

I) a)

```
CREATE TABLE COMMANDES (  
    COMMANDE_ID NUMBER PRIMARY KEY,  
    CLIENT_ID NUMBER FOREIGN KEY  
        REFERENCES CLIENTS(CLIENT_ID),  
    PROD_ID NUMBER FOREIGN KEY  
        REFERENCES PRODUITS(PROD_ID),  
    PROMO_ID NUMBER FOREIGN KEY  
        REFERENCES PROMOTIONS(PROMO_ID),  
    COMMANDE_DATE DATE NOT NULL,  
    COMMANDE_PRIX NUMBER NOT NULL,  
    COMMANDE_QUANTITE NUMBER NOT NULL  
);
```

b) ALTER TABLE COMMANDES
ADD CONSTRAINT C_POST_2015
(CHECK (YEAR(COMMANDE_DATE) > 2015))

CREATE TRIGGER MAX_1_M_Paris
BEFORE INSERT OR UPDATE ON COMMANDES
FOR EACH ROW

DECLARE

e_max-1m-Paris EXCEPTION;

PRAGMA EXCEPTION-INITIALIZE(e_max-1m-Paris,
-20001);

v-total-Paris NUMBER;

BEGIN

SELECT SUM(C.COMMANDE_PRIX) INTO

v-total-Paris

FROM COMMANDES C

```
INNER JOIN CLIENTS c1  
ON c1.CLIENT_ID = c2.CLIENT_ID  
WHERE c1.VILLE = 'Paris';  
IF v-total_paris > 1000000 THEN  
    RAISE e-max_in_paris;  
END;
```

END;
/

II)
a)

```
SELECT p.PROD_ID  
FROM PRODUITS p LEFT JOIN COMMANDES c  
ON c.PROD_ID = p.PROD_ID  
GROUP BY p.PROD_ID  
HAVING COUNT(p.PROD_ID) = 0
```

b)

```
SELECT c.CLIENT_ID, COUNT(*)  
FROM CLIENTS c INNER JOIN COMMANDES cd  
ON c.CLIENT_ID = cd.CLIENT_ID  
GROUP BY c.CLIENT_ID
```

```
c) SELECT c.CLIENT NOM
FROM CLIENTS c INNER JOIN COMMANTES c1
ON c.CLIENT_ID = c1.CLIENT_ID
INNER JOIN PRODUITS p
ON p.PROD_ID = c1.PROD_ID
WHERE p.PROD_CATEGORIE = 'HIFI'
GROUP BY c.CLIENT_ID, c.CLIENT_NOM
HAVING COUNT(DISTINCT p.PROD_ID) = (
    SELECT COUNT(*)
    FROM PRODUITS
    WHERE PROD_CATEGORIE = 'HIFI'
);
```

d) Ville et le nombre de limite
crédit différentes pour la ville
dans les limites de crédit supérieures à
1000 et dont les limites de crédits
moyennes de la ville est entre 5000 et
6000.

1VJ

a)

```
CREATE VIEW best_products
AS
SELECT p.PROD-NOM,
FROM PRODUCTS p INNER JOIN
COMMANDES c
ON c.PROD-ID = p.PROD-ID
GROUP BY c.PROD-ID, p.PROD-NOM
HAVING SUM(PROD-QUANTITE) > 2 * (
    SELECT SUM(PROD-QUANTITE)
    FROM COMMANDES
    GROUP BY PROD-ID
);
```

```
6) SELECT p. PROD-NOM  
FROM PRODUITS p INNER JOIN  
COMMANDES cd ON cd. PROD_ID = p. PROD_ID  
INNER JOIN CLIENTS c  
ON c. CLIENT_ID = cd. CLIENT_ID  
WHERE c. CLIENT-VILLE = 'Paris'  
AND p. PROD-NOM IN best-produits  
GROUP BY p. PROD-NOM, p. PROD_ID  
ORDER BY COUNT(*) DESC  
LIMIT 5;
```



```
SELECT p. PROD-NOM  
FROM PRODUCTS p LEFT JOIN  
COMMANDES cd ON cd. PROD_ID = p. PROD_ID  
LEFT JOIN CLIENTS c  
ON c. CLIENT_ID = cd. CLIENT_ID  
WHERE c. CLIENT-VILLE = 'Lille'  
AND p. PROD-NOM IN best-products  
GROUP BY p. PROD-NOM, p. PROD_ID  
HAVING COUNT(p. prod_ID) = 0.
```

V]

b)

```
CREATE VIEW client_solde_inf_1000  
AS
```

```
SELECT (*) FROM CLIENTS  
WHERE CLIENT_VILLE = 'Paris'  
AND CLIENT_SOLDE < 1000;
```

```
GRANT SELECT, UPDATE ON client_solde_inf_1000 TO timitin.
```

c) Il l'a modifié au dessus de 1000 et donc n'est plus accès.

d)

```
CREATE FUNCTION FAN-PRODUIT  
(IN prod_id NUMBER)
```

```
IS
```

```
DECLARE
```

```
    v_client VARCHAR(250);
```

```
BEGIN
```

```
    SELECT CLIENT_Nom INTO v_client  
    FROM COMMANDES c1 INNER JOIN CLIENTS c2  
    ON c1.CLIENT_ID = c2.CLIENT_ID
```

```
    WHERE c1.PROD_ID = prod_id
```

```
    GROUP BY c1.CLIENT_ID, c1.CLIENT_Nom  
    HAVING SUM(c1.COMMANDE-QUANTITE) = (
```

```
        SELECT MAX(qte)
```

```
        FROM (SELECT SUM(COMMANDE-QUANTITE)
```

```
            FROM COMMANDE WHERE PROD_ID = prod_id  
            GROUP BY CLIENT_ID))
```

```
    RETURN v_client;  
END;  
/
```

e) CREATE TRIGGER promo_date
INSERT OR UPDATE ON PROMOTIONS
FOR EACH ROW

IS

DECLARE

e_date-promo EXCEPTION;

PRAGMA_EXCEPTION_INIT(e_date-promo, 7002)

BEGIN

IF :new.promo_dateFin <

:new.promo_dateDebut THEN

RAISE e_date-promo

END;

END;
/