1. ¿Cómo se llama el grupo de sentencias SELECT, INSERT, DELETE, UPDATE?
2. En SQL, para eliminar las filas duplicadas del resultado de una sentencia SELECT se emplea

a) NO DUPLICATE

b) UNIQUE

c) DISTINCT

d) Las anteriores respuestas no son correctas

1. En SQL, para modificar la estructura de una tabla de una base de datos se emplea la instrucción

a) ALTER TABLE

b) CHANGE TABLE

c) MODIFY TABLE

d) Las anteriores respuestas no son correctas

1. Una sentencia SELECT sin la cláusula WHERE devuelve

a) Todos los registros existentes en la tabla que no estén relacionados con otra tabla

b) Todos los registros existentes en la tabla

c) No se puede ejecutar una sentencia SELECT sin la cláusula WHERE

d) Las anteriores respuestas no son correctas

1. En SQL, para ordenar los datos devueltos por una sentencia SELECT se emplea la cláusula

a) ORDER BY

b) ORDERED BY

c) SORT BY

d) SORTED BY

1. En una cláusula LIKE, ¿cómo se obtienen todos los nombres de personas que comienzan con "Juan"?

a) LIKE "Juan%"

b) LIKE "Juan\*"

c) LIKE "Juan$"

d) LIKE "Juan&"

1. ¿Cuál de las siguientes no es una función de agregación?

a) AVG()

b) FLOOR()

c) SUM()

d) Las anteriores respuestas no son correctas

1. ¿Cómo se borra toda una base de datos con SQL?

a) DELETE DATABASE

b) DROP DATABASE

c) ERASE DATABASE

d) Las anteriores respuestas no son correctas

1. ¿Cuál de las siguientes no es una función de agregación?

a) COUNT()

b) LIMIT()

c) MAX()

d) MIN()

1. ¿En cuál de las siguientes sentencias del lenguaje SQL se emplea la cláusula SET?

a) DELETE

b) DROP

c) SELECT

d) UPDATE

1. ¿Qué instrucción se emplea para eliminar todo el contenido de una tabla, pero conservando la tabla?

a) DELETE TABLE

b) DROP TABLE

c) TRUNCATE TABLE

d) Las anteriores respuestas no son correctas

1. El left outer join es un tipo de outer join ¿Otro tipo de outer join sería?

right outer join

full outer join

1. ¿Que son las funciones de agregación y cuáles son?
2. Con base en el siguiente esquema:

TA\_EMPLOYEES

|  |
| --- |
| EMPLOYEE\_ID |
| FIRST\_NAME |
| LAST\_NAME |
| EMAIL |
| PHONE\_NUMBER |
| HIRE\_DATE |
| JOB\_ID |
| SALARY |
| COMMISSION\_PCT |
| MANAGER\_ID |
| DEPARTMENT\_ID |

TA\_JOBS

|  |
| --- |
| JOB\_ID |
| JOB\_TITLE |
| MIN\_SALARY |
| MAX\_SALARY |

TA\_JOB\_HISTORY

|  |
| --- |
| EMPLOYEE\_ID |
| START\_DATE |
| END\_DATE |
| JOB\_ID |
| DEPARTMENT\_ID |

Realice los siguientes ejercicios:

* 15. Recuperar todos los empleados que tengan un salario entre 1000 y 2000 o menor a 950. Considere lo siguiente:
* Last\_name no inicia con S
* DEPARTMENT\_ID puede ser 100, 120, 150 ó 180
* Last\_name no finaliza con ‘a’

SELECT \*

FROM EMPLOYEES

WHERE SALARY BETWEEN 1000 AND 2000

OR SALARY < 950

AND DEPARTMENT\_ID IN (100, 120, 150, 180)

AND LAST\_NAME NOT LIKE ‘S%’

AND LAST\_NAME NOT LIKE ‘%a’

* 16. Extraer todos los empleados que tengan sueldo mayor a 10000
  + Validar que el sueldo por job\_id este entre 10,000 y 20,000

SELECT \*

FROM EMPLOYEES

WHERE JOB\_ID IN (

SELECT JOB\_ID

FROM JOBS

WHERE MIN\_SALARY >= 10000

AND MAX\_SALARY < 41000);

* 17. Mostrar el nombre y la cantidad de roles que ha tenido un empleado de acuerdo al historial

SELECT E.EMPLOYEE\_ID, E.FIRST\_NAME, E.LAST\_NAME, COUNT(JH.EMPLOYEE\_ID ) AS CANTIDAD\_ROLES

FROM EMPLOYEES E

JOIN JOB\_HISTORY JH

ON E.EMPLOYEE\_ID = JH.EMPLOYEE\_ID

GROUP BY E.EMPLOYEE\_ID, E.FIRST\_NAME, E.LAST\_NAME;