

1.

```
SELECT nom, prenom, fonction, salaire  
FROM Employes  
WHERE salaire BETWEEN 2500 AND 3500;
```

2. Afficher le nom du produit, le nom du fournisseur, le nom de la catégorie et les quantités de produits qui ne sont pas dans les catégories 1, 3, 5 et 7 :

```
SELECT p.nom_produit, f.nom_fournisseur, c.nom_categorie, p.quantite  
FROM Produits p  
JOIN Fournisseurs f ON p.id_fournisseur = f.id_fournisseur  
JOIN Categories c ON p.id_categorie = c.id_categorie  
WHERE c.id_categorie NOT IN (1, 3, 5, 7);
```

3. Afficher le nom du produit, le nom du fournisseur, le nom de la catégorie et les quantités des produits qui ont un numéro de fournisseur entre 1 et 3 ou un code catégorie entre 1 et 3:

```
SELECT p.nom_produit, f.nom_fournisseur, c.nom_categorie, p.quantite  
FROM Produits p  
JOIN Fournisseurs f ON p.id_fournisseur = f.id_fournisseur  
JOIN Categories c ON p.id_categorie = c.id_categorie  
WHERE (p.id_fournisseur BETWEEN 1 AND 3 OR p.id_categorie BETWEEN 1 AND 3)
```

4. Afficher le nom des employés qui ont effectué au moins une vente pour un client parisien :

```
SELECT DISTINCT e.nom  
FROM Employes e  
JOIN Ventes v ON e.id_employe = v.id_employe  
JOIN Clients c ON v.id_client = c.id_client  
WHERE c.ville = 'Paris';
```

5. Afficher le nom des produits et le nom des fournisseurs pour les produits des catégories 1, 4 et 7 :

```
SELECT p.nom_produit, f.nom_fournisseur  
FROM Produits p  
JOIN Fournisseurs f ON p.id_fournisseur = f.id_fournisseur  
WHERE p.id_categorie IN (1, 4, 7);
```

6. Afficher le nom des employés ainsi que le nom de leur supérieur hiérarchique :

```
SELECT e.nom AS employe, s.nom AS superieur  
FROM Employes e  
JOIN Hierarchie h ON e.id_employe = h.id_employe  
JOIN Employes s ON h.id_superieur = s.id_employe;
```

7. Affichez tous les employés (nom, prénom) avec leur supérieur hiérarchique (nom du supérieur). Affichez aussi les employés qui n'ont pas de supérieur. Utilisez LEFT JOIN.

```
SELECT
    e.nom AS nom_employe,
    e.prenom AS prenom_employe,
    s.nom AS nom_superieur
FROM Employes e
LEFT JOIN Hierarchie h ON e.id_employe = h.id_employe
LEFT JOIN Employes s ON h.id_superieur = s.id_employe;
```

Affichez tous les employés qui ont un supérieur (INNER JOIN).

```
SELECT
    e.nom AS nom_employe,
    s.nom AS nom_superieur
FROM Employes e
JOIN Hierarchie h ON e.id_employe = h.id_employe
JOIN Employes s ON h.id_superieur = s.id_employe;
```

8. Affichez tous les supérieurs même s'ils ne supervisent personne (RIGHT JOIN sur Hierarchie).

```
SELECT
    s.nom AS nom_superieur,
    e.nom AS nom_employe
FROM Employes e
RIGHT JOIN Hierarchie h ON e.id_employe = h.id_employe
RIGHT JOIN Employes s ON h.id_superieur = s.id_employe;
```