

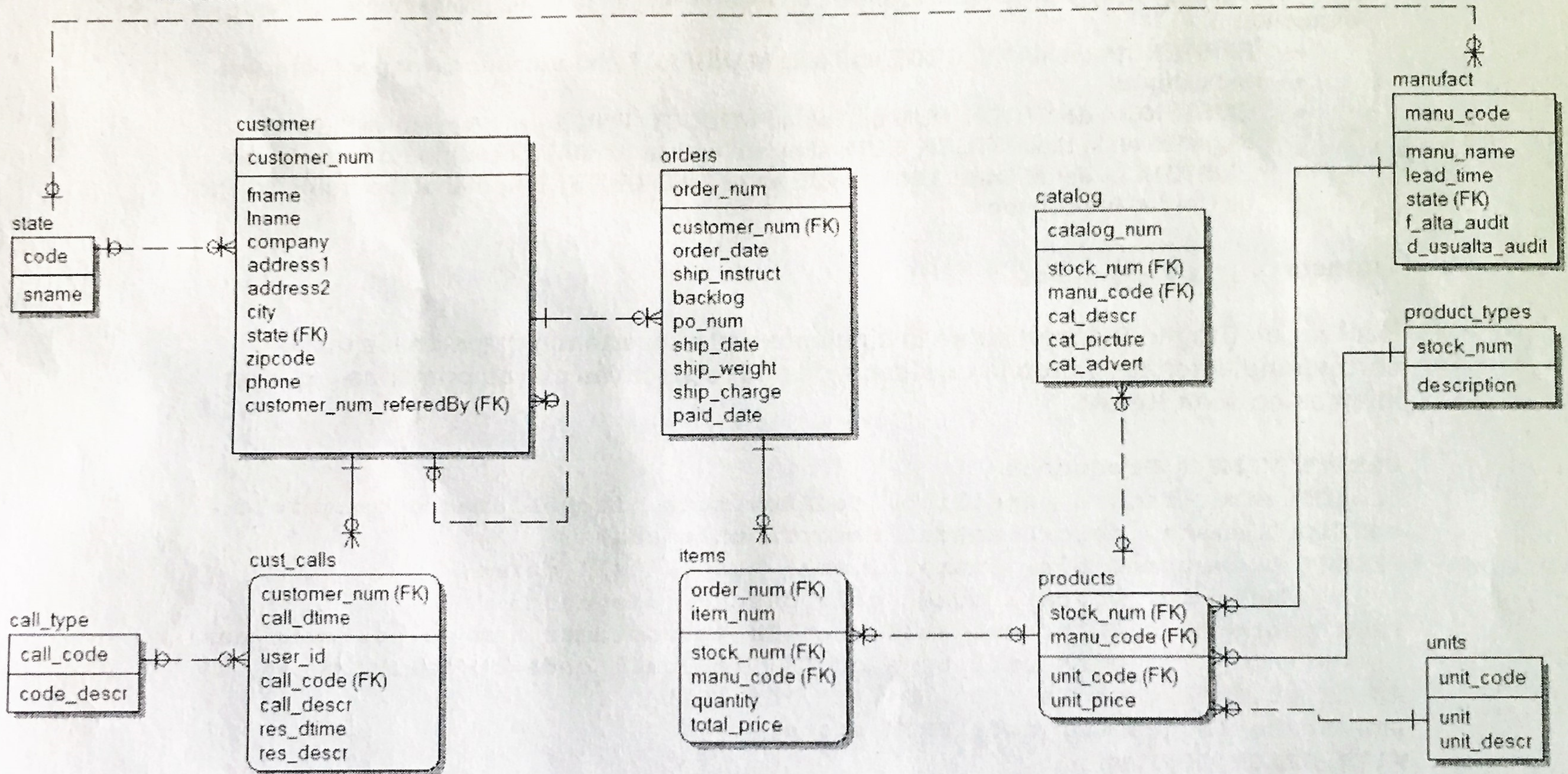
b *A*

PARTE 1

Teoría

- En no más de 15 renglones explique todo lo relacionado con objeto Vista.
- En una carilla explique la Funcionalidad de Seguridad de un RDBMS detallando someramente los objetos relacionados con la misma.

Práctica



- Crear una consulta que devuelva:
La siguientes cuatro atributos

Apellido, Nombre AS Cliente,
 Suma de todo lo comprado por el cliente AS totalCompra,
 Apellido, Nombre AS ClienteReferido,
 Suma de lo comprado por el referido*0.05 AS totalComision

Consideraciones.

- En el caso que un no tenga OCs deberá mostrar 0 en el campo totalCompra
- En el caso que un Cliente no tenga Referidos deberá mostrar al mismo con NULL en las columnas ClienteReferido y totalComision.
- Para calcular la comisión del cliente se deberán sumar (cant*precio) de todos los productos comprados por el ClienteReferido cuyo stock_num sea 1,4,5,6,9. La comisión es del 5%.
- Se deberá ordenar la salida por el Apellido y Nombre del Cliente.

No se pueden utilizar tablas temporales, ni funciones de usuario.

PARTE 2

d. Stored Procedures

Desarrollar un stored procedure maneje la inserción o modificación de un producto determinado.

Parámetros de Entrada STOCK_NUM, MANU_CODE, UNIT_PRICE, UNIT_CODE, DESCRIPTION

Si existe el producto en la tabla PRODUCTS actualizar los atributos que no pertenecen a la clave primaria.

Si no existe el producto en la tabla PRODUCTS Insertar fila en la tabla, previamente validar lo siguiente:

- EXISTENCIA de MANU_CODE en Tabla MANUFACT - Informando Error por Fabricante Inexistente.
- EXISTENCIA de STOCK_NUM en Tabla PRODUCT_TYPES - Si no existe INSERTAR Registro en la tabla STOCK_NUM, si existe realizar UPDATE del atributo 'description'.
- EXISTENCIA del atributo UNIT_CODE en la Tabla UNITS - Informando Error porCodigo de Unidad Inexistente

e. Triggers

Realizar un trigger de Insert sobre la siguiente Vista, insertando datos en las tablas correspondientes validando la existencia de sus respectivas claves primarias y que las mismas no sean NULAS.

```

CREATE VIEW v_Productos
(codCliente, nombre, apellido, codProvincia, fechaLlamado, usuarioId,
codTipoLlamada, descrLlamada, descrTipoLlamada)
SELECT c.customer_num, fname, lname, state, call_dtime,
      user_id, cc.call_code, call_descr, code_descr
FROM customer c JOIN cust_calls cc ON (c.customer_num=cc.customer_num)
      JOIN call_type ct ON (cc.call_code=ct.call_code)
WHERE ct.call_code IN ('B','D','I','L','O')
AND state IN (SELECT code FROM state)
WITH CHECK OPTION
    
```

Tener en cuenta en el trigger la inserción de múltiples filas.

Parte 1		
a	b	c
Nota		

Parte 2	
d	e
Nota	