1. List employee IDs and hire dates for employees hired after '2021-01-01'

```
=  SELECT Employee_Id, Hire_Date
  FROM Employee
  WHERE Hire_Date > '2021-01-01';
```

2. Retrieve the first name, last name, and annual salary (use an alias) for employees earning more than 8000

```
  =SELECT First_Name, Last_Name, Salary AS Annual_Salary
  FROM Employee
  WHERE Salary > 8000;
```

3. Find the employee ID, concatenate first name and last name, add a space between first name and last name, and show the column header as "name":

```
  SELECT Employee_Id, CONCAT(First_Name, ' ', Last_Name) AS Name
  FROM Employee;
```

4. Retrieve employees whose Job_Title contains the word 'Manager' (Use INSTR()):

```
  SELECT E.Employee_Id, E.First_Name, E.Last_Name, J.Job_Title
  FROM Employee E
  JOIN Job J ON E.Job_Id = J.Job_Id
  WHERE INSTR(J.Job_Title, 'Manager') > 0;
```

5. List employees working in departments 10 or 20, ascendingly sorted by hire date:

```
 SELECT Employee_Id, First_Name, Last_Name, Hire_Date
 FROM Employee
 WHERE Department_Id IN (10, 20)
 ORDER BY Hire_Date ASC;
```

6. Retrieve departments where no employees are assigned:

```
 SELECT Department_Id, Department_Name
 FROM Department
 WHERE Department_Id NOT IN (SELECT DISTINCT Department_Id FROM Employee);
```

7. Retrieve employees who have no manager:

```
SELECT Employee_Id, First_Name, Last_Name
FROM Employee
WHERE Manager_Id IS NULL;
```

8. List employees hired on the same date as another employee:

```
SELECT E1.Employee_Id, E1.First_Name, E1.Last_Name, E1.Hire_Date
FROM Employee E1
WHERE EXISTS (
SELECT 1
FROM Employee E2
WHERE E1.Hire_Date = E2.Hire_Date AND E1.Employee_Id != E2.Employee_Id
);
```

9. Retrieve employees whose salary is greater than the company's average salary:

```
SELECT Employee_Id, First_Name, Last_Name, Salary
FROM Employee
WHERE Salary > (SELECT AVG(Salary) FROM Employee);
```

10. Find employees with no commission and a non-null manager ID:

```
SELECT Employee_Id, First_Name, Last_Name, Salary
FROM Employee
WHERE Commission_pct IS NULL AND Manager_Id IS NOT NULL;
```

11. Find the employee ID, first name, and salary for all employees with salaries between 7000 and 10000. The result of the query must be sorted in descending order of salary and ascending order of first name:

```
SELECT Employee_Id, First_Name, Salary
FROM Employee
WHERE Salary BETWEEN 7000 AND 10000
ORDER BY Salary DESC, First_Name ASC;
```

12. Retrieve employees whose hire date is within the top 3 most recent hire dates:

```
 SELECT e.Employee_Id, e.First_Name, e.Last_Name, e.Hire_Date
 FROM Employee e
  JOIN (
   SELECT DISTINCT Hire_Date
   FROM Employee
   ORDER BY Hire_Date DESC
   LIMIT 3
) recent_hires ON e.Hire_Date = recent_hires.Hire_Date;
```