

```

-- 1. Les numéros des clients (sans répétition) qui ont placé au moins une commande
SELECT DISTINCT noClient
FROM Commande;

-- 2. Le numéro et la description des articles dont le numéro est entre 20 et 80 (inclusivement) et le
prix est 10.99 ou 25.99
SELECT noArticle, description
FROM Article
WHERE noArticle BETWEEN 20 AND 80 AND ( prixUnitaire = 10.99 OR prixUnitaire=25.99);

-- 3. Le numéro et la description des articles dont la description débute par la lettre C ou contient la
chaîne 'bl'
SELECT noArticle, description
FROM Article
WHERE (description LIKE 'C%') OR (description LIKE '%bl%');

-- 4. Le numéro et le nom des clients qui ont placé une commande le 9 juillet 2000
SELECT noClient, nomclient
FROM Client NATURAL JOIN Commande
WHERE datecommande = '09/07/2000';

-- 5. Les noms des clients, numéros de commande, date de commande et noArticle pour
-- les articles livrés le 4 juin 2000 dont la quantité livrée est supérieure à 1
SELECT nomClient, noCommande, dateCommande, noArticle
FROM Client
NATURAL JOIN Commande
NATURAL JOIN LigneCommande
NATURAL JOIN DetailLivraison
NATURAL JOIN Livraison
WHERE dateLivraison = '04/06/2000' AND quantiteLivree >1;

-- 6. La liste des dates du mois de juin 2000 pour lesquelles il y a au moins une livraison ou une
commande.
--Les résultats sont produits en une colonne nommée DateÉvénement.
SELECT DISTINCT *
FROM
    (SELECT datecommande AS DateEvenement
    FROM Commande
    WHERE datecommande LIKE '%06/2000%')
    union
    (SELECT datelivraison AS DateEvenement
    FROM Livraison
    WHERE datelivraison LIKE '%06/2000%');

-- 7. Les noArticle et la quantité totale livrée de l'article incluant les articles
--dont la quantité totale livrée est égale à 0.
SELECT noArticle, SUM(COALESCE(quantiteLivree,0)) AS "QUANTITE LIVREE"
FROM DetailLivraison NATURAL RIGHT OUTER JOIN Article
GROUP BY noArticle;

-- 8. Les noArticle et la quantité totale livrée de l'article pour les articles dont le
--prix est inférieur à $20 et dont la quantité totale livrée est inférieure à 5
SELECT noArticle, QUANTITELIVREE AS "QUANTITE LIVREE"
FROM (SELECT noArticle, QUANTITELIVREE
FROM (SELECT noArticle, SUM(COALESCE(quantiteLivree,0)) AS QUANTITELIVREE
FROM DetailLivraison NATURAL RIGHT OUTER JOIN Article
GROUP BY noArticle)
WHERE QUANTITELIVREE<5) NATURAL JOIN Article
WHERE prixUnitaire<20
ORDER BY noArticle;

```

```

-- 9. Le noLivraison, noCommande, noArticle, la date de la commande, la quantité commandée,
--la date de la livraison, la quantité livrée et le nombre de jours écoulés entre la commande et la
--livraison dans le cas où ce nombre a dépassé 2 jours et le nombre de jours écoulés depuis la commande
--jusqu'à aujourd'hui est supérieur à 100
SELECT noLivraison, noCommande, noArticle, datecommande, quantite, datelivraison, quantiteLivree,
(dateLivraison-dateCommande) AS "NOMBRE JOURS ECOULES"
FROM Commande
NATURAL JOIN LigneCommande
NATURAL JOIN DetailLivraison
NATURAL JOIN Livraison
WHERE (dateLivraison-dateCommande > 2)
AND (SYSDATE-datecommande>100);

-- 10. La liste des Articles triée en ordre décroissant de prix et pour chacun des prix en
--ordre croissant de numéro
SELECT noArticle,description,prixUnitaire ,quantiteEnStock
FROM Article
ORDER BY prixUnitaire DESC, noArticle ASC;

-- 11. Le nombre d'articles dont le prix est supérieur à 25 et le nombre d'articles
--dont le prix est inférieur à 15 (en deux colonnes)
SELECT *
FROM (SELECT Count(prixUnitaire) AS "NOMBREPLUSCHERQUE25"
FROM Article
WHERE prixUnitaire>25
),( SELECT Count(prixUnitaire) AS "NOMBREM0INSCHERQUE15"
FROM Article
WHERE prixUnitaire<15 );

-- 12. Les noCommande des commandes qui n'ont aucune livraison correspondante
SELECT noCommande
FROM Commande NATURAL LEFT OUTER JOIN DetailLivraison
WHERE noLivraison IS NULL;

-- 13. En deux colonnes, les paires de numéros de commandes (différentes) qui sont faites à la même date
ainsi que la date de commande. Il faut éviter de répéter deux fois la même paire.
SELECT DISTINCT c1.noCommande, c2.noCommande, c2.dateCommande
FROM Commande c1 , Commande c2
WHERE c1.dateCommande = c2.dateCommande AND c1.noCommande != c2.noCommande AND c1.noCommande <
c2.noCommande;

-- 14. Le montant total commandé pour chaque paire (dateCommande, noArticle) dans les cas où le montant
total dépasse 50$.
SELECT t1.dateCommande, t1.noArticle, SUM(t1.MONTANTTOTALCOMMANDE) as "MONTANT TOTAL COMMANDE"
FROM(SELECT dateCommande, noArticle,
(quantite * prixUnitaire) AS MONTANTTOTALCOMMANDE
FROM Commande
NATURAL JOIN LigneCommande
NATURAL JOIN Article
WHERE (quantite*prixUnitaire)>50) t1
GROUP BY t1.noArticle, t1.datecommande;

-- 15. Les noArticle des articles commandés dans toutes et chacune des commandes du client 20
SELECT noArticle
FROM Article
WHERE NOT EXISTS
(SELECT noCommande
FROM Commande
WHERE noClient = 20 AND NOT EXISTS
(SELECT *
FROM LigneCommande
WHERE noArticle = Article.noArticle
AND noCommande = Commande.noCommande));

```