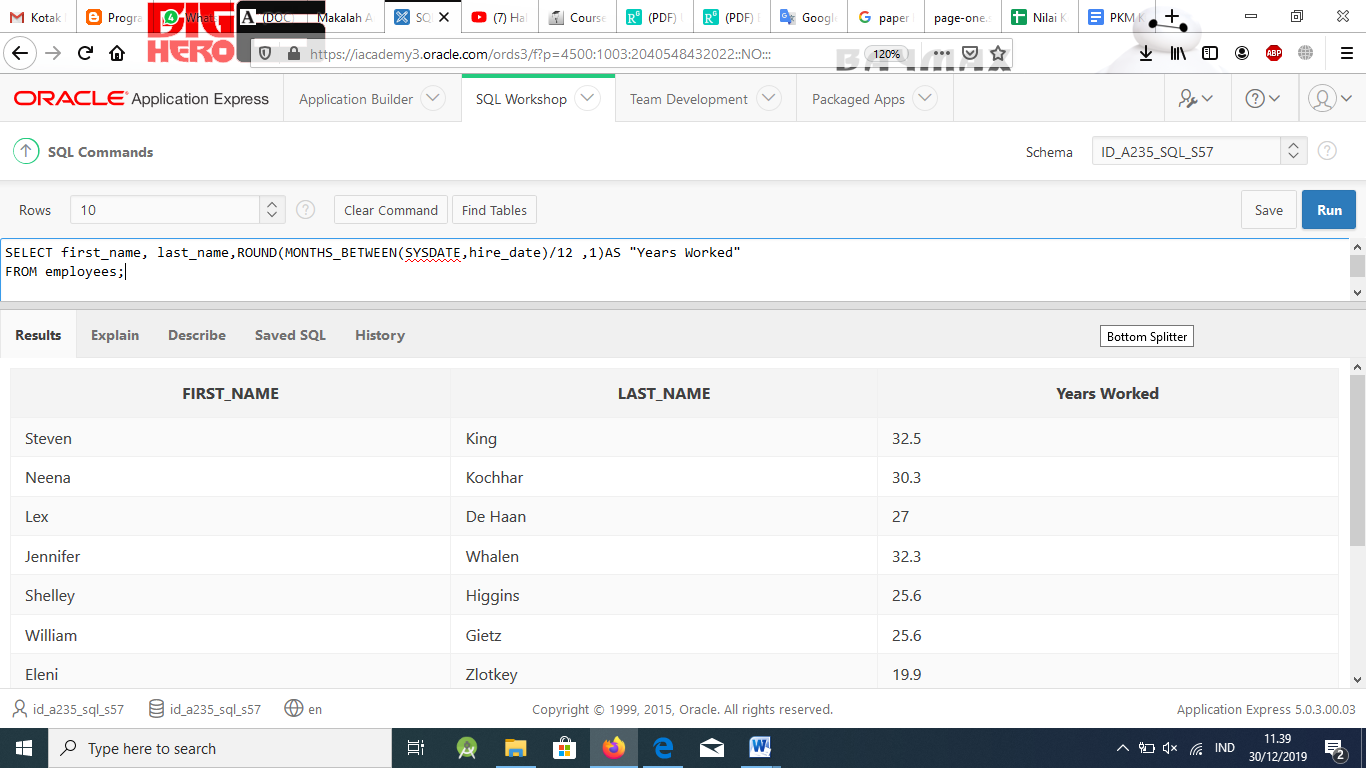
T g. TO\_CHAR(SYSDATE, 'Month fmdd, yyyy') will format the date

1. Employees in the Oracle database who have worked more than 10 years will be given one extra week of vacation. Create a report showing first name, last name, and years worked. Round the result to one decimal place.

# Solution:

SELECT first\_name, last\_name,ROUND(MONTHS\_BETWEEN(SYSDATE,hire\_date)/12 ,1)AS "Years Worked"

FROM employees;



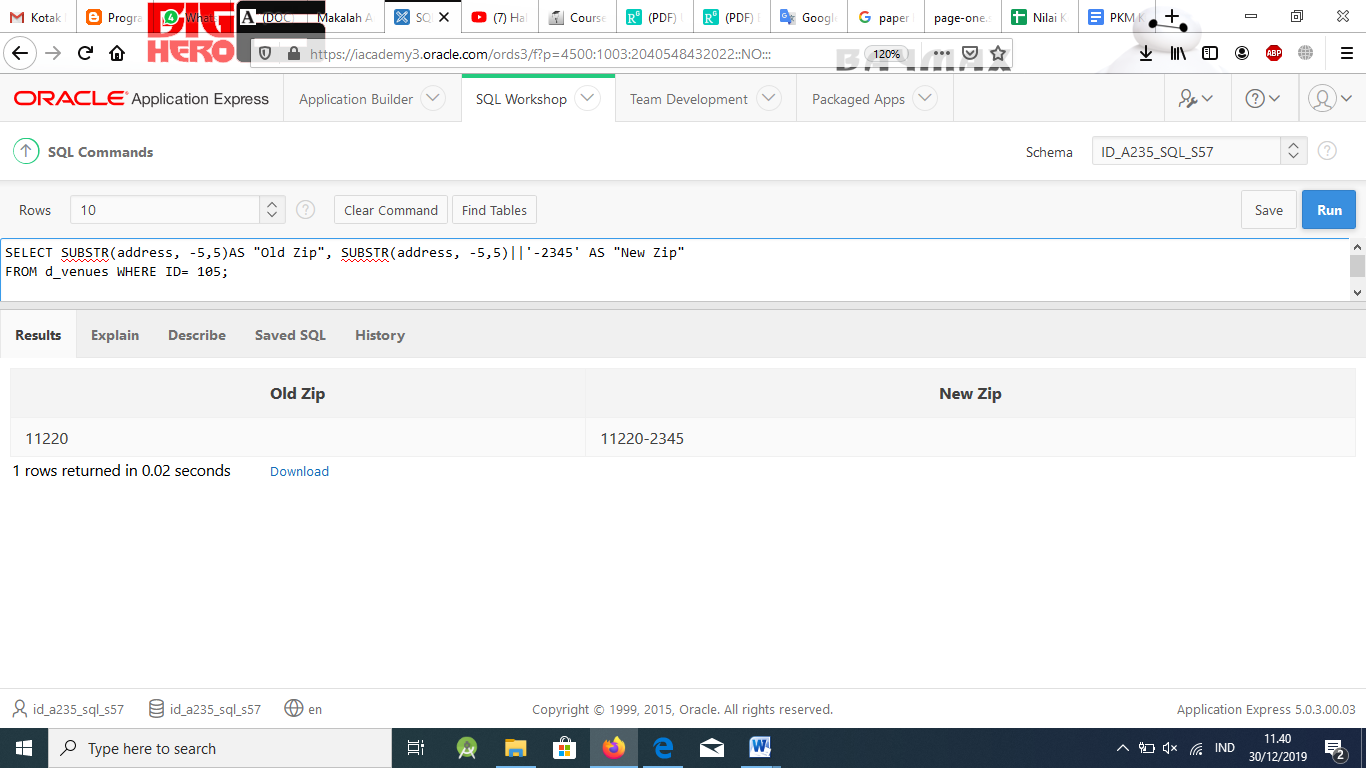
1. The manager of DJs on Demand needs you to change the zip code for ID 105, New York venues. She wants to show the old zip code in one column and the new zip code in another column. The new zip code is the same as the old zip code except -2345 needs to be added to the end. The output should appear as shown in the table.

|  |  |
| --- | --- |
| OLD ZIP | NEW ZIP |
| 11220 | 11220-2345 |

# Solution:

SELECT SUBSTR(address, -5,5)AS "Old Zip", SUBSTR(address, -5,5)||'-2345' AS "New Zip"

FROM d\_venues WHERE ID= 105;

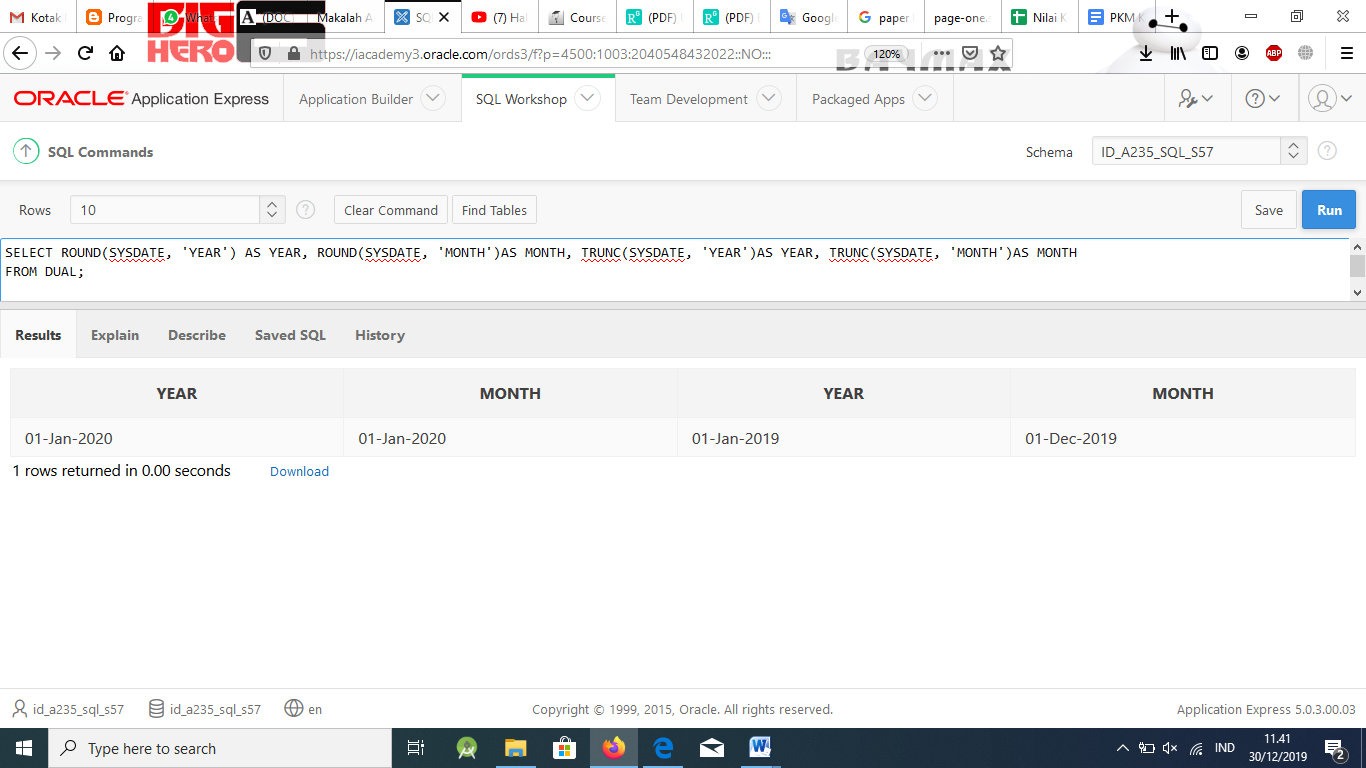


1. Create a query using one SELECT statement that returns today’s date. Assign an alias to each column.
   1. Rounded to the nearest year
   2. Rounded to the nearest month
   3. Truncated to the nearest year
   4. Truncated to the nearest month

# Solution:

SELECT ROUND(SYSDATE, 'YEAR') AS YEAR, ROUND(SYSDATE, 'MONTH')AS MONTH, TRUNC(SYSDATE, 'YEAR')AS YEAR, TRUNC(SYSDATE, 'MONTH')AS MONTH

FROM DUAL;

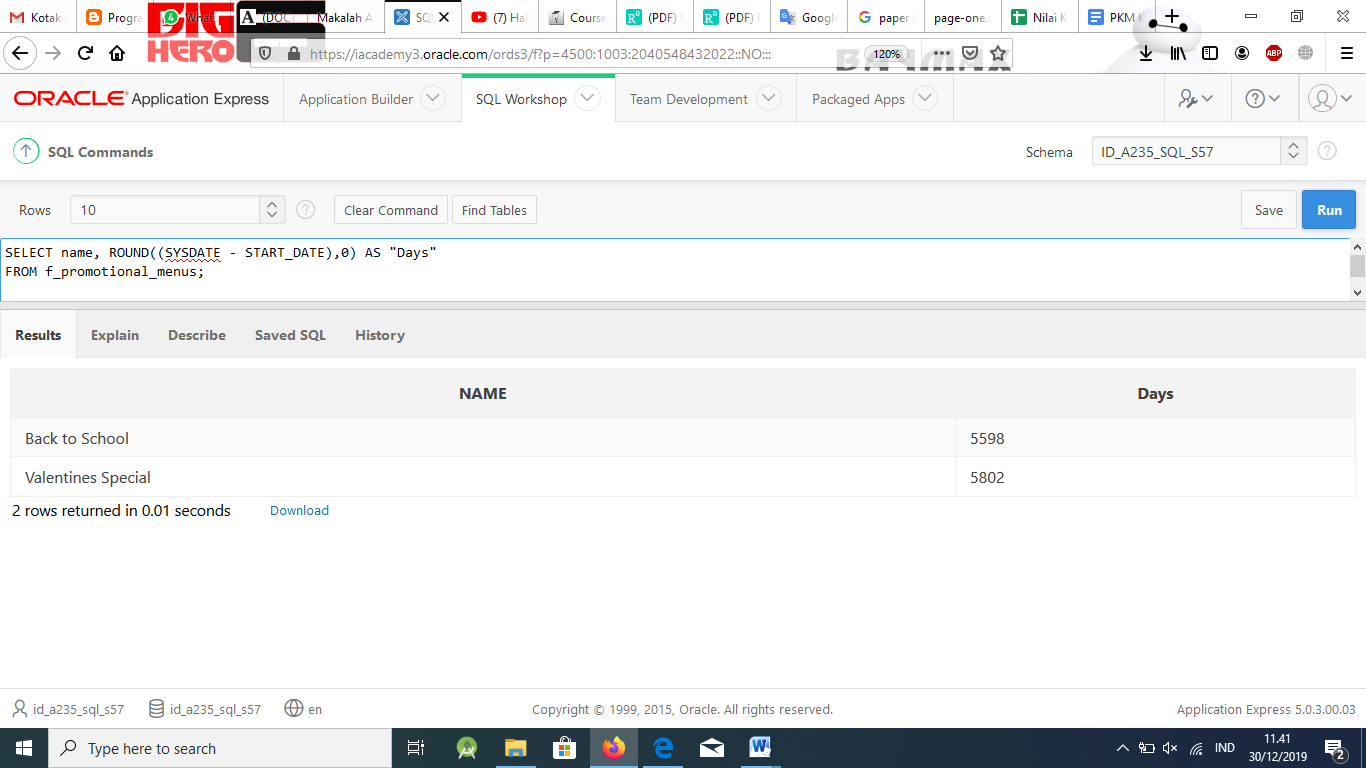


1. You need to find out how many days it has been since the start of the Global Fast Foods promotional menus. Round the result to the nearest day.

# Solution:

SELECT name, ROUND((SYSDATE - START\_DATE),0) AS "Days"

FROM f\_promotional\_menus;

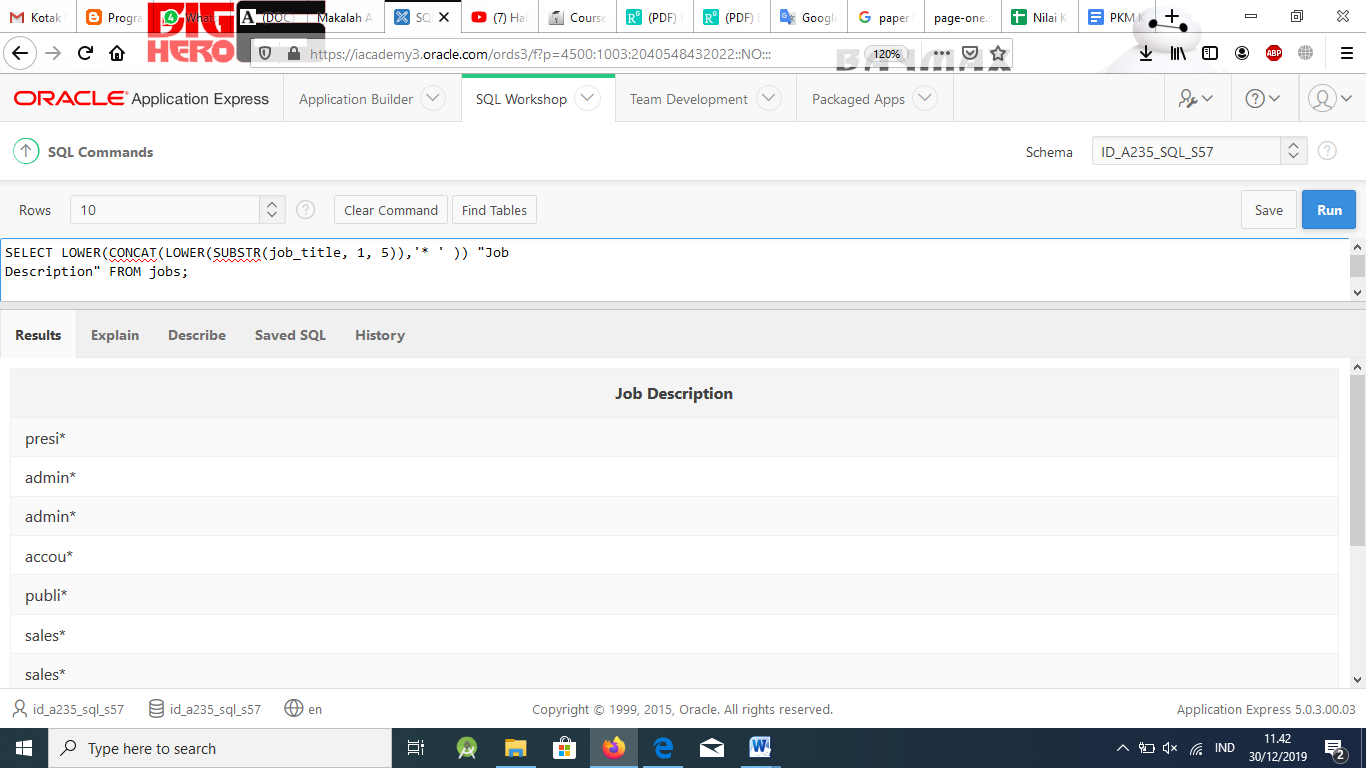


1. The Human Resources department (Oracle database) has decided that, for their purposes, the job title for all employees will be the first five letters of the job title followed by an asterisk. For example, the accounting manager will be changed to accou\* Create a query to accomplish this request.

# Solution:

SELECT LOWER(CONCAT(LOWER(SUBSTR(job\_title, 1, 5)),'\* ' )) "Job

Description" FROM jobs;



1. What is the order of operations in question 11?

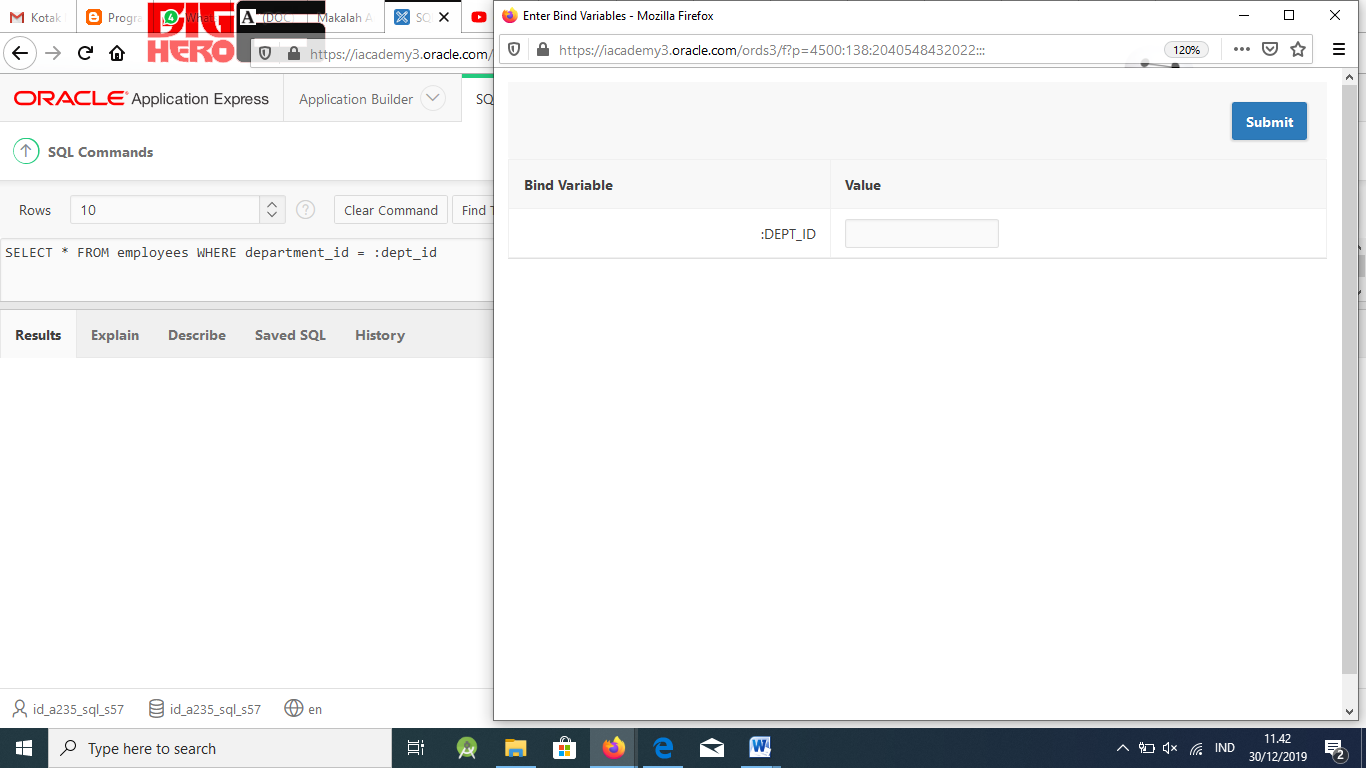
# Solution:

The expression will be evaluated from the innermost expression outward.

1. Write a query to return all the rows and columns from the employees table, but make the department\_id a substitution variables, and then execute your query with two or three different department\_id’s to test it.

# Solution:

SELECT \* FROM employees WHERE department\_id = :dept\_id



**Part 2**

1. Use the Oracle database employees table. Write and execute one query for each term.

LOWER UPPER INITCAP CONCAT SUBSTR LENGTH INSTR LPAD RPAD TRIM REPLACE ROUND TRUNC MOD

MONTHS\_BETWEEN ADD\_MONTHS

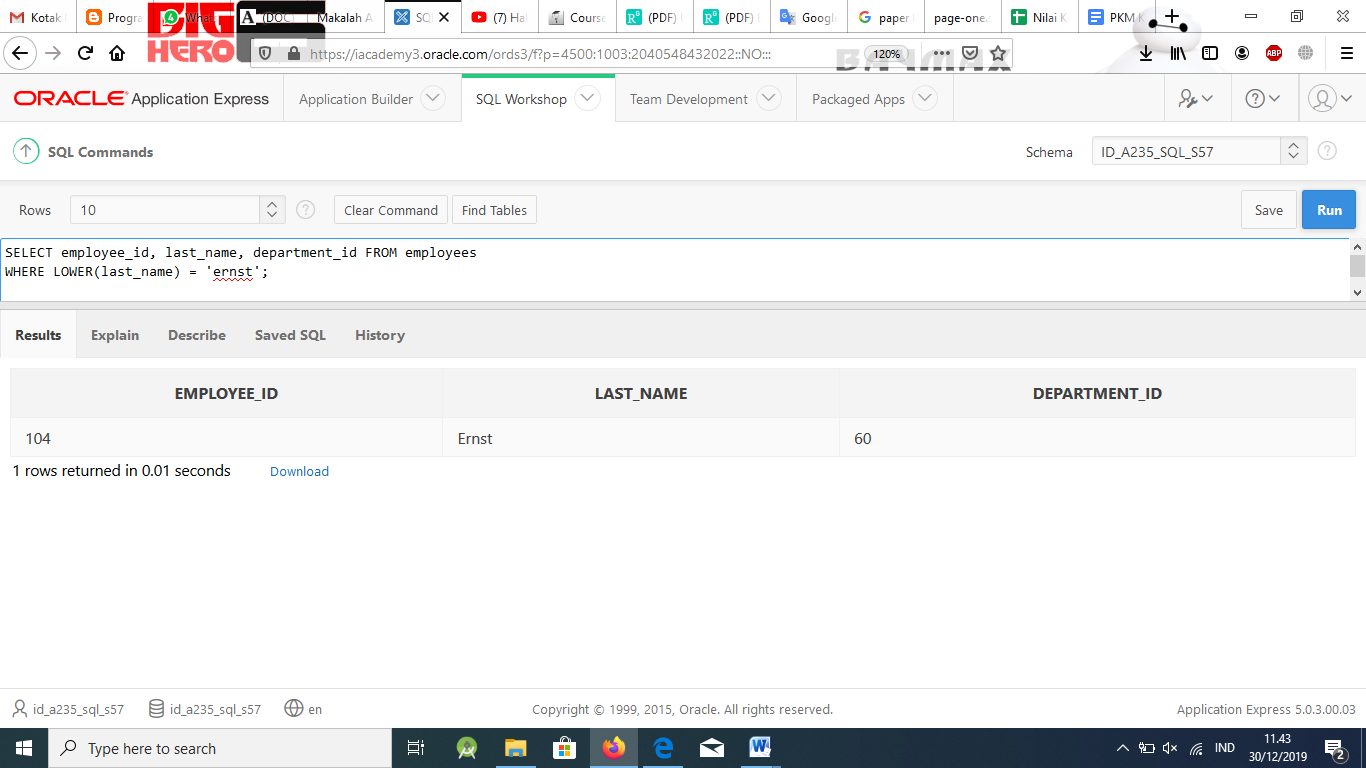
NEXT\_DAY LAST\_DAY SYSDATE

# Solution:

These are just a few example queries. Students should run their queries against the database to check their accuracy.

SELECT employee\_id, last\_name, department\_id FROM employees

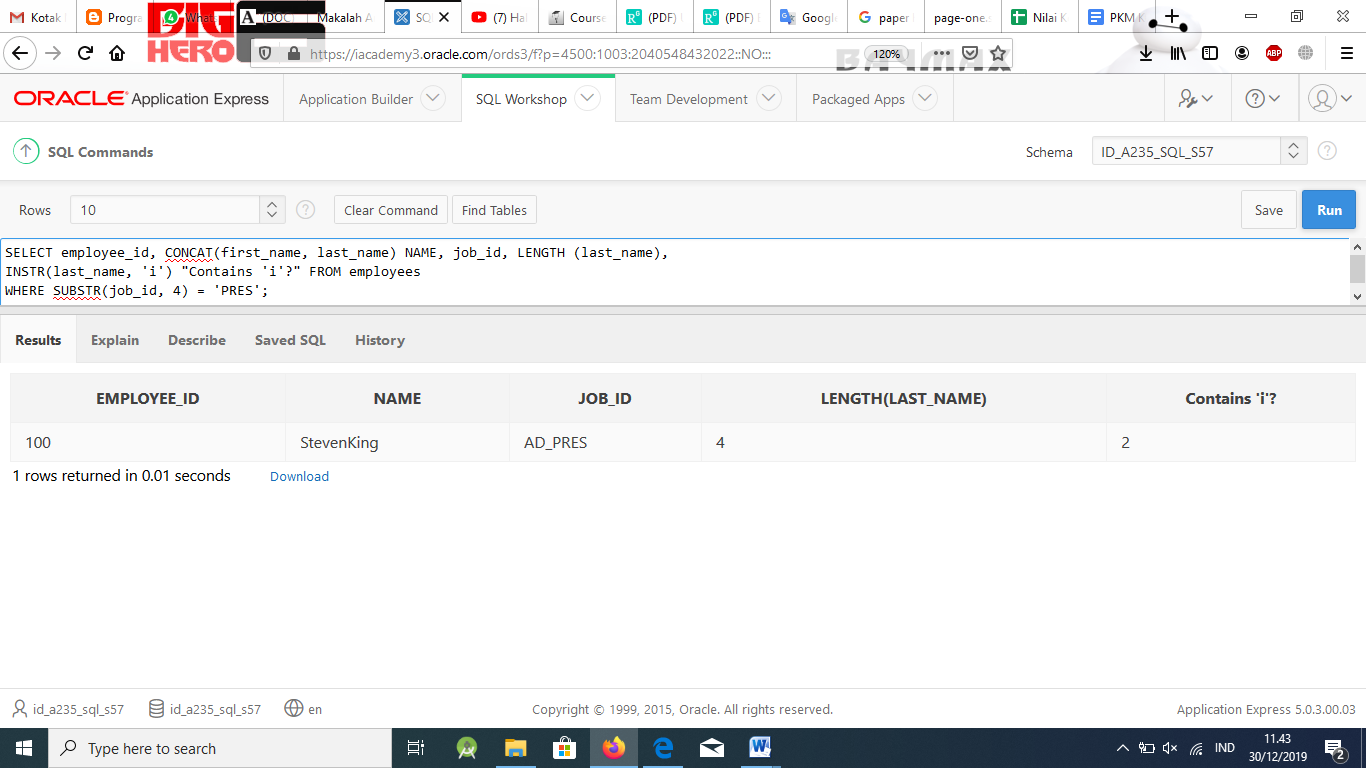
WHERE LOWER(last\_name) = 'ernst';



SELECT employee\_id, CONCAT(first\_name, last\_name) NAME, job\_id, LENGTH (last\_name),

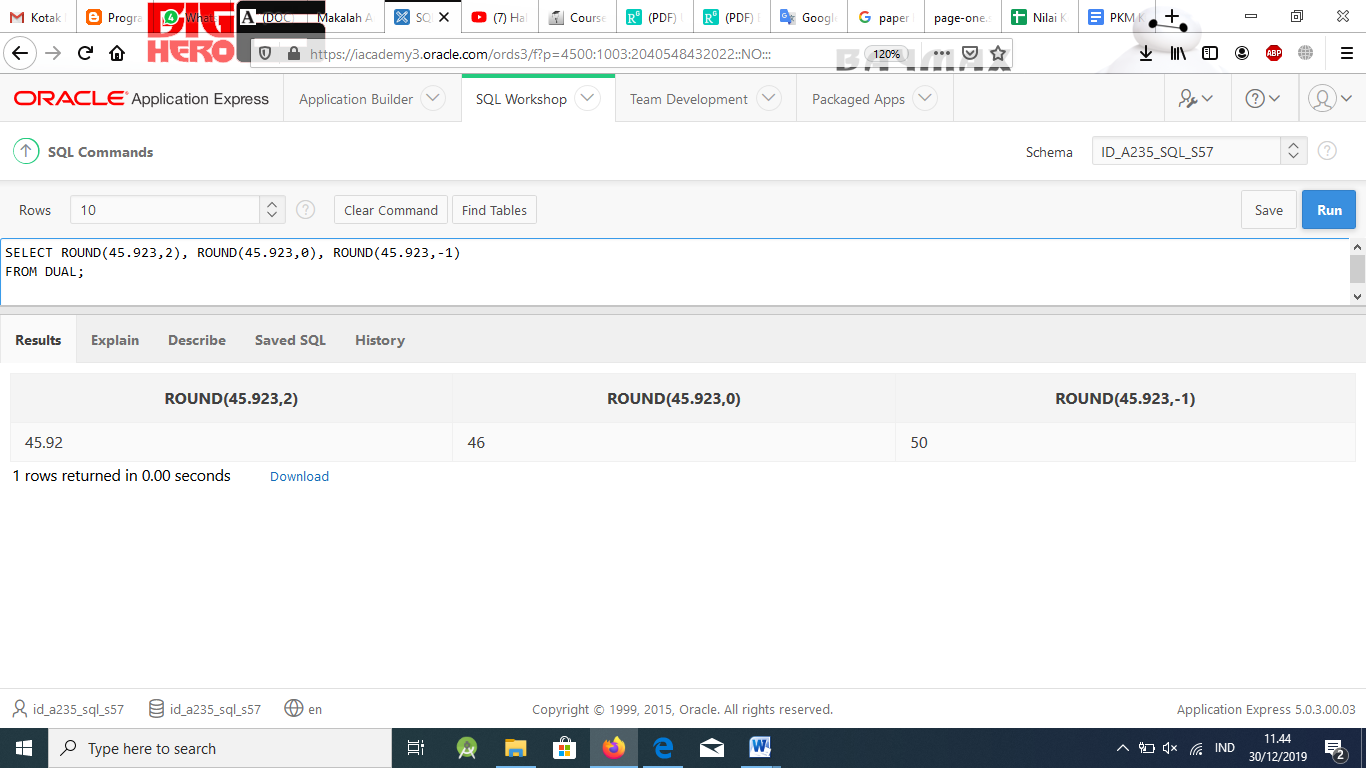
INSTR(last\_name, 'i') "Contains 'i'?" FROM employees

WHERE SUBSTR(job\_id, 4) = 'PRES';



SELECT ROUND(45.923,2), ROUND(45.923,0), ROUND(45.923,-1)

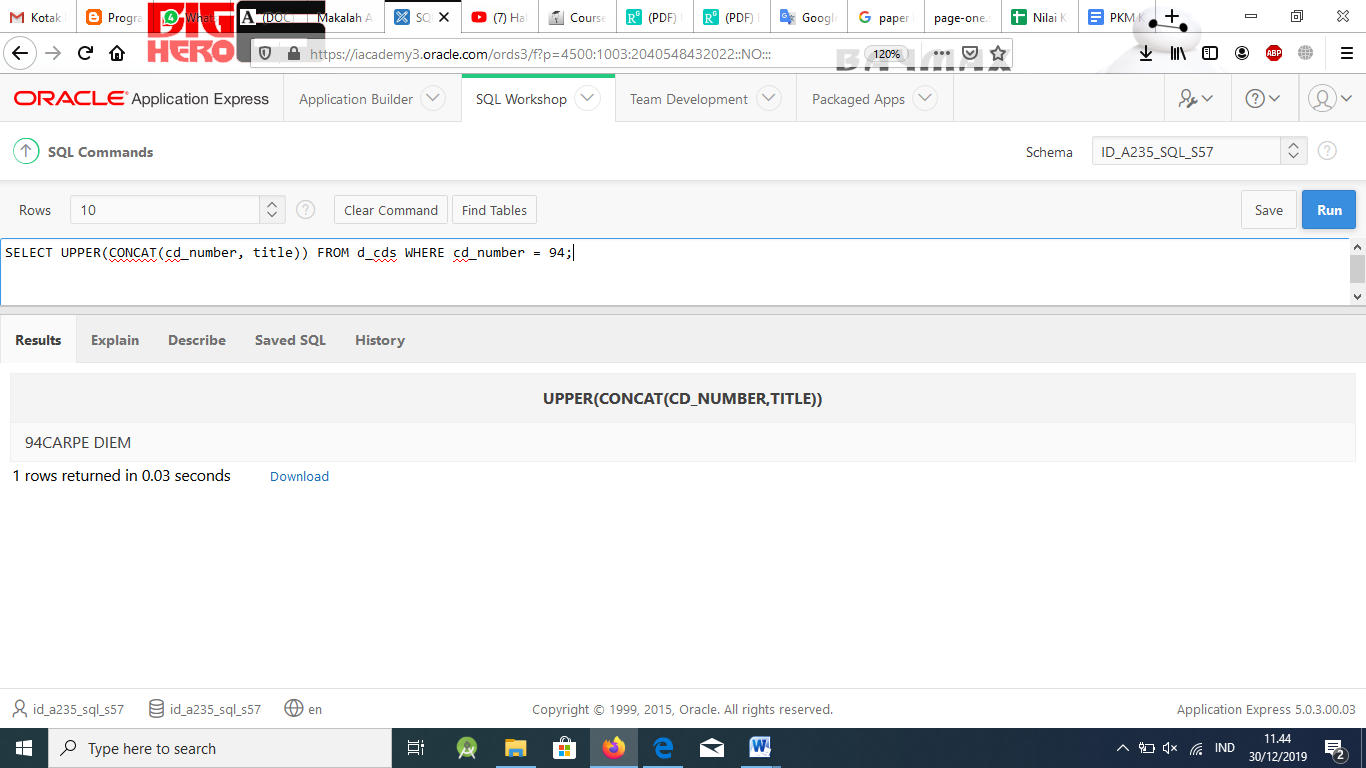
FROM DUAL;



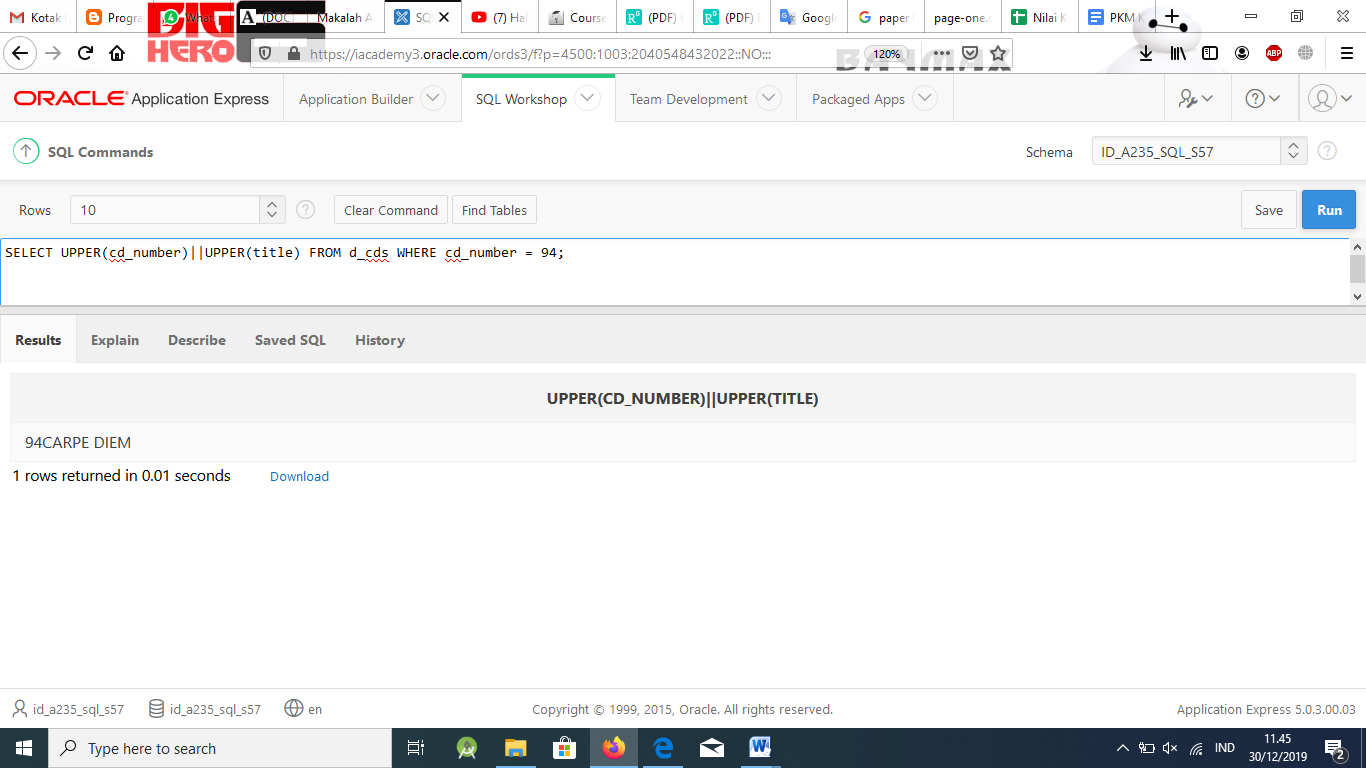
1. Create two different queries that produce the cd number and title as “94CARPE DIEM.” Use the d\_cds table in the DJs on Demand database.

# Solution:

SELECT UPPER(CONCAT(cd\_number, title)) FROM d\_cds WHERE cd\_number = 94;



SELECT UPPER(cd\_number)||UPPER(title) FROM d\_cds WHERE cd\_number = 94;



1. Mark the following statements as True or False.

a. LOWER converts numbers to lowercase.

b. Use RPAD to move numbers to the right to place an \* on the left.

c. TRIM can be used to trim one or more characters from a string.

d. LENGTH returns a number.

e. SUBSTR is used to substitute one string for another.

f. CONCAT is limited to using two parameters.

g. TRUNC will return zero decimal places if a decimal value is omitted.

# Solution:

F a. LOWER converts numbers to lowercase.

F b. Use RPAD to move numbers to the right to place an \* on the left.

F c. TRIM can be used to trim one or more characters from a string.

T d. LENGTH returns a number.

F e. SUBSTR is used to substitute one string for another.

T f. CONCAT is limited to using two parameters.

T g. TRUNC will return zero decimal places if a decimal value is omitted.

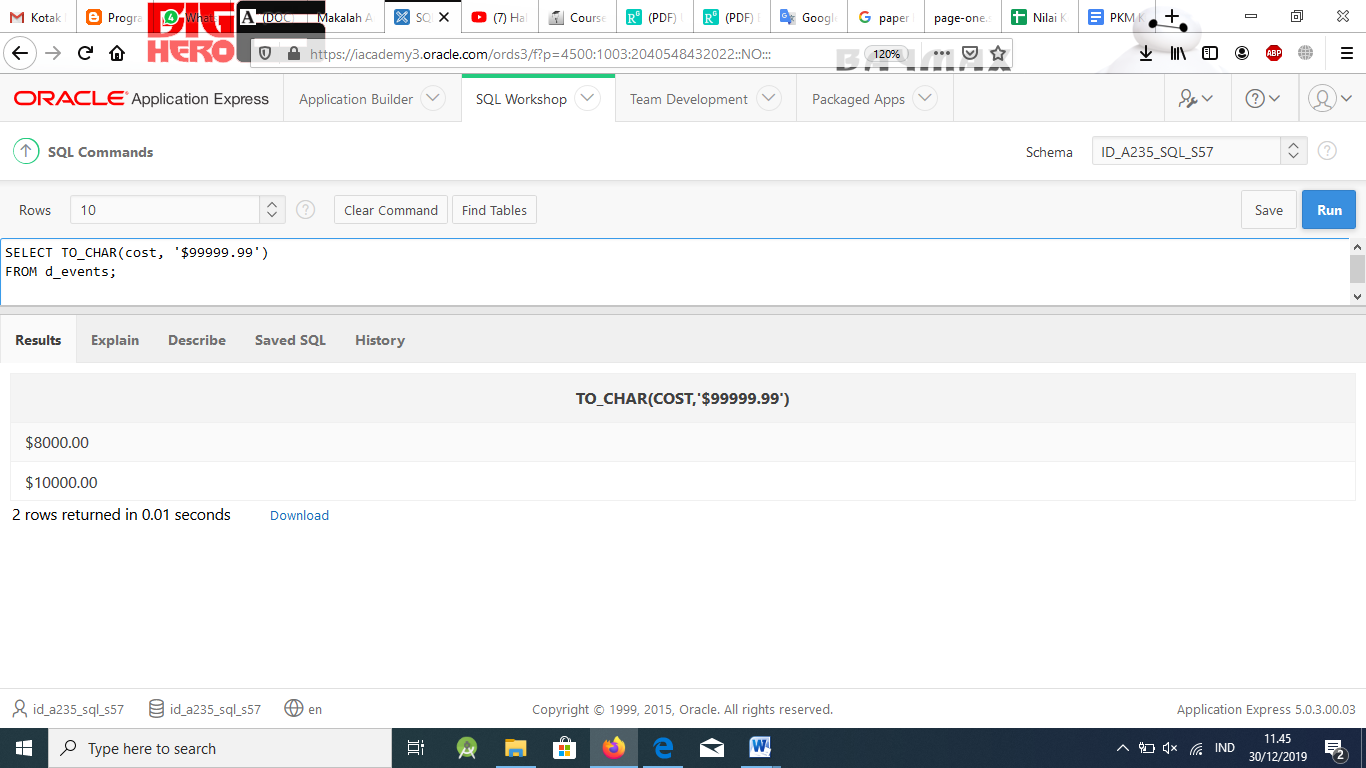
1. Create a query to show the cost of events for DJs on Demand in the format

$0000.00

# Solution:

SELECT TO\_CHAR(cost, '$99999.99')

FROM d\_events;



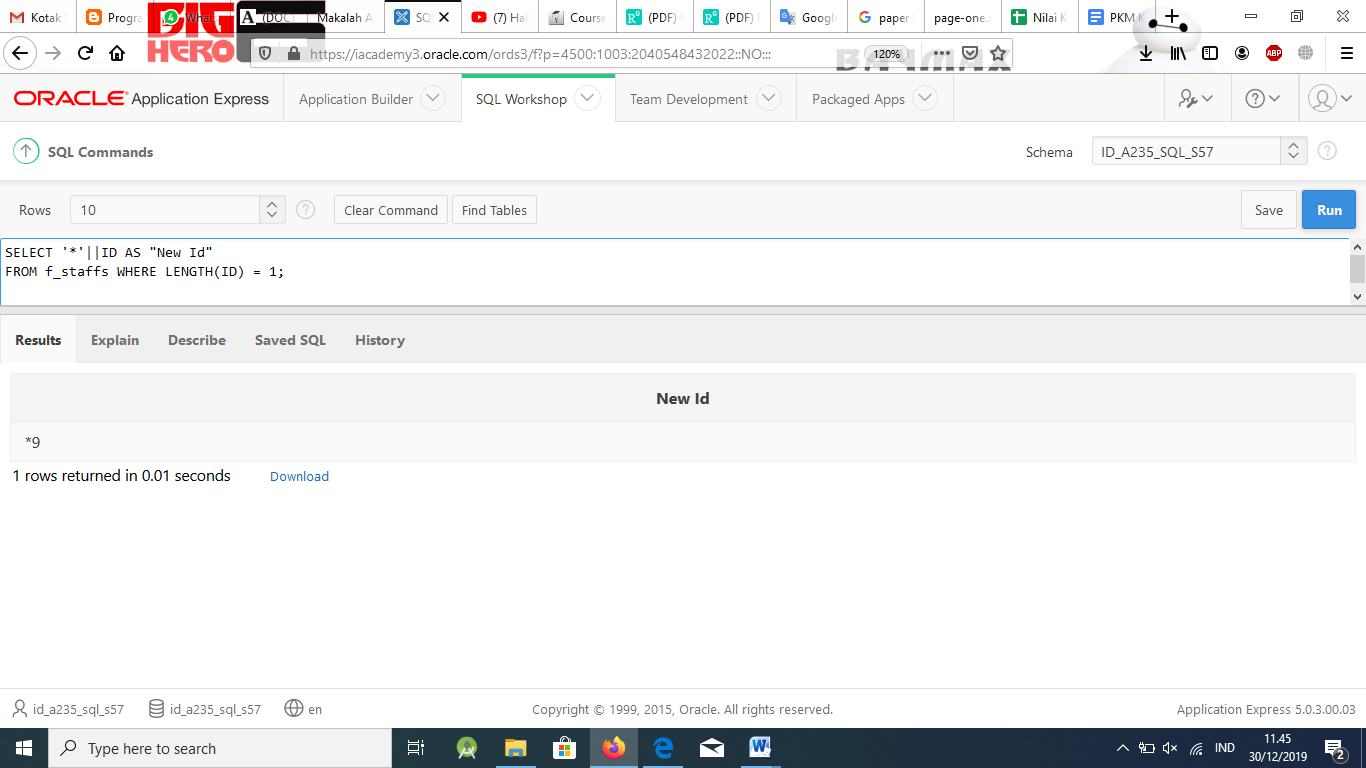
1. For the f\_staffs table in the Global Fast Foods database, change the ID of those staff members whose IDs are only single digits to two digits by adding an asterisk

\* to the front of the number. For example, change ID 9 to ID \*9.

# Solution:

SELECT '\*'||ID AS "New Id"

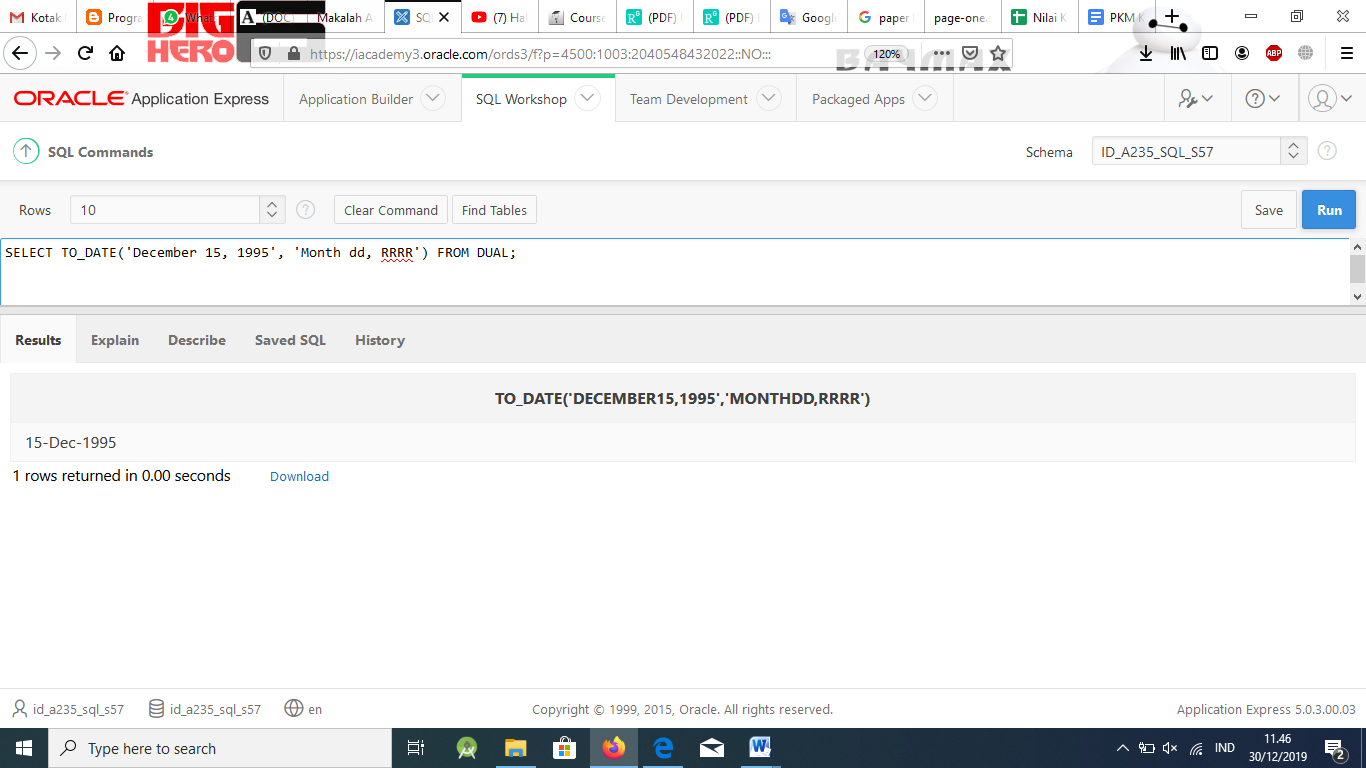
FROM f\_staffs WHERE LENGTH(ID) = 1;



1. As the database administrator, you have been asked to store historical employee records in the current database. The records have termination dates from 1990 to 1999. Write a query using DUAL to show how you could store 15-DEC-95.

# Solution:

SELECT TO\_DATE('December 15, 1995', 'Month dd, RRRR') FROM DUAL;

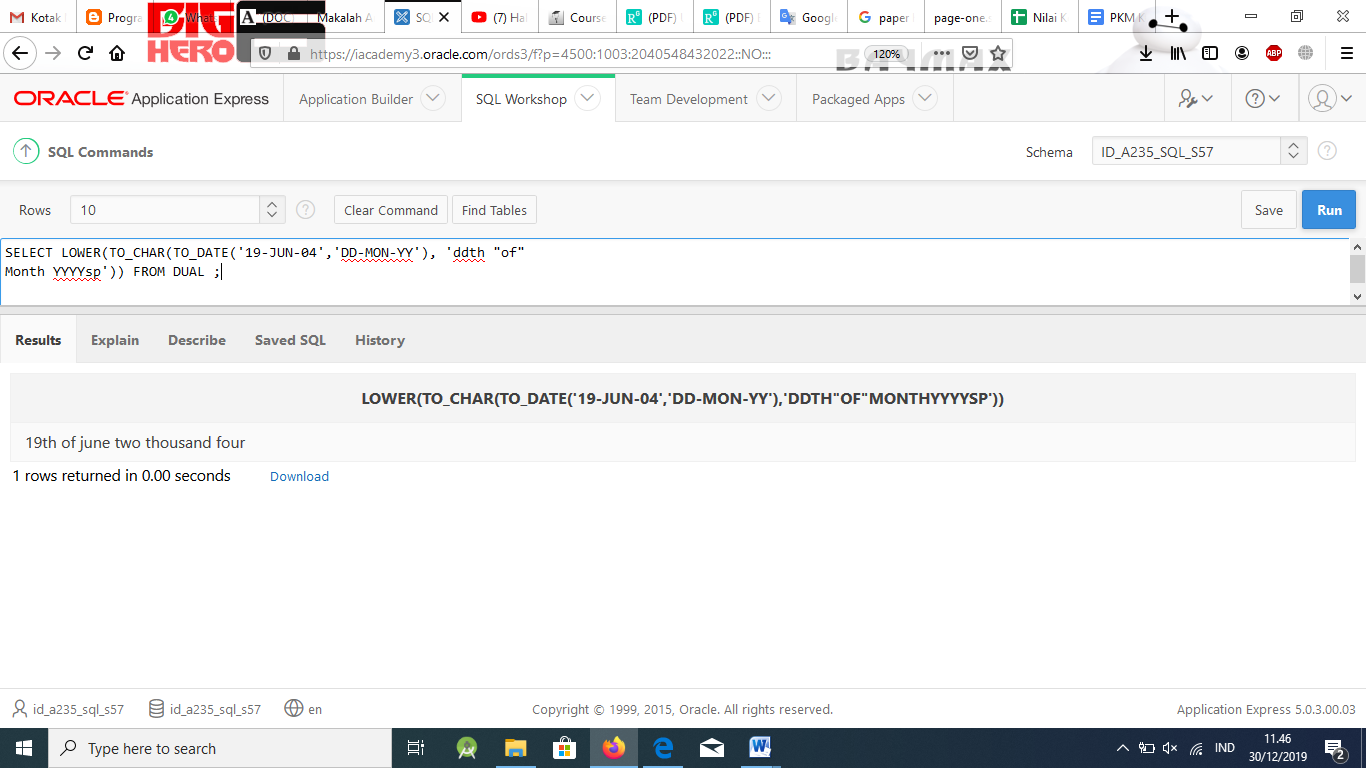


1. Using DUAL, format 19-JUN-04 to appear as: 19th of june two thousand four

# Solution:

SELECT LOWER(TO\_CHAR(TO\_DATE('19-JUN-04','DD-MON-YY'), 'ddth "of"

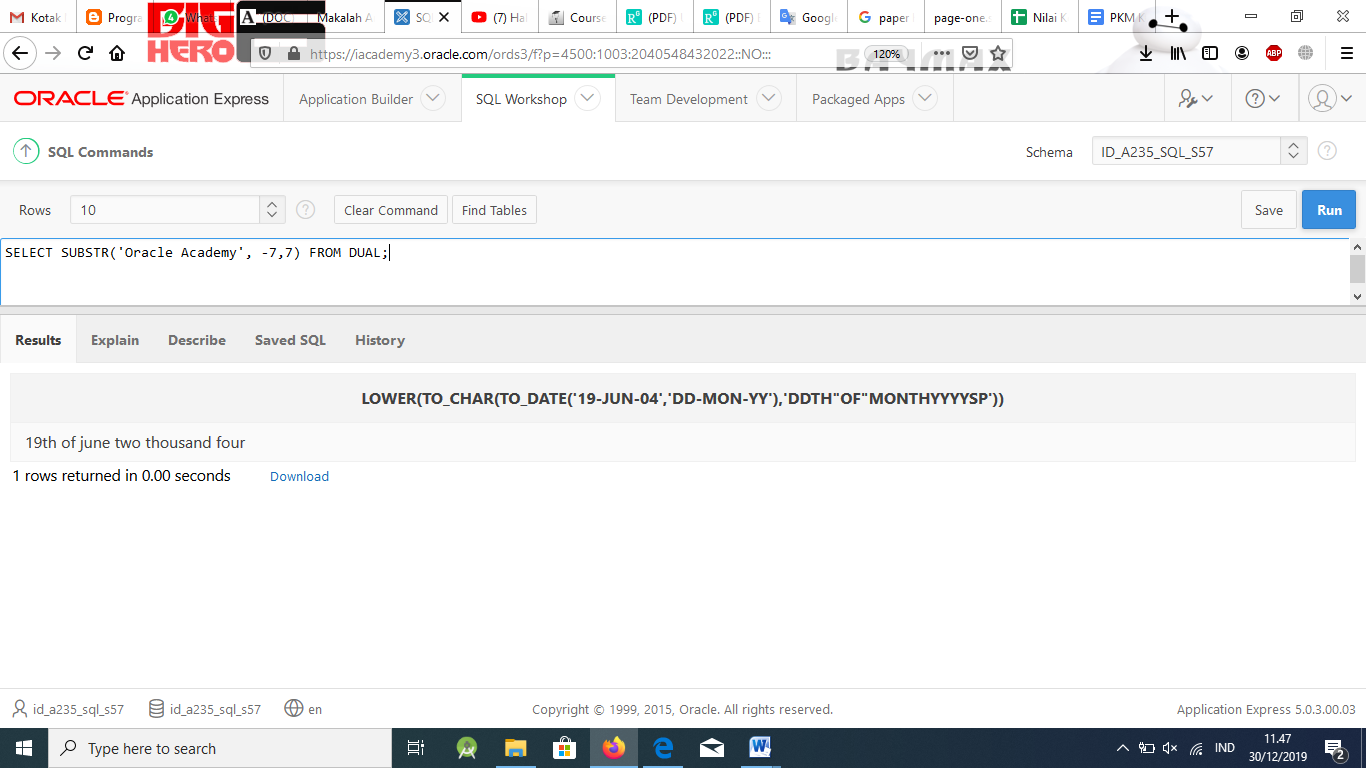
Month YYYYsp')) FROM DUAL ;



1. Create a query that will return only the last word from “Oracle Academy.”

# Solution:

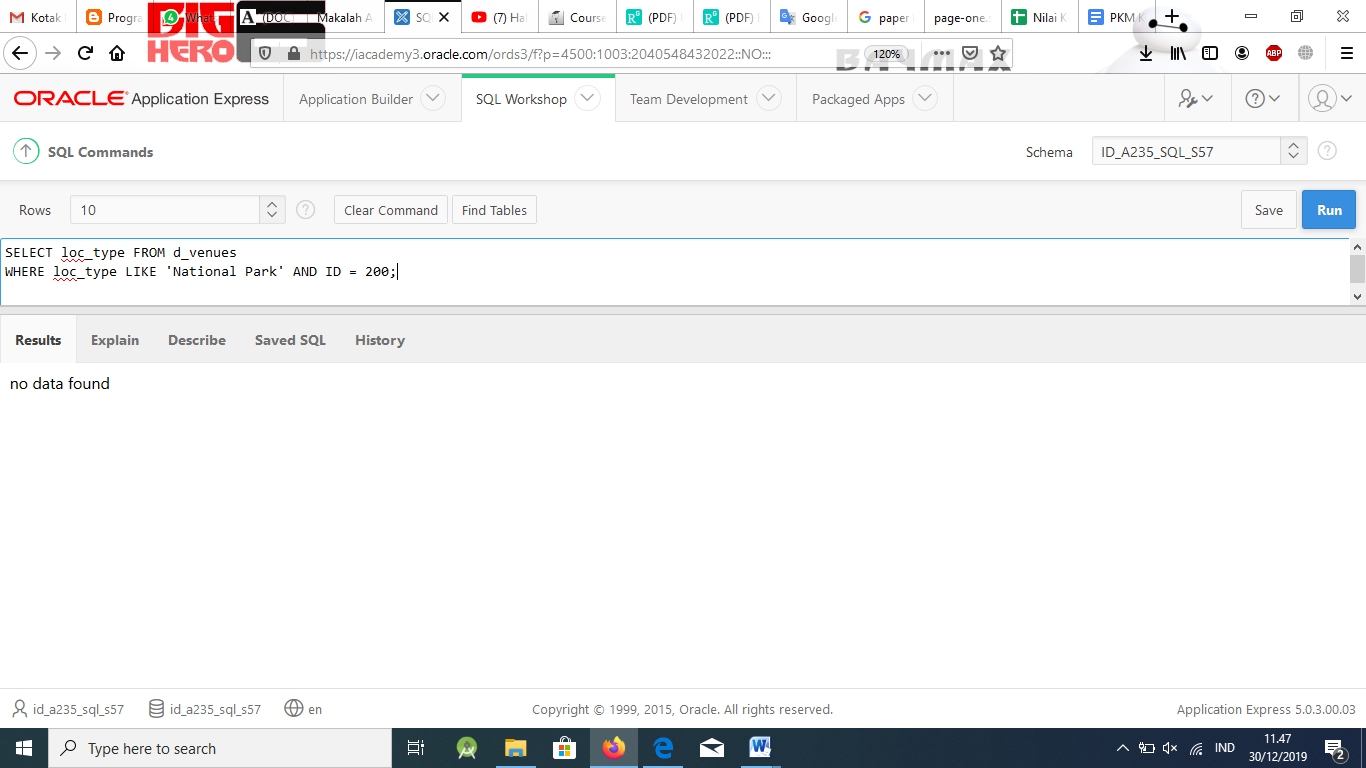
SELECT SUBSTR('Oracle Academy', -7,7) FROM DUAL;



1. Lance and Arnie created the following SQL query but it did not return any results. What is the problem with this query?

SELECT loc\_type FROM d\_venues

WHERE loc\_type LIKE 'National Park' AND ID = 200;



# Solution:

There is nothing wrong with the syntax; there is no matching data in the database.

1. What type of function would you use in each case? D = Date function

N = Number function

C = Conversion/Character functions G = General function

CE = Conditional expression

a. To convert varchar2 to number data

b. To format a date to other than the default format

c. To convert a date such as June 19, 2000 to default format

d. To format a number to appear as currency

e. To substitute a value in a table for null

f. To do an IF-THEN-ELSE statement

g. To find the first not null expression among a list of expressions

h. To replace a section of a string with another string

i. To format a 20th-century date

j. To present output all in uppercase

k. To find the numeric position of a character in a string

l. To find the last day of the month

# Solution:

\_C a. To convert varchar2 to number data

\_C b. To format a date to other than the default format

\_C c. To convert a date such as June 19, 2000 to default format

\_N d. To format a number to appear as currency

\_G e. To substitute a value in a table for null

\_CE\_f. To do an IF-THEN-ELSE statement

\_G g. To find the first not null expression among a list of expressions

\_C h. To replace a section of a string with another string

\_C i. To format a 20th-century date

\_C j. To present output all in uppercase

\_C k. To find the numeric position of a character in a string

\_D l. To find the last day of the month

# Part 3

1. A/An is when the rows of the tables are combined with each other and produce new rows. The number of rows is equivalent to the product of the number of rows in each table.

# Solution:

Cartesian product or cross join.

1. A/An is used when you need to query a table that has a relationship to itself.

# Solution:

SELF JOIN

1. A/An preserves unmatched rows from one or both tables, returning the rows that are matched and unmatched from one or both tables.

# Solution:

OUTER JOIN

1. In an outer join, a plus sign (+) is placed on the side of the join that is

information.

# Solution:

Deficient or is missing

1. A is used when a column in one table does not correspond directly to a column in another table.

# Solution:

Nonequijoin

1. The join condition is always placed in the clause of the SELECT statement.

# Solution:

WHERE

1. A/An is used to preface the column name in order to clarify which table and column are participating in the join.

# Solution:

table alias

1. Table aliases are created in the clause of the SELECT statement.

# Solution:

FROM

1. In a full outer join, a row that does not contain data will/will not appear in the result set if the row satisfies the join condition.

# Solution:

Will

1. Table aliases cannot exceed characters in length.

# Solution:

30

1. Identify the Oracle syntax to signify an outer join .

# Solution:

**+**

1. If a join condition is written: WHERE e.client\_number = c.client\_number, what kind of join would it be if we wanted all the information in the e table even if the c table has missing data?

# Solution:

LEFT OUTER JOIN

1. Joins that are based on hierarchical relationships such as manager and employee are called .

# Solution:

Self joins

1. How many join conditions does it take to join three tables?

# Solution:

2

1. What does the term “proprietary syntax” mean?

# Solution:

Oracle syntax that has been developed by the company

1. What type of join condition is best when two columns with the same name but different datatypes exist in two different tables?

# Solution:

Join the tables with the USING clause.

1. What type of join(s) are based on all columns in two tables that have the same name?

# Solution:

Equijoin and natural join

1. Another name for a cross-join is a .

# Solution:

Cartesian product

1. When specifying a join, you need to identify the what, where, and how. Match up these terms to the SQL syntax words: FROM, WHERE, SELECT.

# Solution:

What - SELECT

Where - FROM How - WHERE

1. For each join condition listed, mark T for those that can include an alias in the syntax:

cross-join

equijoin

nonequijoin

natural join

full outer join

left outer join

USING clause

# Solution:

\_T cross-join

\_T equijoin

\_T nonequijoin

natural join

\_T\_ full outer join

\_T\_ left outer join

USING clause

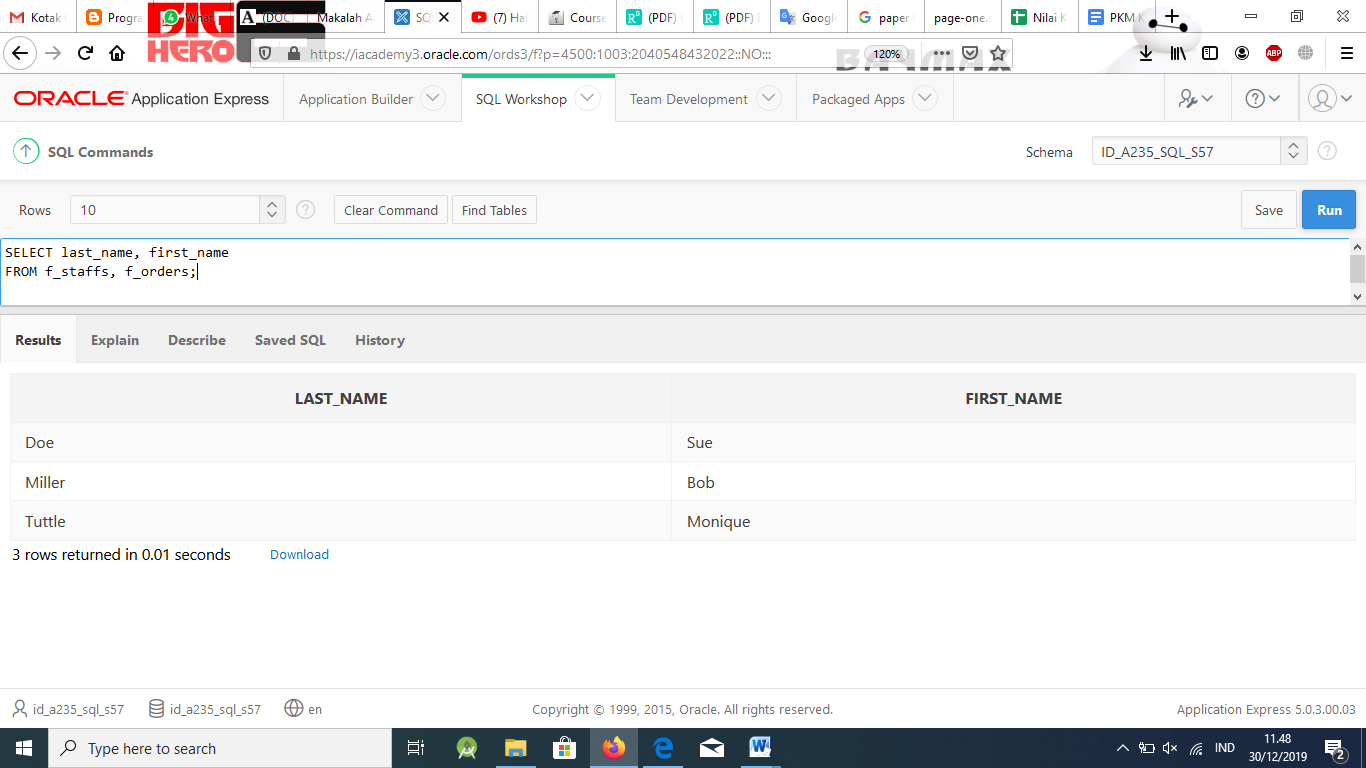
1. Which two types of join conditions cannot use an alias?

# Solution:

* + Joins with the USING clause
  + Natural join when the columns have the same name

1. The query below is an example of what kind of join? SELECT last\_name, first\_name

FROM f\_staffs, f\_orders;



|  |  |  |
| --- | --- | --- |
| LAST\_NAME | SID | MID |
| Abernathy | 349 | Null |
| Avery | 342 | Null |
| Barnaby | 1420 | 349 |
| Chang | 555 | 349 |
| Dixon | 554 | 349 |
| Evans | 873 | 342 |
| Franco | 1933 | 342 |
| Gallaway | 943 | 349 |

STUDENTS

|  |  |
| --- | --- |
| COURSES | CID |
| Oracle\_Internet\_Academy\_DM | 101 |
| Oracle\_Internet\_Academy\_Java | 102 |
| Oracle\_Internet\_Academy\_Java | 103 |

COURSES

|  |  |
| --- | --- |
| SID | CID |
| 1420 | 101 |
| 555 | 101 |
| 554 | 101 |
| 873 | 102 |
| 342 | 103 |
| 349 | 103 |
| 943 | 103 |
| 1933 | 103 |

ENROLLED

# Solution:

Cross join or Cartesian product

1. Use the three tables shown above to answer the following questions:
2. What kind of join would you use to join all three tables?

Write the syntax that would produce the desired result.

# Solution:

EQUIJOIN

SELECT s.name, c.name

FROM students s, course c, enrolled e WHERE (s.sid = e.sid) AND (e.cid = c.cid);

1. Name two tables that could be used to retrieve data from a natural join. Write the syntax that would produce the desired result.

# Solution:

FROM students NATURAL JOIN enrolled or FROM courses NATURAL JOIN enrolled;

1. What kind of join would you use to return only those students who have mentors?

Write the syntax that would produce the desired result.

# Solution:

SELF JOIN

SELECT s.name, m.name FROM students s, students m WHERE s.mid=m.sid;

1. What kind of join would you use to return all students whether they have a mentor or not.

Write the syntax that would produce the desired result.

# Solution:

SELECT s.name, m.name FROM students s, students m WHERE s.mid=m.sid(+); SELECT s.name, m.name

FROM students s LEFT OUTER JOIN students m ON (s.mid=m.sid);

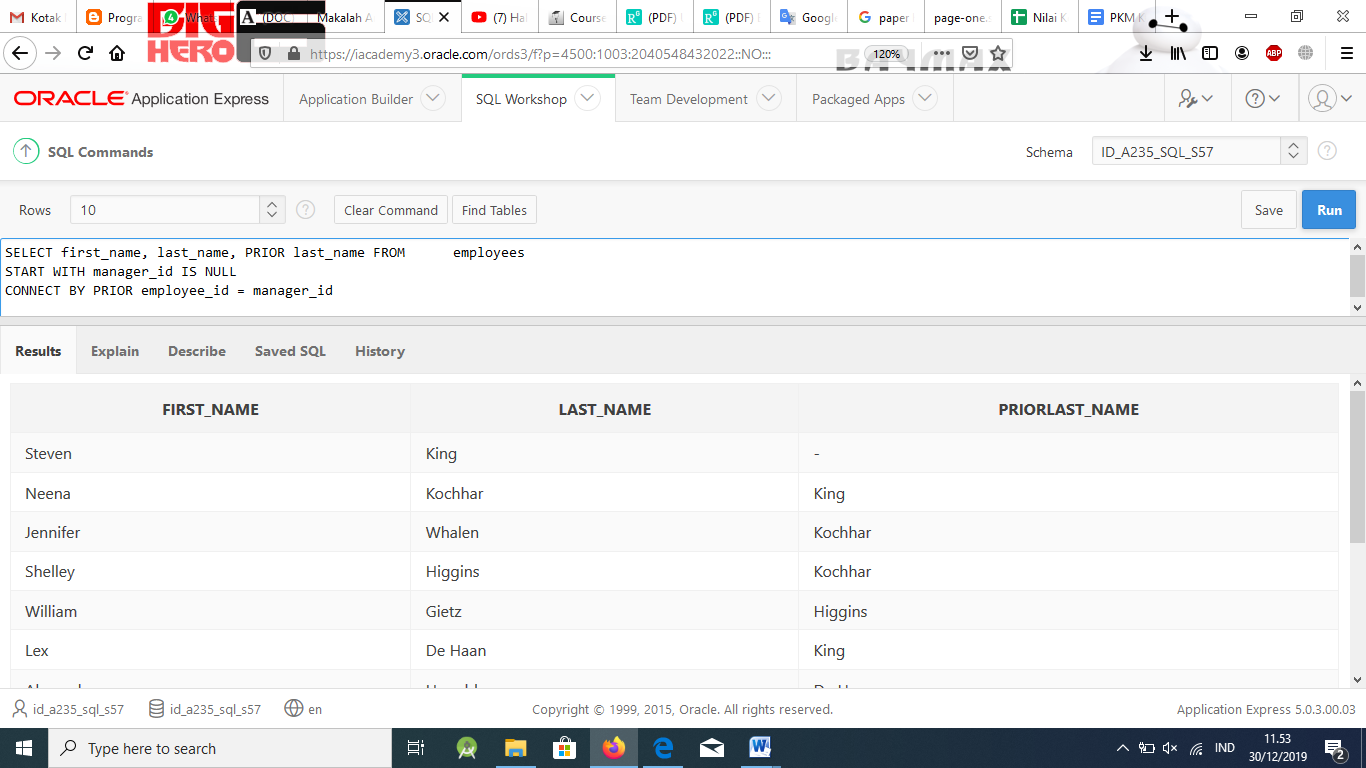
1. Write a hierarchical query to display the organization chart of the employees table. Ensure the query starts with the employee who has no manager. Make the query return the employee’s first and last names along with the last name of his manager.

# Solution:

SELECT first\_name, last\_name, PRIOR last\_name FROM employees

START WITH manager\_id IS NULL

CONNECT BY PRIOR employee\_id = manager\_id

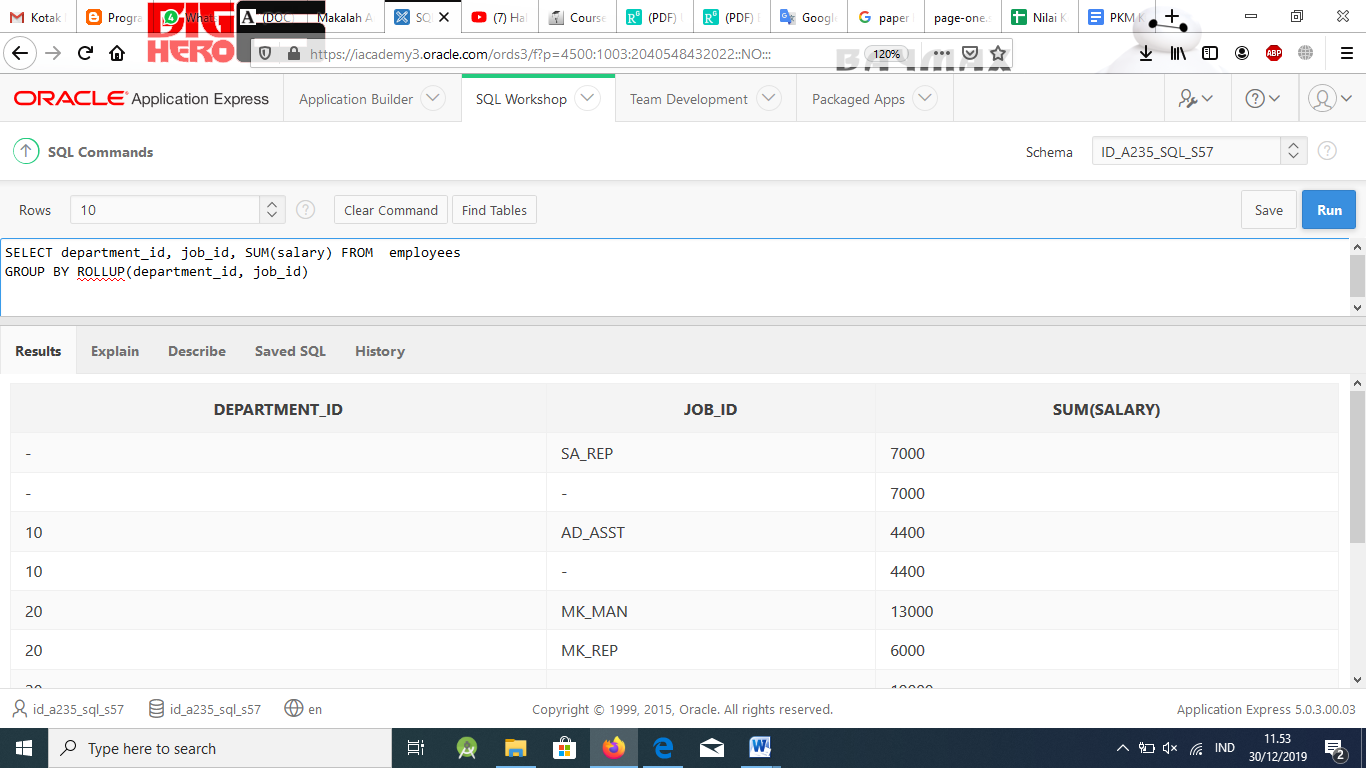


1. Write a query that displays the department\_id, job\_id, and the total salary for each job\_id within a department. Make the query return subtotals of salary per department\_id and a grand total of all salaries across the company. Use the EMPLOYEES table for this query.

# Solution:

SELECT department\_id, job\_id, SUM(salary) FROM employees

GROUP BY ROLLUP(department\_id, job\_id)



1. Write a query that will combine the following columns: employee first and last\_name, department\_id, job\_id, and salary from the employees table, the employee\_id, start\_date, end\_date, and job\_id from job\_history and the department\_name and location\_id from the departments table. The queries should return one single output.

# Solution:

SELECT first\_name, last\_name, department\_id, job\_id,

salary,

to\_number(null) employee\_id, to\_date(null) start\_date, to\_date(null) end\_date, to\_char(null) department\_name, to\_number(null) location\_id

FROM employees UNION

SELECT null, null,

null, job\_id, null,

employee\_id, start\_date, end\_date, null,

null

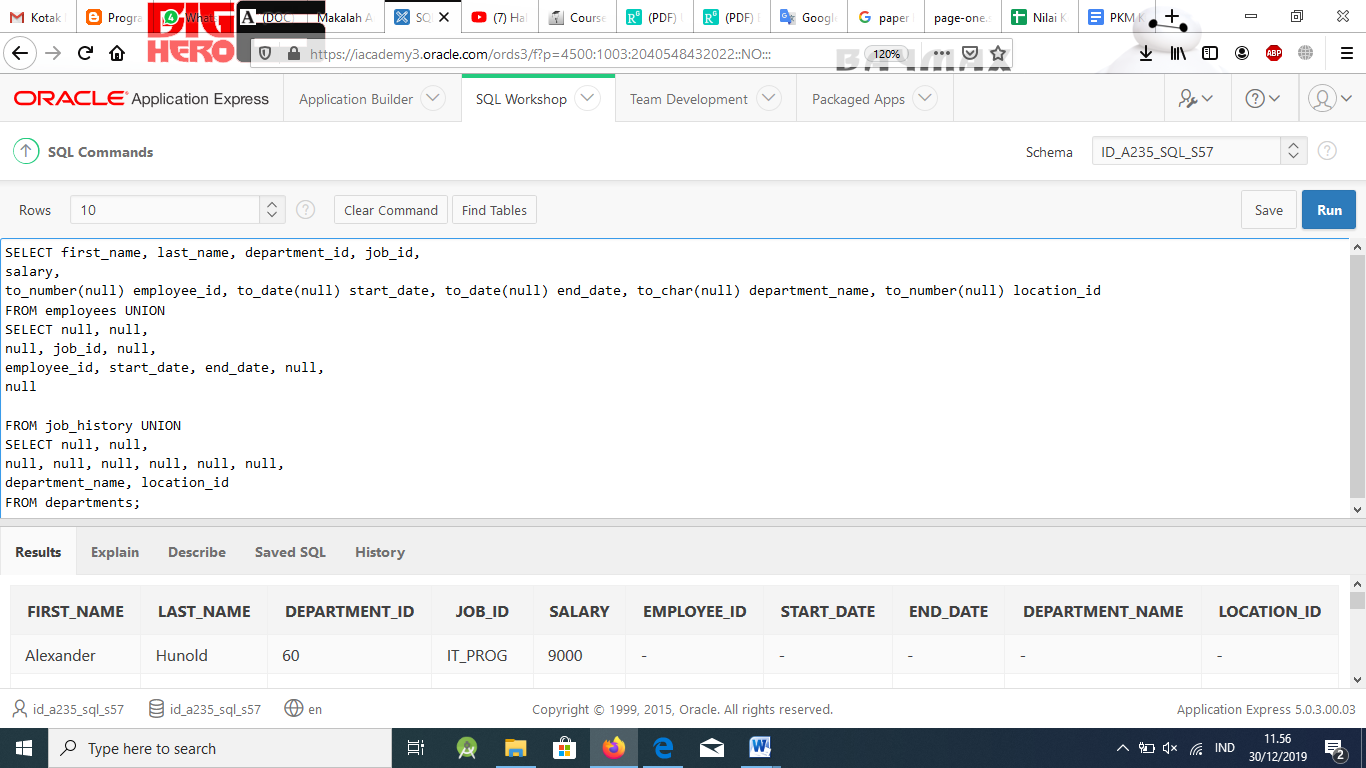
FROM job\_history UNION

SELECT null, null,

null, null, null, null, null, null,

department\_name, location\_id

FROM departments;



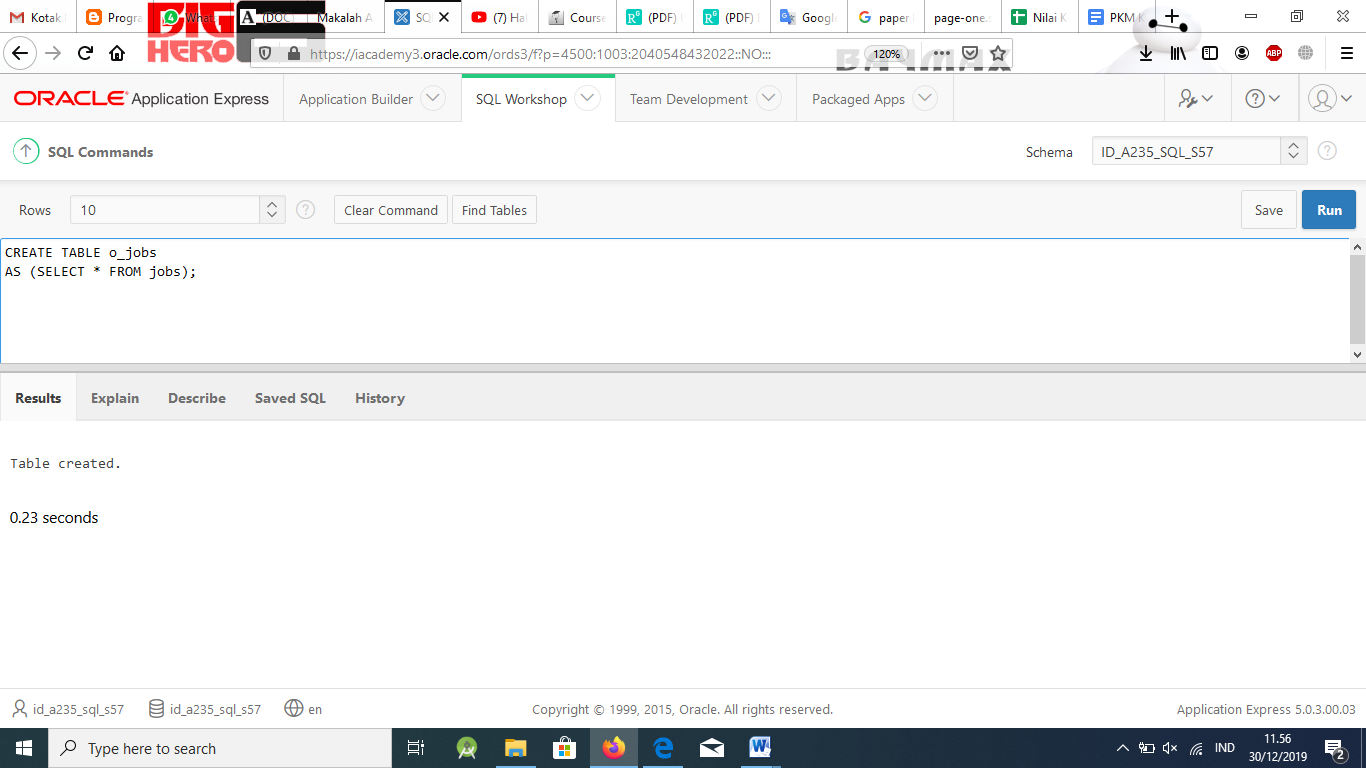
# Part 4

1. Create copies of the following Oracle database tables and name them as specified:
   * employees copied as o\_employees
   * departments copied as o\_departments
   * jobs copied as o\_jobs

# Solution:

Create the three o\_tables, jobs, employees and departments using the syntax: CREATE TABLE o\_jobs

AS (SELECT \* FROM jobs);



1. As the DBA for Oracle, you have been asked to update the database with new information.

Last month, O created a new department in Seattle called Human Resources. The department was assigned ID 210. Employees in this department have the job title Human Resources Manager, and job ID HR\_MAN. The salary for all new employees in this department ranges from a minimum of $4500 to a maximum of

$5500.

Add the Human Resources job to the o\_jobs table.

|  |  |  |
| --- | --- | --- |
| EMPLOYEE\_ID | FIRST\_NAME | LAST\_NAME |
| 210 | Ramon | Sanchez |
| 211 | Tai | Sugita |
| 212 | Alina | Arcos |

# Solution:

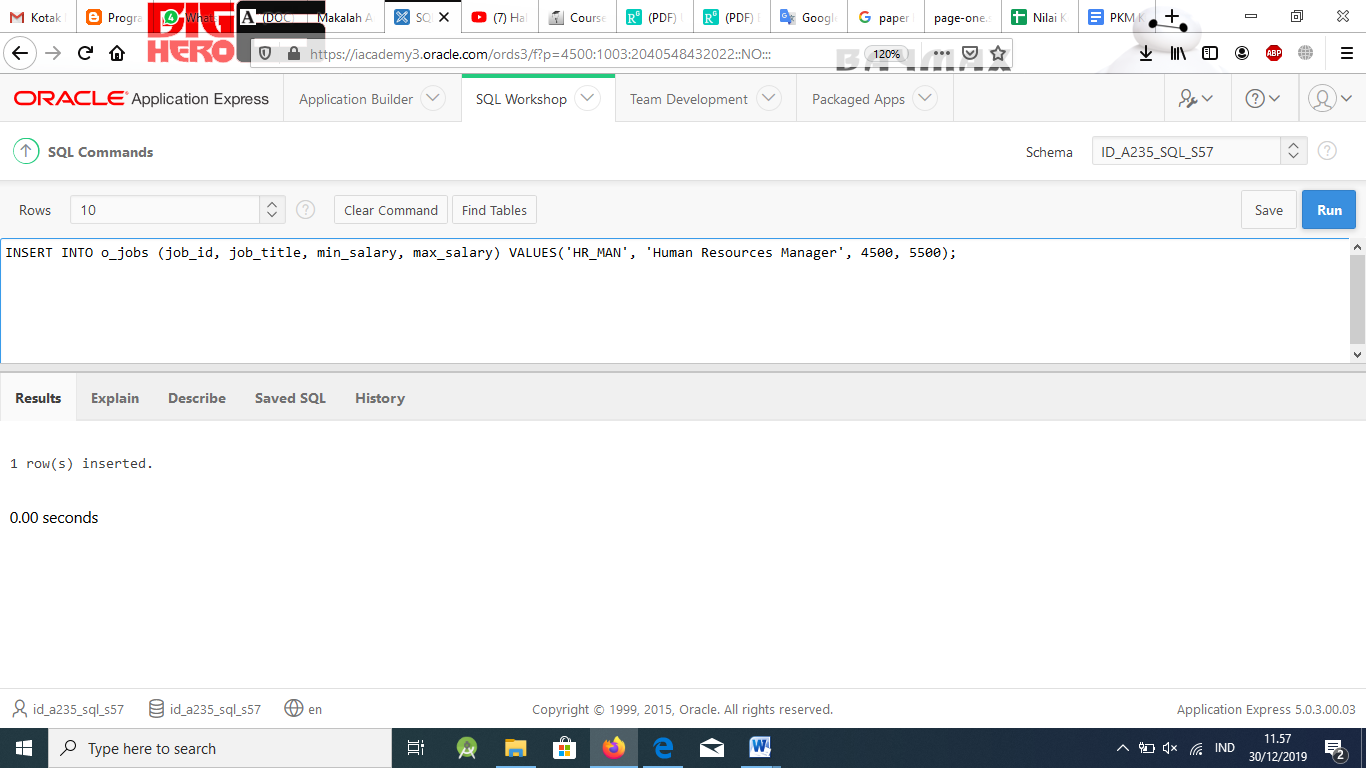
Add the Human Resources job to the jobs table:

INSERT INTO o\_jobs (job\_id, job\_title, min\_salary, max\_salary) VALUES('HR\_MAN', 'Human Resources Manager', 4500, 5500);

Don’t accept this assignment until the correct output is produced. Students should have added three employees to the o\_employees table, added the 'HR\_MAN' job to the o\_jobs table, and added the ‘Human Resources’ department

to the o\_departments table. Verify that students were successful in producing the correct output (ask them to print or show you their output)

Changes to these tables will be done in Lesson 3.



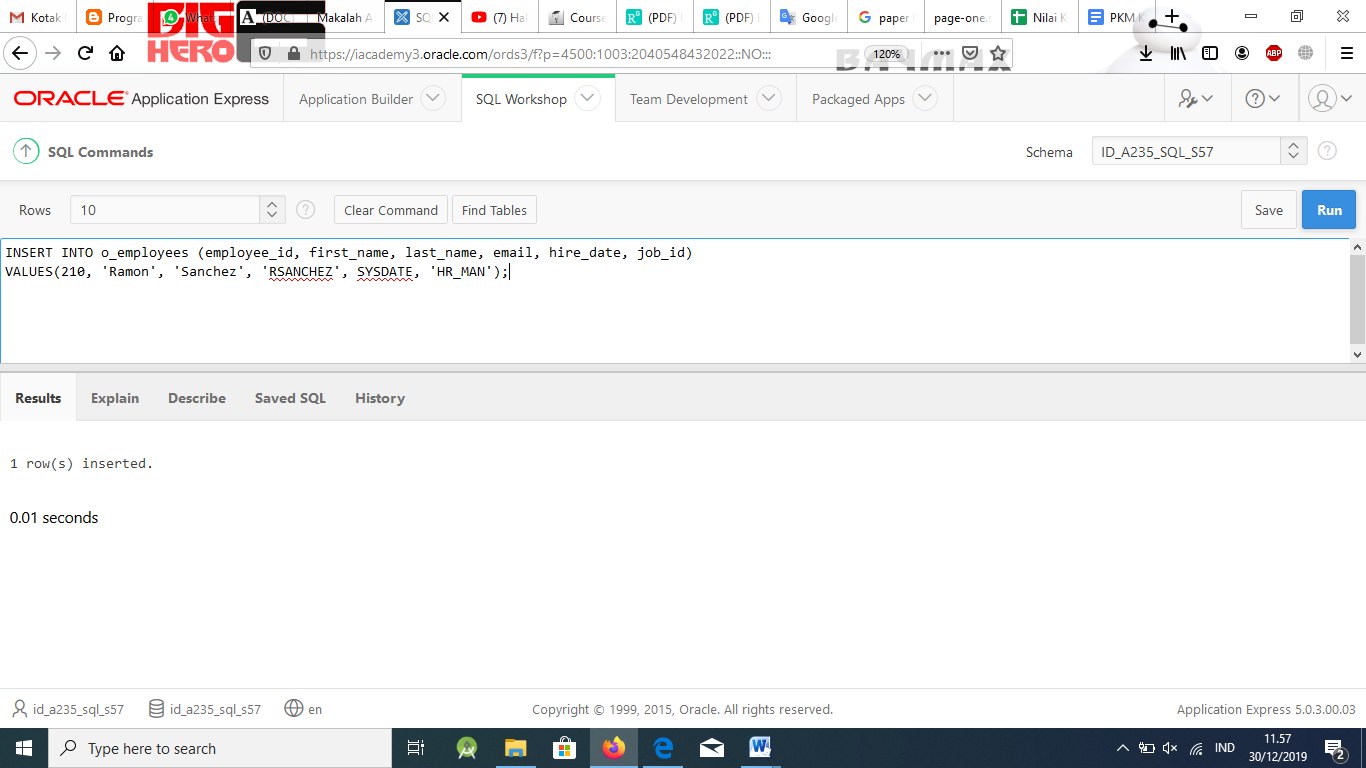
1. Three new employees hired for this department are shown in the table. Add them to the o\_employees table.

Each employee will need an email address created from the first letter of the employee’s first name combined with the employee’s last name (Bob Smith would be BSMITH). Use the current date as the hire date.

# Solution:

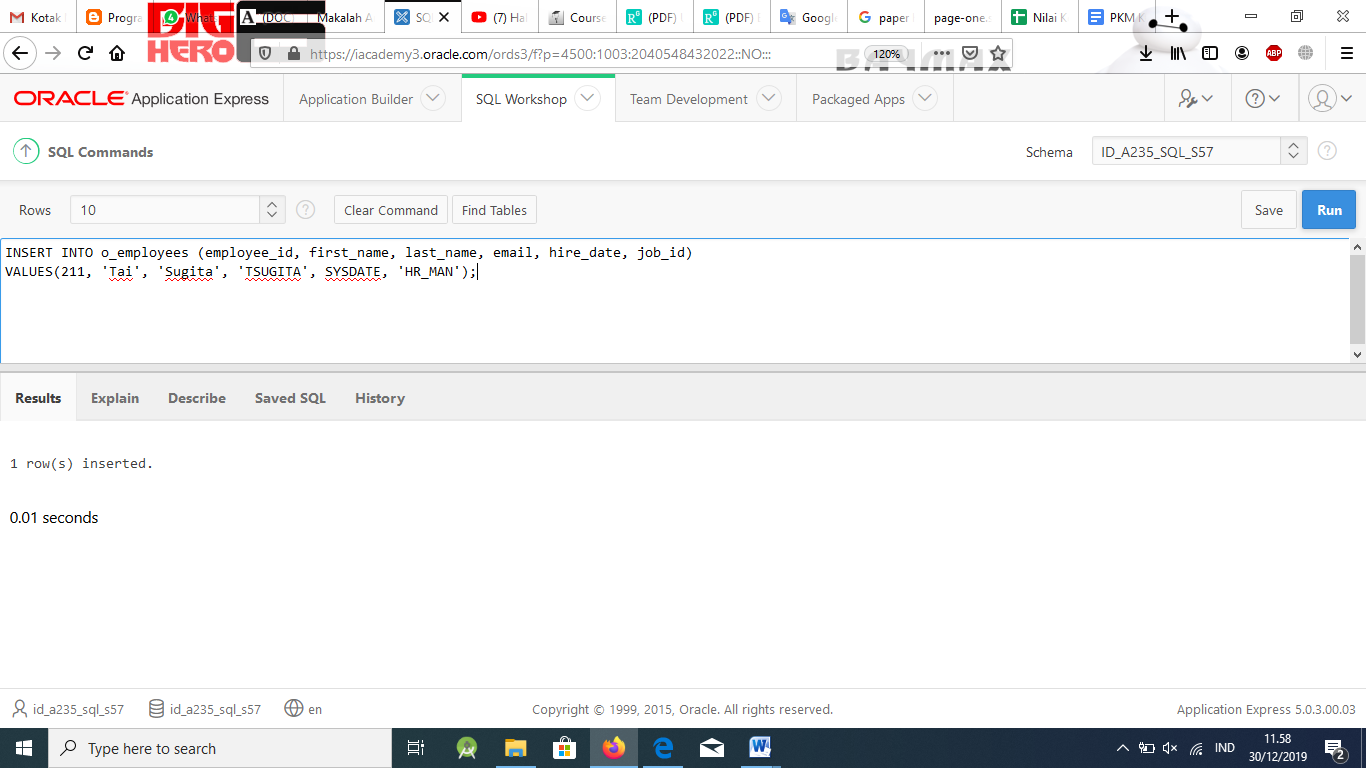
INSERT INTO o\_employees (employee\_id, first\_name, last\_name, email, hire\_date, job\_id)

VALUES(210, 'Ramon', 'Sanchez', 'RSANCHEZ', SYSDATE, 'HR\_MAN');



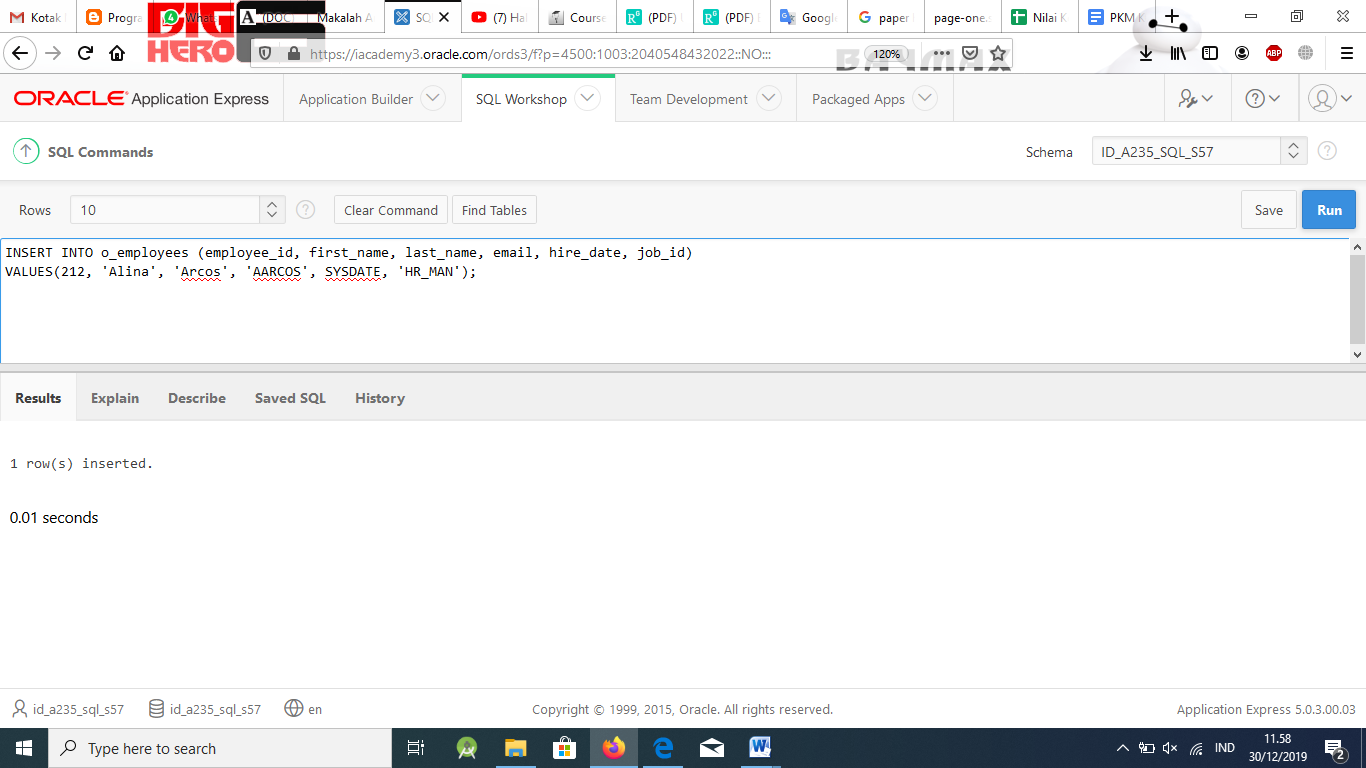
INSERT INTO o\_employees (employee\_id, first\_name, last\_name, email, hire\_date, job\_id)

VALUES(211, 'Tai', 'Sugita', 'TSUGITA', SYSDATE, 'HR\_MAN');



INSERT INTO o\_employees (employee\_id, first\_name, last\_name, email, hire\_date, job\_id)

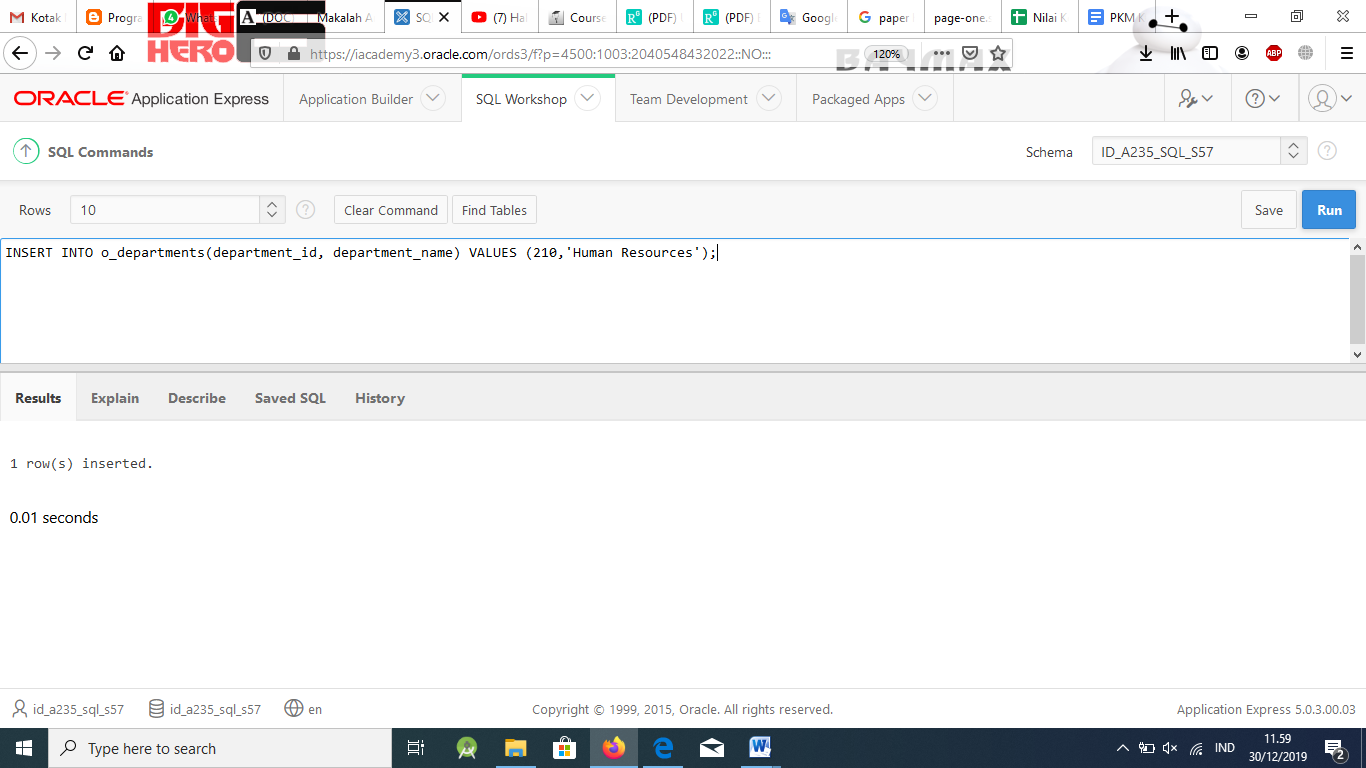
VALUES(212, 'Alina', 'Arcos', 'AARCOS', SYSDATE, 'HR\_MAN');



1. Add Human Resources to the o\_departments table.

# Solution:

INSERT INTO o\_departments(department\_id, department\_name) VALUES (210,'Human Resources');



Don’t accept this assignment until the correct output is produced. Students should have added three employees to the o\_employees table, added the 'HR\_MAN' job to the o\_jobs table, and added the ‘Human Resources’ department to the o\_departments table. Verify that students were successful in producing the correct output (ask them to print or show you their output).

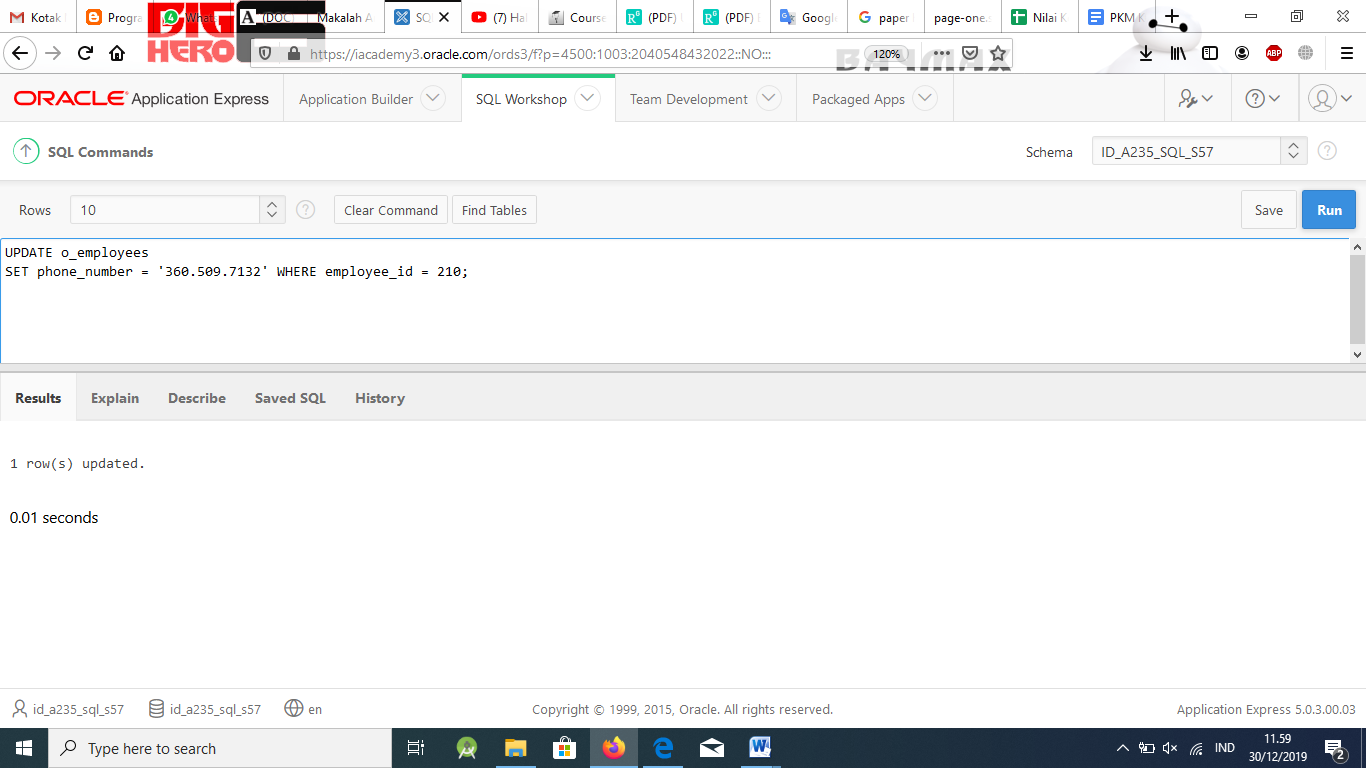
Changes to these tables will be done in Lesson 3.

1. Several changes need to be made to the o\_employees, o\_departments, and o\_jobs tables.
   * Ramon Sanchez has a new phone number: 360-509-7132.
   * The location for the Seattle Human Resources department is 1700.
   * Ramon Sanchez, Tai Sugita, and Alina Arcos need their department ID updated to 210 if it has not been done as yet.
   * Ramon’s salary is $5000; Tai earns $5100.
   * Delete Alina Arcos, who has decided to return to college to finish her sociology degree.

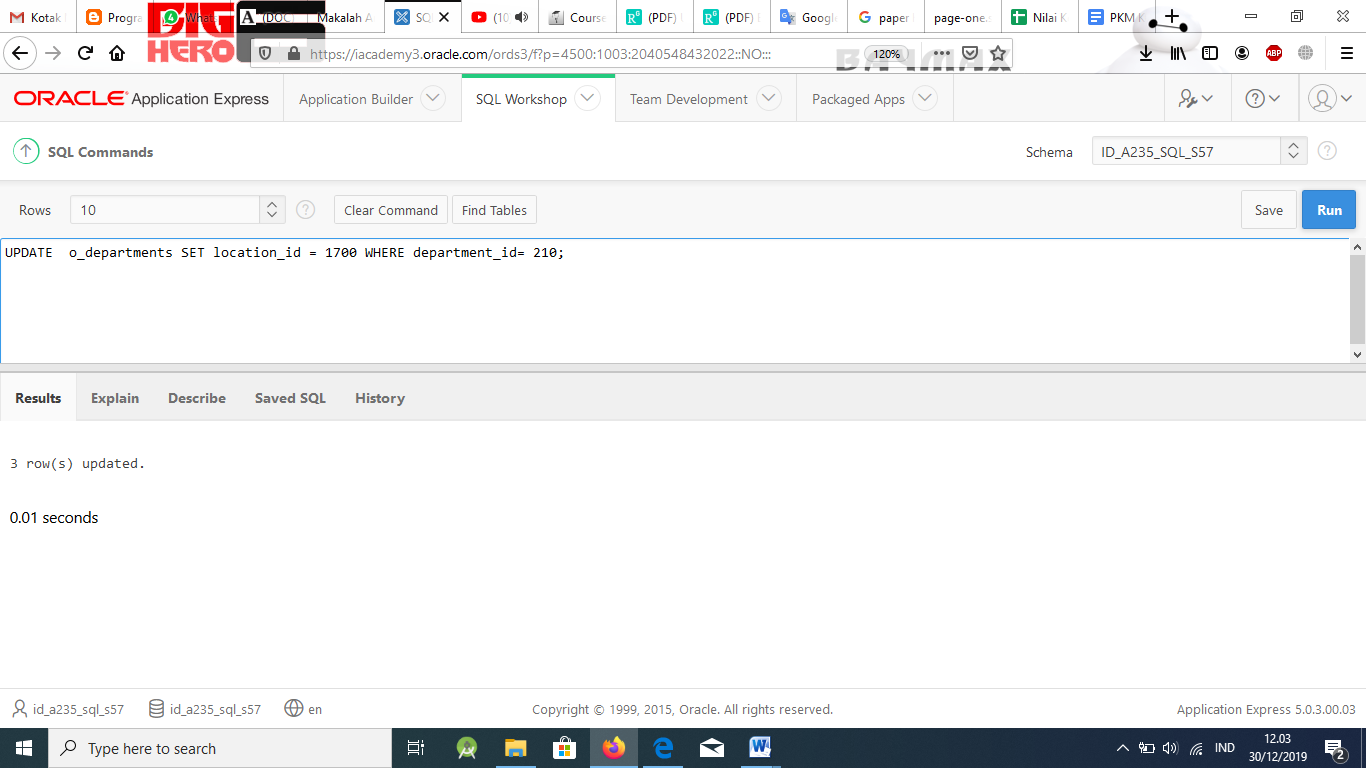
# Solution:

UPDATE o\_employees

SET phone\_number = '360.509.7132' WHERE employee\_id = 210;



UPDATE o\_departments SET location\_id = 1700 WHERE department\_id= 210;



UPDATE o\_employees SET department\_id = 210

WHERE employee\_id in (210,211,212);

UPDATE o\_employees SET salary = 5000

WHERE employee\_id = 210;

UPDATE o\_employees SET salary = 5100

WHERE employee\_id = 211;

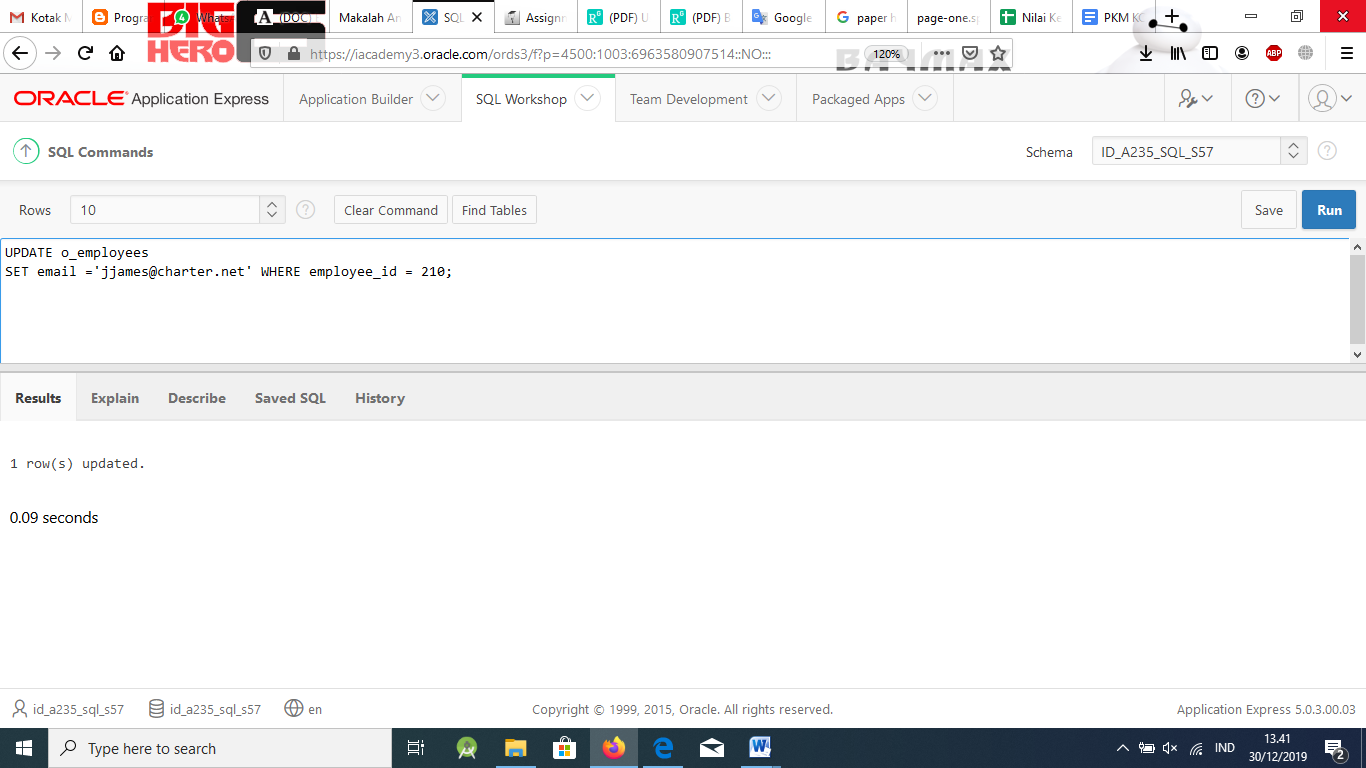
DELETE FROM o\_employees WHERE employee\_id = 212;

Additional information: Verify that the new changes have been made to the o\_tables as specified. Explain that DML statements should be made to primary- key values such as employee ID instead of last name or first name just in case two employees have the same name. Two employees won’t have the same ID.

For example, in a large company, two employees may be named Jennifer James, but each employee will have a different employee identification number. When changes are made to Jennifer James’s email address, it is best to use the query shown on the left in the following example rather than the query shown on the right.

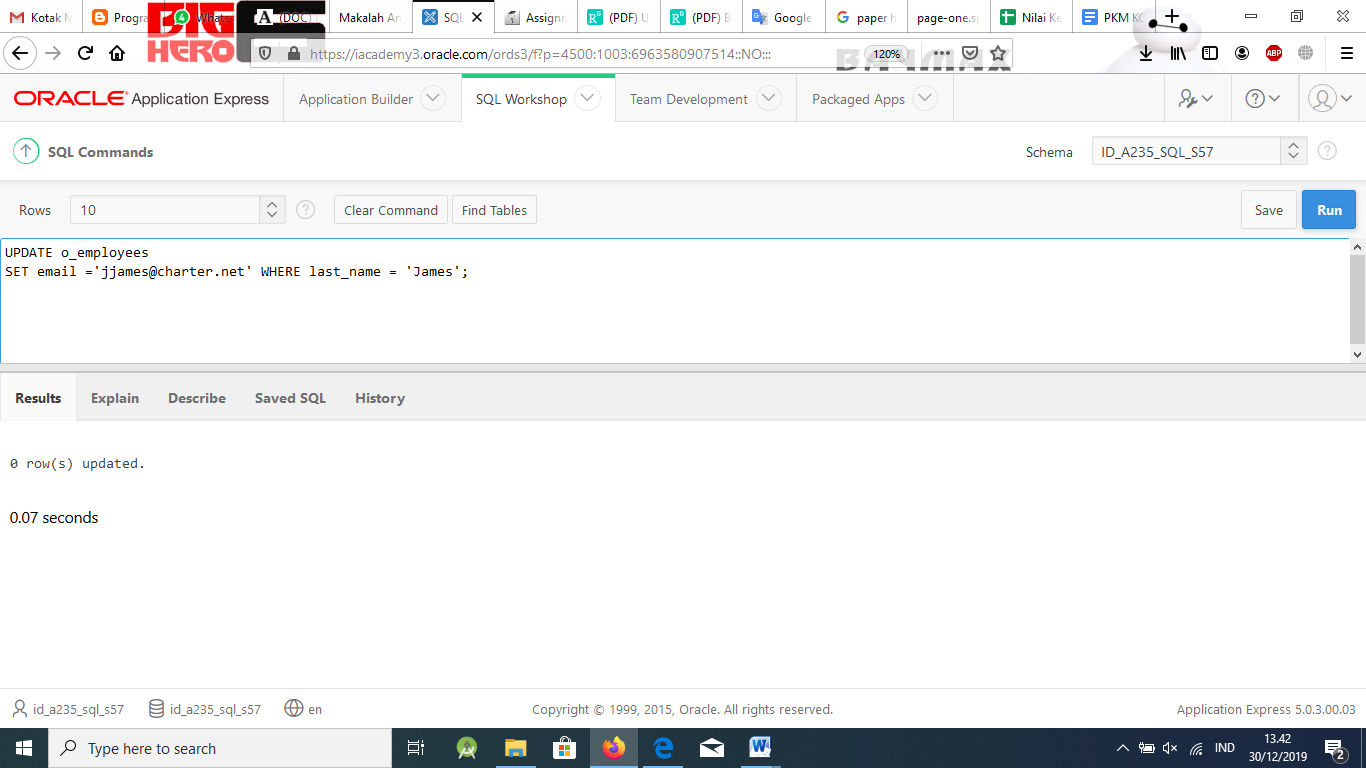
UPDATE o\_employees

SET email ='jjames@charter.net' WHERE employee\_id = 210;



UPDATE o\_employees

SET email ='jjames@charter.net' WHERE last\_name = 'James';



# Part 5

1. What purpose do constraints serve in a database?

# Solution:

Constraints enforce rules and prevent deletion of tables when there are dependencies.

1. Why would you choose to create a CHECK constraint? Give an example.

# Solution:

CHECK constraints specify a condition that must be True. Example: A negative salary amount cannot be entered in a table.

1. How can you find out what constraints have been added to a column?

# Solution:

Query the data dictionary USER\_CONSTRAINTS.

1. When can a constraint be defined for a table?

# Solution:

When a table is created or added after the table is created (depending on data already in the table).

1. What’s the advantage of naming your own constraints?

# Solution:

The SYSn name does not explicitly name which column the constraint refers to.

1. Why is referential integrity a concern for the DBA?

# Solution:

Referential integrity prevents additions, deletions, or modifications of information that violates business rules.

1. Why would you choose to define a constraint at the column level rather than in one place at the table level?

# Solution:

It is really a personal preference, but NON NULL constraints must be created at the column level and composite keys at the table level.

1. Why would a UNIQUE constraint be created?

# Solution:

To prevent duplicate information in a column that is not a primary-key column.

1. Can more than one column be not null when a UNIQUE constraint is used? Why or why not?

# Solution:

Yes, because nulls are not equal to anything – just can’t have identical values in non-null columns.

1. Give an example of a composite unique key.

# Solution:

last name and email address

1. Only primary key(s) can be created for each table.

# Solution:

one

1. What does a primary key enforce?

# Solution:

nonduplicate values in a column

1. What is another name for a FOREIGN KEY constraint?

# Solution:

referential integrity constraint

1. What does ON DELETE CASCADE accomplish?

# Solution:

ODC deletes the corresponding foreign key referenced row when parent rows are deleted.

1. What is the function of CASCADE in the ALTER CONSTRAINT syntax?

# Solution:

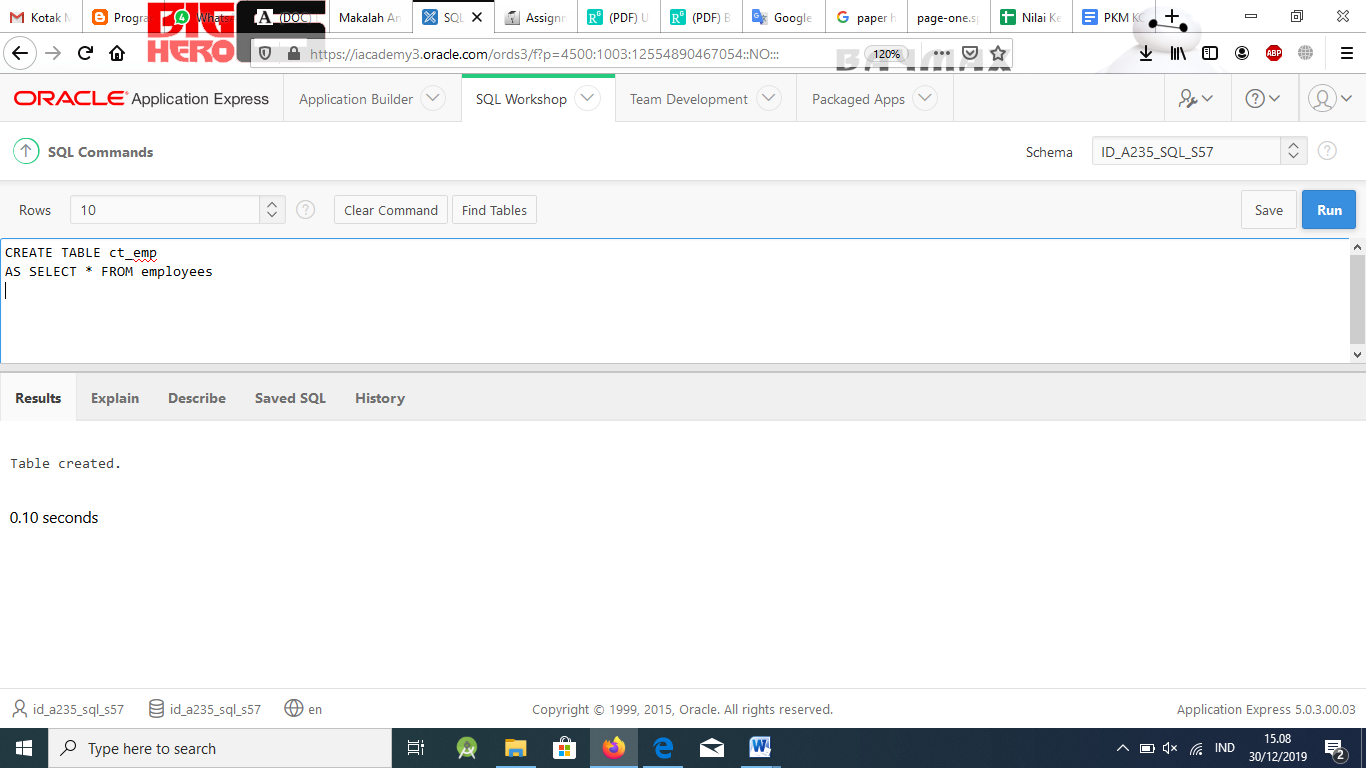
Release the foreign key constraint on the parent table.

1. Create a copy of the employees table and call it your\_initials\_emp. Add a column to the new table called department\_name, and make this column the same dataype and length as the department\_name column in the departments table. Write a correlated update statement to update all the department\_name columns in your new table to be the same as the department\_name column values in the departments table itself.

# Solution:

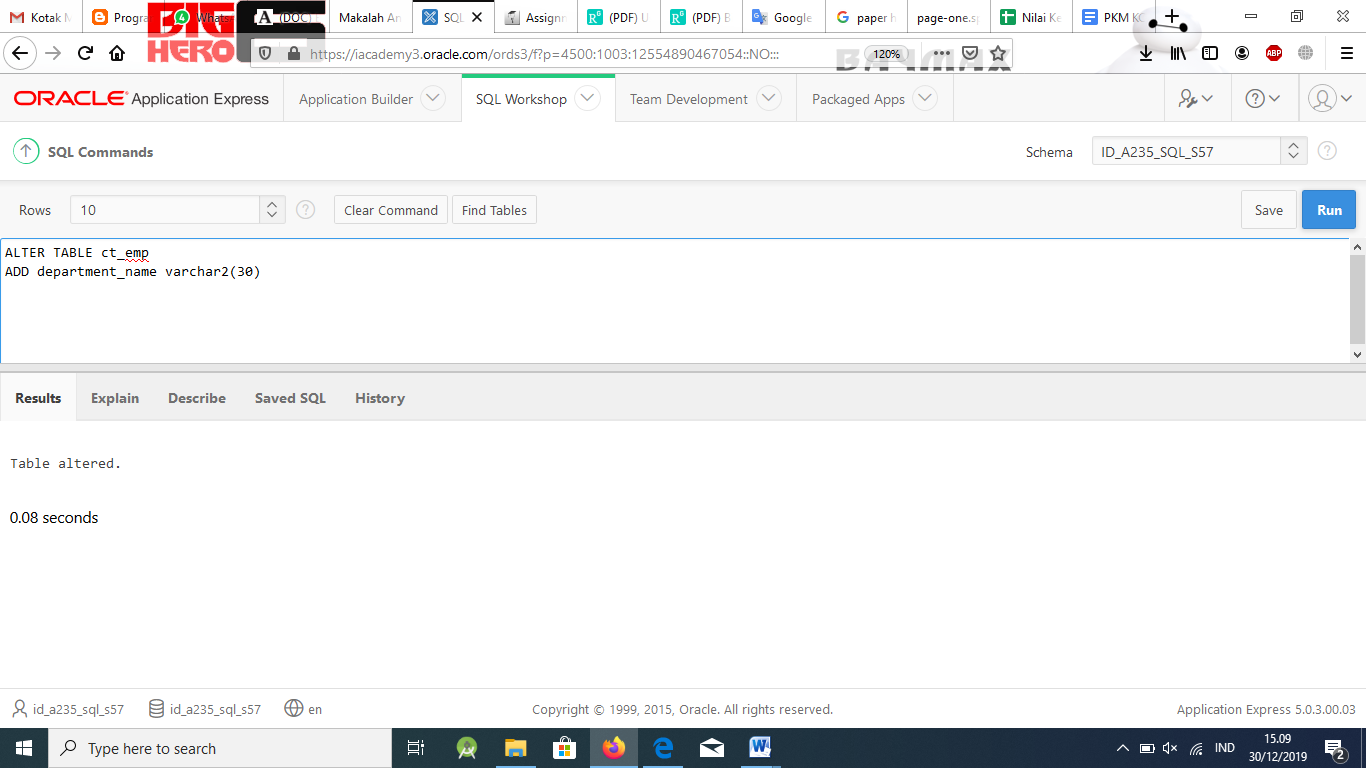
CREATE TABLE ct\_emp

AS SELECT \* FROM employees – answers will vary here in the table name part.



ALTER TABLE ct\_emp

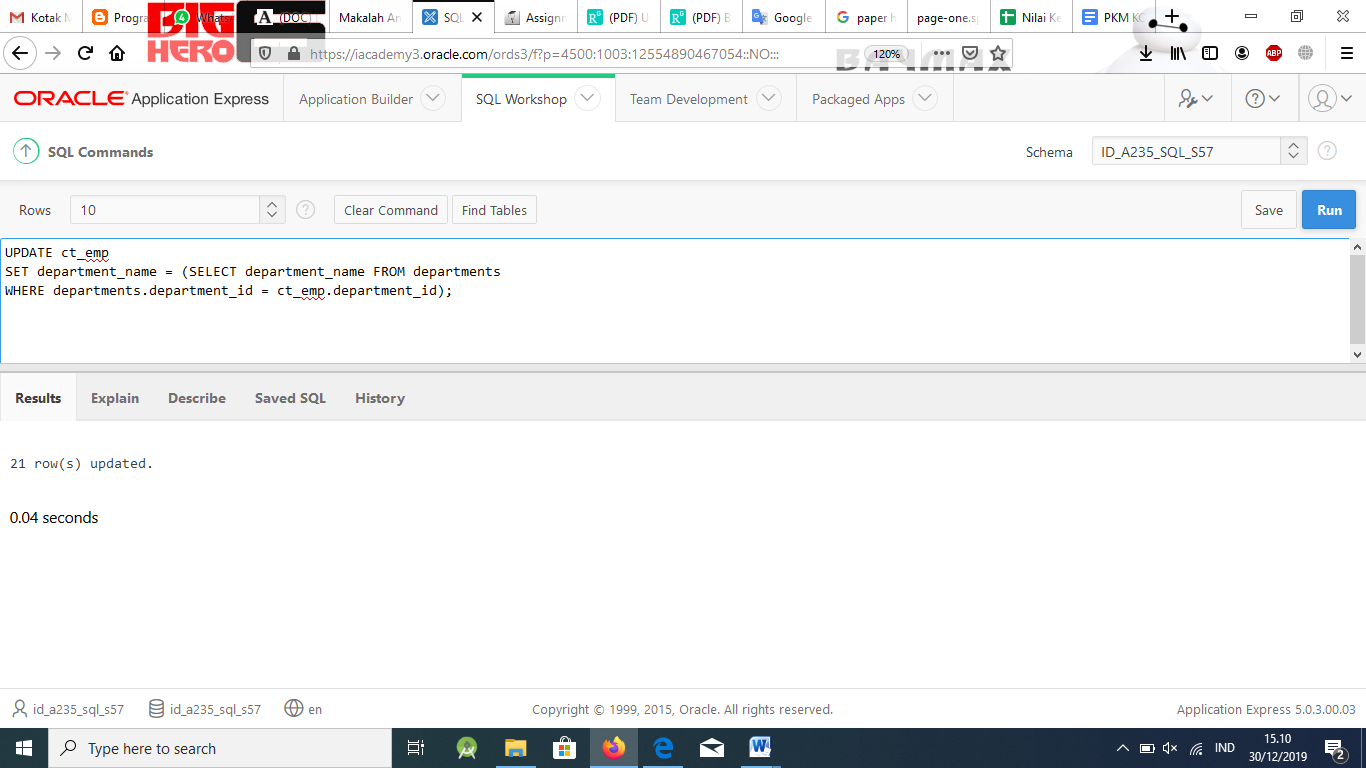
ADD department\_name varchar2(30)



UPDATE ct\_emp

SET department\_name = (SELECT department\_name FROM departments

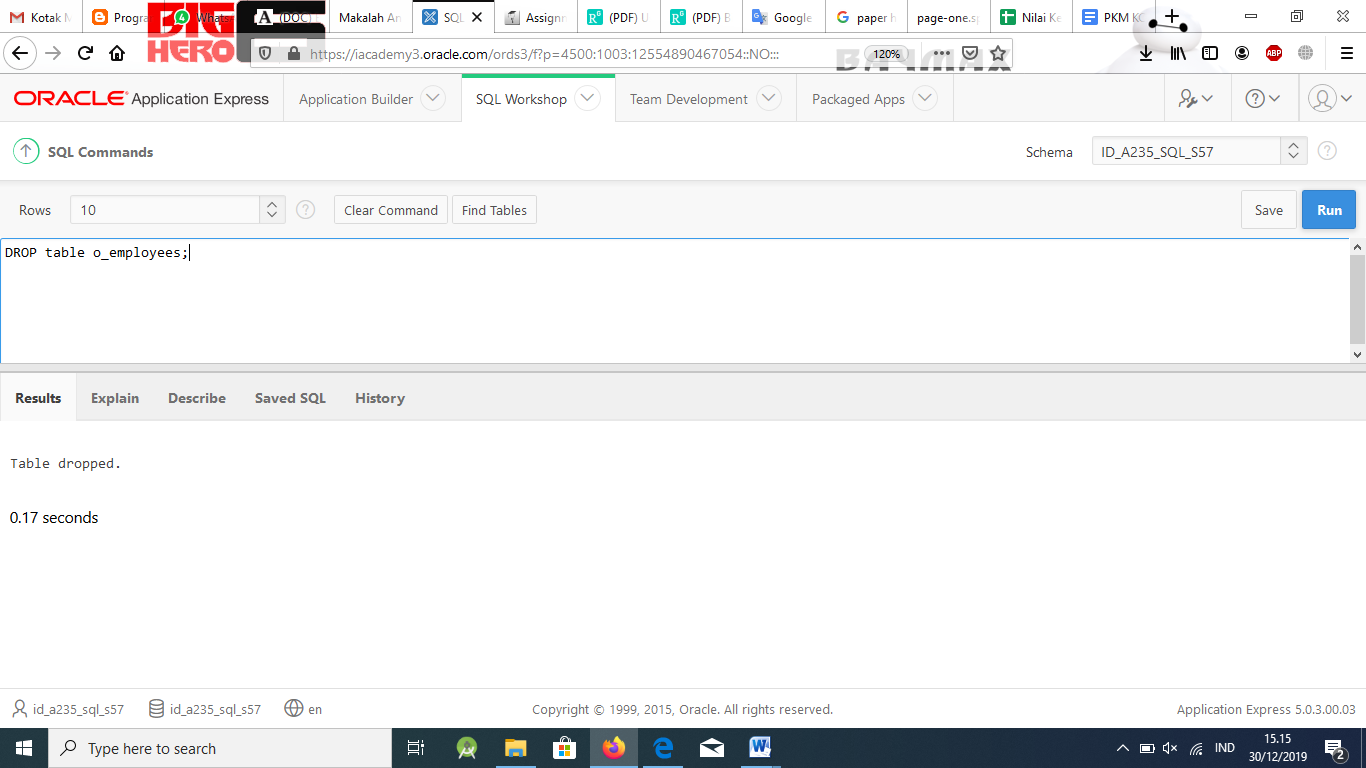
WHERE departments.department\_id = ct\_emp.department\_id);



1. Drop the o\_employees table. Investigate where it went. Try to select from the dropped table. Now try to select from the dropped table using its bin-name. Restore the table.

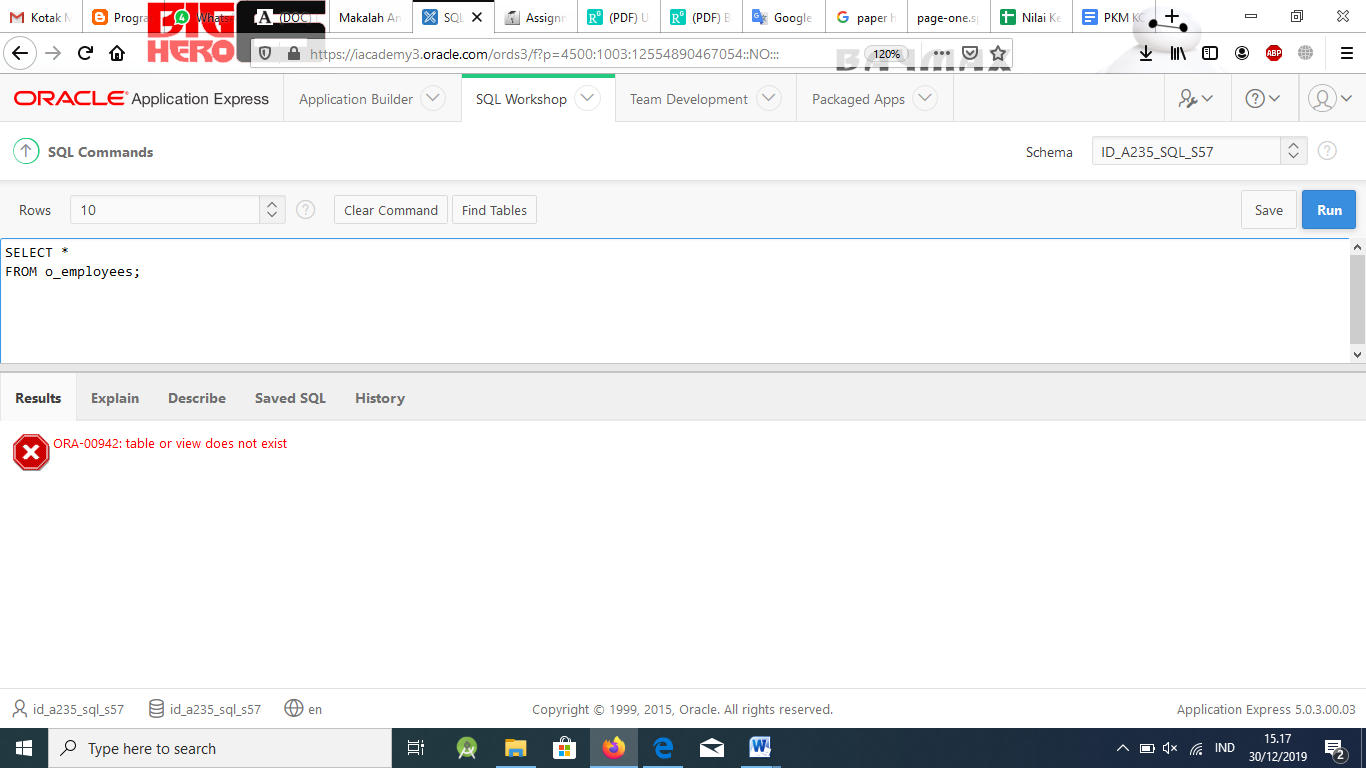
# Solution:

DROP table o\_employees;



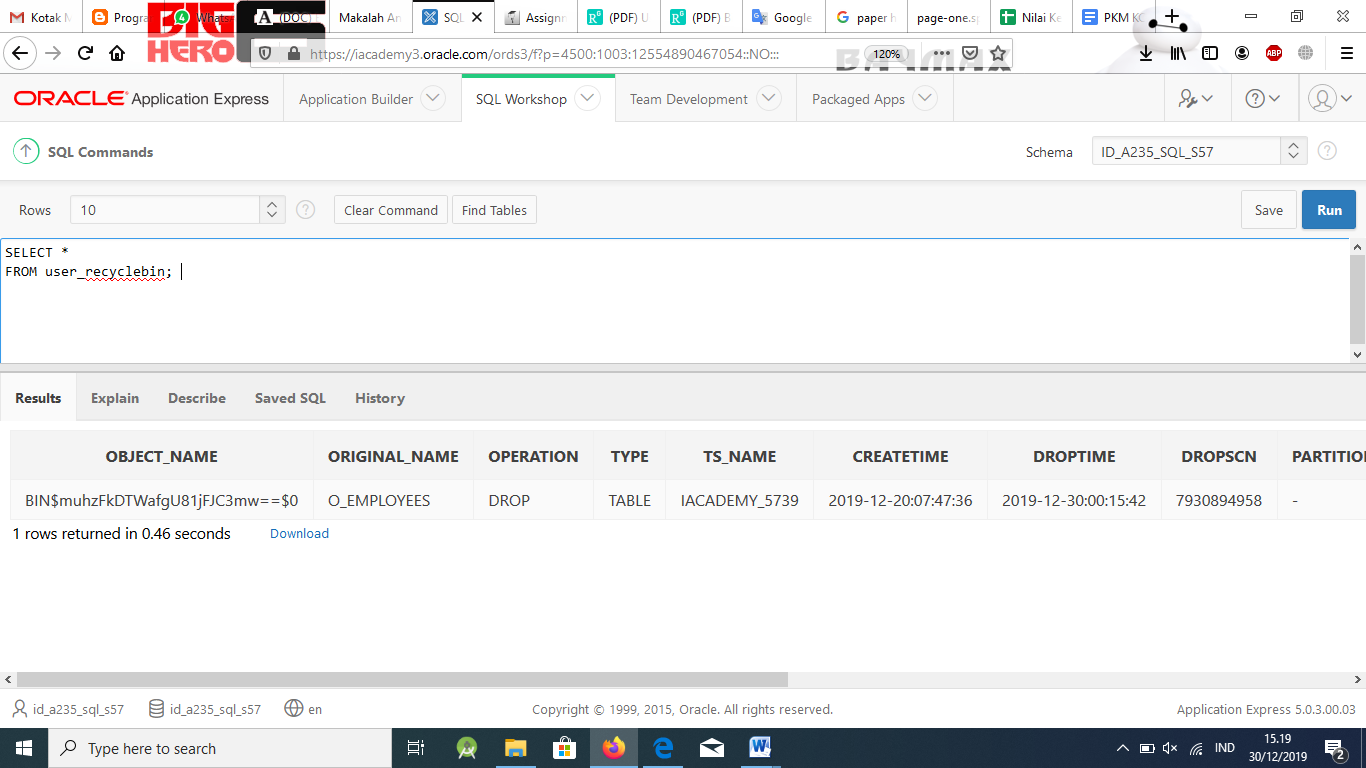
SELECT \*

FROM o\_employees;



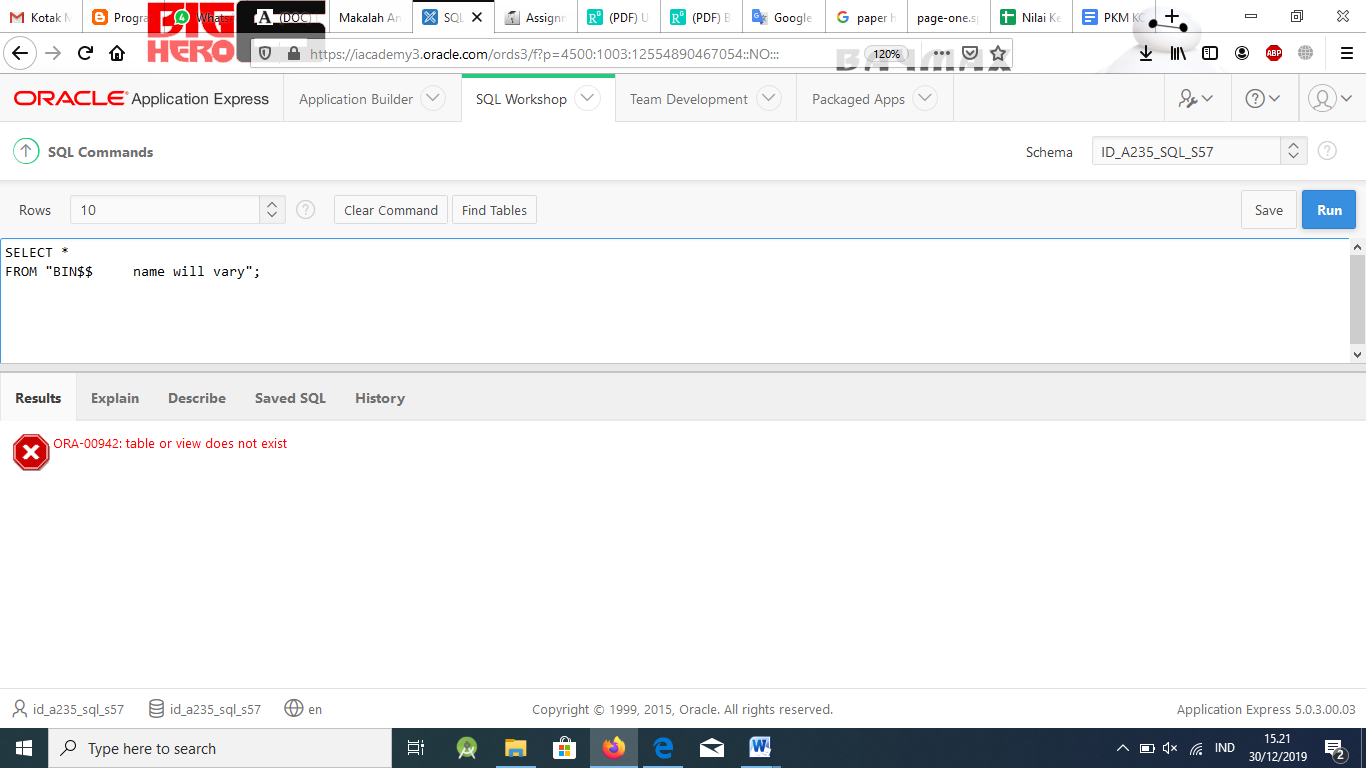
SELECT \*

FROM user\_recyclebin;



SELECT \*

FROM "BIN$$ name will vary";



FLASHBACK TABLE o\_employees TO BEFORE DROP;

# Part 6

1. Which of the statements listed below are true about the following sequence? The sequence was used to generate numbers for the DJs on Demand D\_CDS table cd\_numbers column.

CREATE SEQUENCE cd\_numbers\_sq INCREMENT BY 15

START WITH 105

MAXVALUE 999 NOMINVALUE CYCLE NOCACHE

a. The value 165 will not be generated by this sequence.

b. The value 999 will not be generated by this sequence.

c. This sequence would be appropriate for PRIMARY KEY values.

d. If the D\_CDS table is deleted, the sequence is also deleted.

e. If the system fails sequence, values will be lost.

f. USER\_OBJECTS documents this sequence in the data dictionary.

g. The START WITH value could be changed with an ALTER SEQUENCE.

# Solution:

b. The value 999 will not be generated by this sequence

f. USER\_OBJECTS documents this sequence in the data dictionary.

1. In the CREATE TABLE statement shown below, circle the letter of the line(s) that will automatically create an index.
   1. CREATE TABLE employees(
   2. employee\_id NUMBER(6),
   3. last\_name VARCHAR2(25) NOT NULL,
   4. email VARCHAR2(25) CONSTRAINT emp\_email\_uk UNIQUE,
   5. salary NUMBER(8,2),
   6. commission\_pct NUMBER(2,2),
   7. hire\_date DATE NOT NULL,
   8. CONSTRAINT employee\_id\_pk PRIMARY KEY(employee\_id),
   9. CONSTRAINT emp\_dept\_fk FOREIGN KEY(department\_id)REFERENCES
   10. departments(department\_id)

# Solution:

d. email VARCHAR2(25) CONSTRAINT emp\_email\_ukUNIQUE,

h. CONSTRAINT employee\_id\_pk PRIMARY KEY(employee\_id),

1. What data dictionary view contains the sequences created by a schema?
   1. SEQUENCE\_VIEW
   2. USER\_SEQUENCES
   3. SEQUENCE\_NAMES
   4. USER\_VIEWS

# Solution:

b. USER\_SEQUENCES

1. What is the proper syntax for changing the emp\_email\_index on the email column of the employees table?
   1. CREATE INDEX emp\_email\_index ON employees (email);
   2. ALTER INDEX emp\_mail\_index FROM employees(email);
   3. CREATE AND REPLACE INDEX emp\_email\_index ON employees, COLUMN = email;
   4. ADD INDEX emp\_email\_index INTO employees (email);
   5. None of the above

Indexes cannot be modified – instead, DROP and re-create.

# Solution:

e. None of the above

Indexes cannot be modified – instead, DROP and re-create.

1. Which data dictionary view is used to view the table name, the index name, and the column name that the index is based on?
   1. USER\_TABLE\_INDEXES
   2. USER\_COLUMNS
   3. USER\_IND\_COLUMNS
   4. ALL\_INDEXES

# Solution:

c. USER\_IND\_COLUMNS

1. If the PUBLIC option is not included in the CREATE SYNONYM syntax, which of the following is True?
   1. Users cannot lengthen object names.
   2. The DBA does not have access to the synonym.
   3. The synonym is not accessible to all users.
   4. All object names must be qualified with PUBLIC.

# Solution:

c. The synonym is not accessible to all users.

1. What are the last three numbers created by the sequence?

CREATE SEQUENCE sample\_seq INCREMENT BY 3

START WITH 6

MAXVALUE 14 NOMINVALUE CYCLE NOCACHE

a. 3, 6, 9

b. 6, 9, 12

c. 9, 12, 15

d. 9, 12, 14

# Solution:

b. 6, 9, 12

1. Which of the following is a good reason to create an index?
   1. A column has a large number of null values.
   2. A column has a narrow range of values.
   3. Most queries are expected to return more than 5% of the rows.
   4. A table is updated frequently.

# Solution:

c. Most queries are expected to return more than 5% of the rows.

1. Which of the following constraints can be created at the table level?
   1. NOT NULL
   2. FOREIGN KEY
   3. PRIMARY KEY
   4. UNIQUE COMPOSITE KEY

# Solution:

1. FOREIGN KEY
2. PRIMARY KEY
3. UNIQUE COMPOSITE KEY
4. Which of the following cannot contain a single-row function?
   1. SELECT
   2. FROM
   3. WHERE
   4. ORDER BY

# Solution:

b. FROM