

1. Comprueba que tienes desactiva la confirmación automática.
2. Introduce en la tabla departamentos, un nuevo departamento con identificador 300, nombre Finanzas su jefe tiene el código 110 y su código de localización es 1800. comprueba que la fila se ha insertado correctamente en la tabla, si es así, confirma la operación.

```
insert into departments  
values(300, 'Finanzas', 110, 1800)
```

3. Introduce en la tabla departamentos, un nuevo departamento con identificador 400, nombre Finanzas2 su jefe tiene el código 389 y su código de localización es 1800. ¿Qué respuesta obtienes?¿Porqué?

```
insert into departments  
values(400, 'Finanzas2', 389, 1800)
```

**No permite introducirlo porque no existe el empleado de codigo 389**

4. Introduce en la tabla empleados el empleado Ricardo Marcelo con identificador 401, su mail es [rmarcelo@gmail.com](mailto: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)
```

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

### **Rollback**

6. 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))
```

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

```
insert into prueba  
select employee_id, first_name, hire_date, salary from employees  
where department_id=50
```

8. 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 390, 'Julio', hire_date, salary from employees  
where employee_id=119
```

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

```
insert into prueba  
select employee_id, last_name, hire_date, salary from employees  
where department_id=90
```

10. Borra todas las filas de la tabla prueba. Confirma la operación

**Delete from prueba; o bien Truncate table prueba**

11. 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, last_name, hire_date, salary from employees  
where salary in (select min(salary) from employees  
group by department_id)
```

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

```
update prueba  
set salary=5000  
where id=200
```

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

```
update prueba  
set salary=salary*1.03
```

14. 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)
```

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

```
update employees  
set department_id=100  
where employee_id= 105
```

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

```
update prueba  
set (name, salary)=(select last_name, salary from employees where employee_id=100)  
where id=107
```

17. 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

```
update employees e  
set department_id=(select department_id from employees where employee_id=109),  
salary=(select max_salary from jobs where job_id=e.job_id)  
where employee_id= 107
```

18. Ejecuta la siguiente sentencia ¿Qué respuesta da Oracle?¿Porqué?

```
insert into employees (employee_id, last_name, email, job_id)values (800, 'Lopez',  
'lopez@hotmail.com', 'SA_PAT')
```

**No permite porque la fecha de contratación es obligatoria**

19. Ejecuta ahora la siguiente orden, ¿Qué respuesta da Oracle?¿Porqué?

```
insert into employees (employee_id, last_name, email, hire_date,job_id)values (800, 'Lopez',  
'lopez@hotmail.com',sysdate, 'SA_PAT')
```

**No permite porque el job\_id no existe en la tabla jobs**

20. 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 Examinar>Tabla en la página inicial de la BD**

```
Update employees set commission_pct =300 where employee_id =100
```

21. 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

22. 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

```
update jobs2 set min_salary=max_salary*0.75  
where job_id like 'A%'
```

23. Borra la fila de la tabla job correspondiente trabajo Sales Manager

```
delete from jobs2  
where job_id='SA_MAN'
```

24. Borra los trabajos de la tabla jobs2 en los que el salario máximo mayor que 1.7 veces el salario mínimo.

```
delete from jobs2  
where max_salary>1.7*min_salary
```

25. Crea la tabla paises de forma que sea una copia de la tabla countries y borra despues todos los paises que están en el mismo continente que (ver ejercicio21).Australia

```
delete from paises  
where region_id =(select region_id from paises where country_name='Australia')
```

26. Borra de la tabla regions el continente 3, region\_id=3 ¿Qué ocurre?¿Porqué?

**Delete from regiones where region\_id=3. No permite porque hay registros relacionados en otra tabla**

27. Borra todos los datos de la tabla paises2 de dos formas distintas.

**Delete from paises2 o bien truncate table paises2**

28. Confirma las operaciones

**Commit**