1. **Comprueba que tienes desactivada la confirmación automática, si no es así desactívala para poder continuar.**

show autocommit

1. **Introduce en la tabla departments, un nuevo departamento con identificador 300, nombre Finanzas su jefe tiene el código 110 y su código de localización es 1800.**

INSERT INTO departments

VALUES (300,'Finanzas',110,1800);

**Comprueba que la fila se ha insertado correctamente en la tabla, si es así, confirma la operación.**

1. **Introduce en la tabla department, un nuevo departamento con identificador 400, nombre Finanzas2 su jefe tiene el código 389 y su código de localización es 1800.**

**¿Qué respuesta obtienes? ¿Por qué?**

No encuentra la relacion entre el dato que he dado en la fk en la pk de la otra tabla

1. **Introduce en la tabla employees el empleado Ricardo Marcelo con identificador 401, su mail es rmarcelo@gmail.com , la fecha de contratación es la actual, va a ganar 1600 € y trabajará de Jefe de Ventas ,SA\_MAN. No confirmes los cambios**

insert into employees(employee\_id, first\_name, last\_name, email, hire\_date, job\_id, salary)

values(401,'Ricardo','Marcelo','rmarcelo@gmail.com',sysdate,'SA\_MAN',1600);

1. **Aborta la operación anterior y comprueba que la fila ha desaparecido de la tabla.**

rollback

1. **Crea una nueva tabla llamada prueba según la siguiente sintaxis:**

**create table prueba(**

**id number(6) primary key,**

**name varchar2(20),**

**hire\_date date,**

**salary number(8,2));**

1. **Introduce en la tabla prueba los datos correspondientes de todos los empleados del departamento 50**

INSERT INTO prueba select employee\_id, first\_name||' '||last\_name as nombre, hire\_date,salary from employees where department\_id=30;

1. **Introduce en la tabla prueba un nuevo empleado con identificador 390, nombre Julio y el resto de datos son los mismos que los del empleado con identificador 119.**

insert into prueba select e1.employee\_id, e1.first\_name, e2.hire\_date, e2.salary from employees e1,(select hire\_date, salary from employees e2

where employee\_id=119); //ESTA MAL

insert into prueba select 390, 'JULIO', hire\_date, salary

from employees

where employee\_id=119; //ESTA BIEN

1. **Introduce en la tabla prueba los datos de los empleados del departamento 90.**

insert into prueba select employee\_id,first\_name,hire\_date,salary from employees where department\_id=90;

1. **Borra todas las filas de la tabla prueba. Confirma la operación.**

truncate table prueba;

1. **Introduce en la tabla prueba los datos de los empleados con el menor salario de cada uno de sus departamentos.**

insert into prueba select employee\_id, first\_name, hire\_date, salary from employees,(select min(salary) as salario ,department\_id as dep from employees

group by department\_id)

where employees.salary=salario and employees.department\_id=dep;

1. **Modifica el salario del empleado 200 de la tabla prueba a 5000 €.**

update prueba set salary=5000 where id=200;

1. **Modifica los salarios de la tabla prueba y auméntalos un 3%.**

update prueba set salary=salary\*1.03;

1. **Modifica la fecha de contratación de los empleados de la tabla prueba y auméntala 1 año (Utiliza la función ADD\_MONTHS(fecha,n)).**

update prueba set hire\_date=add\_months(hire\_date, 12);

1. **El empleado de la tabla employees 105 ha sido trasladado al departamento 100.**

update employees set department\_id=100 where employee\_id=105;

1. **El empleado 107 de la tabla prueba cambia su nombre y salario por el nombre y salario del empleado 100.**

update employees set department\_id=100 where employee\_id=105;

1. **El empleado 107 ha cambiado al departamento de Daniel Faviet(109) y su salario ha cambiado al salario máximo correspondiente a su puesto de trabajo. Ejecuta la sentencia que modifique los datos.**
2. **Ejecuta la siguiente sentencia ¿Qué respuesta da Oracle? ¿Por qué?**

**insert into employees (employee\_id, last\_name, email, job\_id)values (800, 'Lopez', 'lopez@hotmail.com', 'SA\_PAT')**

Que no se puede dejar hire\_date como null porque al crearse se expecifico que no podía ser nulo (NOTNULL)

1. **Ejecuta ahora la siguiente orden, ¿Qué respuesta da Oracle? ¿Por qué?**

**insert into employees (employee\_id, last\_name, email, hire\_date,job\_id)values (800, 'Lopez', 'lopez@hotmail.com',sysdate, 'SA\_PAT')**

SA\_PAT NO ESTA EN LA TABLA JOBS DE LA Q VIENE JOB\_ID

1. **Ejecuta la siguiente orden ¿Qué respuesta da Oracle?¿Como es está definida la columna comisión\_pct? Consulta la estructura de la tabla desde el explorador de objetos.**

**Update employees set commission\_pct =300 where employee\_id =100**

NO SE PUEDE PORQUE COMMISSION\_PCT ESTA COMO NUMBER Y COMO MUCHO DOS DIGITOS

1. **Crea la tabla jobs2 con la siguiente sintaxis:**

**Create table jobs2 as select \* from jobs**

**Comprueba su estructura mediante la orden Desc jobs2. Comprueba ahora qué datos tiene la tabla jobs2**

1. **Actualiza el salario mínimo de la tabla jobs de modo que sea igual al salario máximo multiplicado por 0.75 para todos aquellos puestos de trabajo cuyo nombre empieza por A**
2. **Borra la fila de la tabla job correspondiente trabajo Sales Manager.**
3. **Borra los trabajos de la tabla jobs2 en los que el salario máximo mayor que 1.7 veces el salario mínimo.**
4. **Crea la tabla países de forma que sea una copia de la tabla countries (ver ejercicio21).**
5. **Borra de la tabla regions el continente 3, region\_id=3 ¿Qué ocurre?¿Por qué? Borra todos los datos de la tabla paises2 de dos formas distintas.**
6. **Borra todos los datos de la tabla paises de dos formas distintas.**
7. **Confirma las operaciones.**