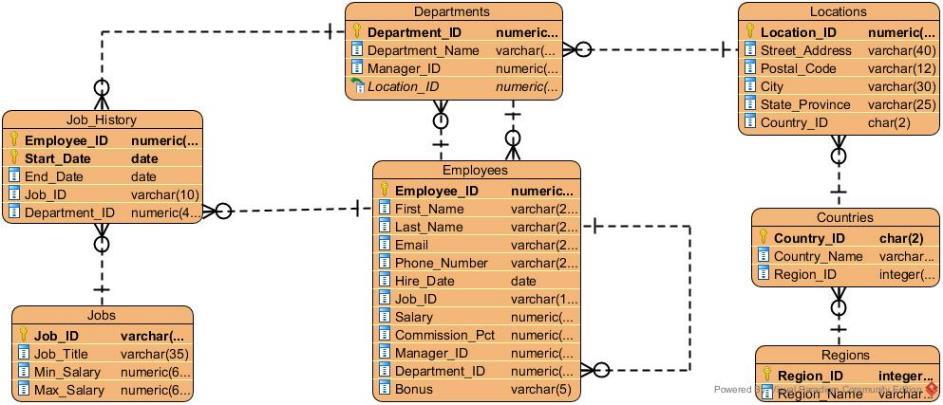
# Topic 9 – Data Definition Language

## Familiarise yourself with the Human Resource (HR) schema from Oracle

Whereas in the lectures we use a small database consisting of only three table, for the labs we will be using the demo tables (schema) supplied by Oracle. The schema describes a human resources database used by a large multi-national company.

Use the Entity Relationship Diagram (ERD) below to familiarise yourself with the tables, relationships and attributes.



# Worked Examples

### Q1. The company has given a pay rise of 10% to every employee whose salary is less than half the company’s average salary. Write the SQL to update the table appropriately.

This question is about changing the data in a table and so uses the UPDATE statement. The general format for the UPDATE statement is:

UPDATE table\_name

SET column\_name = new\_value

WHERE condition;

In this case, the condition relies on information from the entire table so we need a subquery for that part. Since we are designing a query with a subquery, let’s follow the usual three steps. First, we will design the outer query using a placeholder:

UPDATE employees

SET salary = salary \* 1.1

WHERE salary < half\_average;

Next, we design the inner query. We need a query that will give us half the average salary:

SELECT AVG(salary)\*0.5

FROM employees;

Finally, we insert the inner query into the outer query and we have our final SQL:

UPDATE employees

SET salary = salary \* 1.1

WHERE salary < (SELECT AVG(SALARY)\*0.5 FROM employees);

**IMPORTANT NOTE**

Whenever you make changes to the data in this worksheet, remember to undo them. This is done by rolling back the changes. You can either use the F12 key or press the Rollback button:



### Q2. Add a new column to the locations table, giving the date when the company first operated there

This question is asking us to change the structure of an existing table by adding a new column. We do this with the ALTER statement. There is no single convenient general structure of the ALTER statement, but for adding new columns the structure is

ALTER TABLE table\_name

ADD column\_name data\_type;

In this case we need a date, so the DATE type is best and a sensible name for the column is something like “Date Acquired”. The SQL statement is therefore:

ALTER TABLE locations

ADD “Date Acquired” DATE;

If you run the command and then run SELECT \* FROM locations; you should see the new column filled with NULL values.

For now, let’s remove the added column to undo the change. For that we need another ALTER statement. This time the format is:

ALTER TABLE table\_name

DROP COLUMN column\_name;

For our case it would be:

ALTER TABLE locations

DROP COLUMN “Date Acquired”;

### Q3. Insert a new record about a new employee called “James Morrison” who will start working in the IT department on 5th January next year as a programmer. His salary is not determined yet but he has been assigned employee ID 210.

This question obviously requires an INSERT statement in order to add new information. Because we don’t have the complete information we must specify which columns we are going to insert data into.

When inserting the data we need to provide a department\_id and job\_id. We could look those up from the relevant tables but we can be cleverer and use a subquery in the INSERT statement to automatically get the relevant ID from the name.

The final SQL would be:

INSERT INTO employees (employee\_id, first\_name, last\_name, email,

hire\_date, job\_id, department\_id)

VALUES(210, ‘James’, ‘Morrison’, ‘JMORRISON’, ’05-JAN-18’,

(SELECT job\_id FROM jobs WHERE job\_title = ‘Programmer’),

(SELECT department\_id FROM departments

WHERE department\_name = ‘IT’));

Make sure to remove the new record using a DELETE statement:

DELETE FROM employees WHERE employee\_id = 210;

# Deliberate Practice: Write the SQL

### Q1. Add a new column to the departments table called “Notes”

ALTER TABLE departments

ADD “Notes” VARCHAR2(200);

### Q2. The company is looking to expand and is going to start operation in France, add it to the countries table.

INSERT INTO countries (country\_id, country\_name, region\_id) VALUES (‘FR’, ‘France’, (SELECT region\_id FROM regions WHERE region\_name = ‘Europe’));

### Q3. All managers in the company are getting a 2.5% pay rise. Update the employees table appropriately.

UPDATE employees

SET salary = salary\*1.025

WHERE employee\_id = ANY(SELECT manager\_id FROM employees);

### Q4. List the region names and the number of people working in each one.

SELECT region\_name, COUNT(employee\_id)

FROM employees

INNER JOIN departments USING(department\_id)

INNER JOIN locations USING(location\_id)

INNER JOIN countries USING(country\_id)

RIGHT JOIN regions USING(region\_id)

GROUP BY region\_name;

### Q5. Write an SQL statement to make the result of Q4 into a View.

CREATE VIEW *reg\_view* AS  
SELECT region\_name, COUNT(employee\_id)

FROM employees

INNER JOIN departments USING(department\_id)

INNER JOIN locations USING(location\_id)

INNER JOIN countries USING(country\_id)

RIGHT JOIN regions USING(region\_id)

GROUP BY region\_name;

### Q6. Eleni Zlotkey was found to have stolen from the company and has been fired. Update the database accordingly.

DELETE employee

WHERE first\_name = “Eleni” AND surname = “Zlotkey”;

### Q7. Write the SQL statements to create the table we have been using throughout the lectures and to insert the data they have.

CREATE TABLE "EMPLOYEES"

("EMPLOYEE\_ID" NUMBER(6,0),

"FIRST\_NAME" VARCHAR2(20),

"LAST\_NAME" VARCHAR2(25),

"EMAIL" VARCHAR2(25),

"PHONE\_NUMBER" VARCHAR2(20),

"HIRE\_DATE" DATE,

"JOB\_ID" VARCHAR2(10),

"SALARY" NUMBER(8,2),

"COMMISSION\_PCT" NUMBER(2,2),

"MANAGER\_ID" NUMBER(6,0),

"DEPARTMENT\_ID" NUMBER(4,0),

"BONUS" VARCHAR2(5),

CONSTRAINT "EMP\_SALARY\_MIN" CHECK (salary > 0),

CONSTRAINT "EMP\_ID\_PK" PRIMARY KEY ("EMPLOYEE\_ID")

);

INSERT INTO employees(employee\_id,first\_name,last\_name,email,phone\_number,hire\_date,job\_id,salary,commission\_pct,manager\_id,department\_id)

VALUES(100,'Steven','King','SKING','515.123.4567',TO\_DATE('1987-06-17','yyyy-mm-dd'),'AD\_PRES',24000,null,null,90);

INSERT INTO employees(employee\_id,first\_name,last\_name,email,phone\_number,hire\_date,job\_id,salary,commission\_pct,manager\_id,department\_id)

VALUES(101,'Neena','Kochhar','NKOCHHAR','515.123.4568',TO\_DATE('1989-09-21','yyyy-mm-dd'),'AD\_VP',17000,null,100,90 );

INSERT INTO employees(employee\_id,first\_name,last\_name,email,phone\_number,hire\_date,job\_id,salary,commission\_pct,manager\_id,department\_id)

VALUES(102,'Lex','De Haan','LDEHAAN','515.123.4569',TO\_DATE('1993-01-13','yyyy-mm-dd'),'AD\_VP',17000,null,100,90 );

INSERT INTO employees(employee\_id,first\_name,last\_name,email,phone\_number,hire\_date,job\_id,salary,commission\_pct,manager\_id,department\_id)

VALUES(200,'Jennifer','Whalen','JWHALEN','515.123.4444',TO\_DATE('1987-09-17','yyyy-mm-dd'),'AD\_ASST',4400,null,101,10 );

INSERT INTO employees(employee\_id,first\_name,last\_name,email,phone\_number,hire\_date,job\_id,salary,commission\_pct,manager\_id,department\_id)

VALUES(205,'Shelley','Higgins','SHIGGINS','515.123.8080',TO\_DATE('1994-06-07','yyyy-mm-dd'),'AC\_MGR',12000,null,101,110 );

INSERT INTO employees(employee\_id,first\_name,last\_name,email,phone\_number,hire\_date,job\_id,salary,commission\_pct,manager\_id,department\_id)

VALUES(206,'William','Gietz','WGIETZ','515.123.8181',TO\_DATE('1994-06-07','yyyy-mm-dd'),'AC\_ACCOUNT',8300,null,205,110 );

INSERT INTO employees(employee\_id,first\_name,last\_name,email,phone\_number,hire\_date,job\_id,salary,commission\_pct,manager\_id,department\_id, bonus)

VALUES(149,'Eleni','Zlotkey','EZLOTKEY','011.44.1344.429018',TO\_DATE('2000-01-29','yyyy-mm-dd'),'SA\_MAN',10500,.2,100,80, '1500' );

INSERT INTO employees(employee\_id,first\_name,last\_name,email,phone\_number,hire\_date,job\_id,salary,commission\_pct,manager\_id,department\_id, bonus)

VALUES(174,'Ellen','Abel','EABEL','011.44.1644.429267',TO\_DATE('1996-05-11','yyyy-mm-dd'),'SA\_REP',11000,.3,149,80,'1700' );