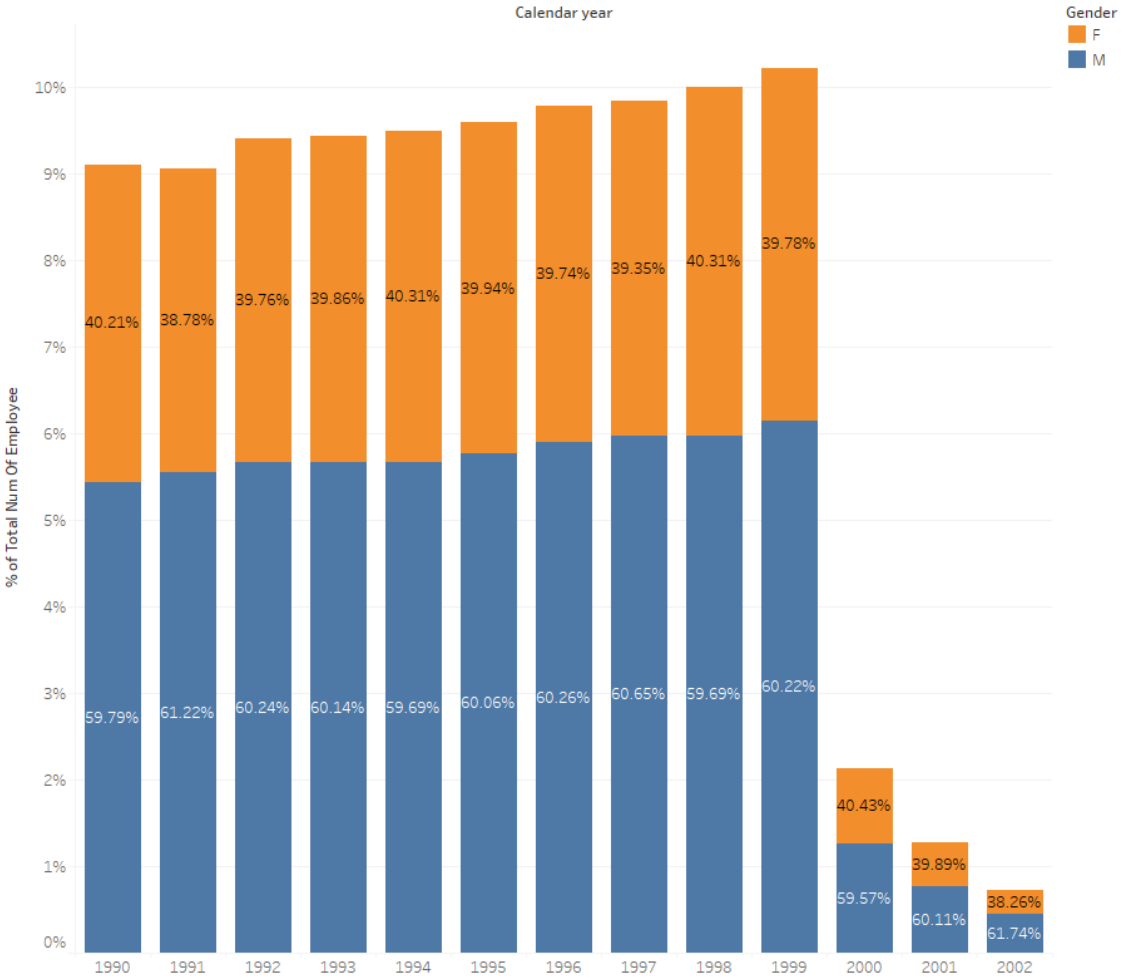
SQL + Tableau examples

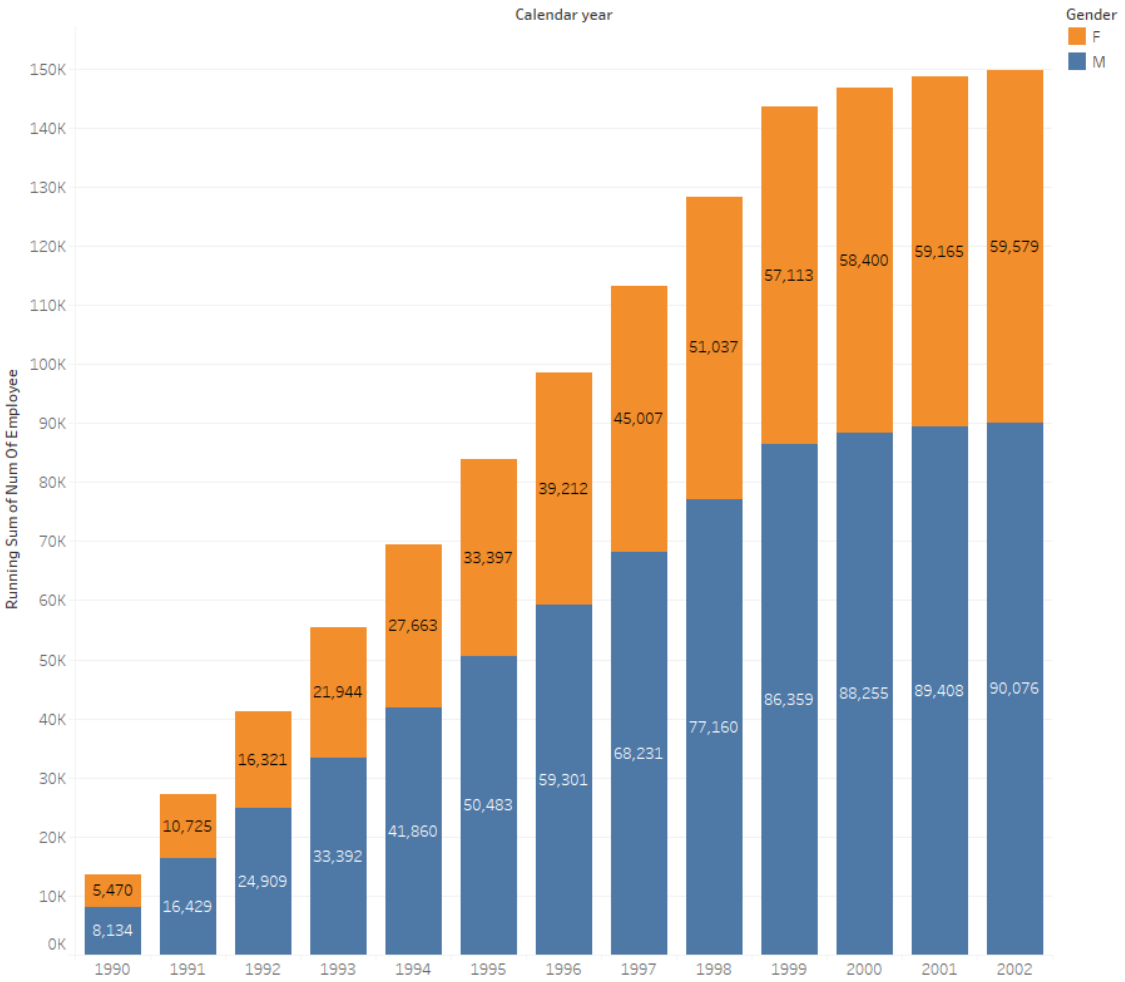
Example 1: number of employees in both genders after 1990

```
SELECT
        YEAR(d.from_date) AS Calendar_year,
        e.gender,
   COUNT(e.emp_no) as num_of_employee
FROM
        t_employees e
   JOIN
   t_dept_emp d ON d.emp_no = e.emp_no
GROUP BY calendar_year, e.gender
HAVING  calendar_year >= 1990;
```

Separate view:



Accumulative view:

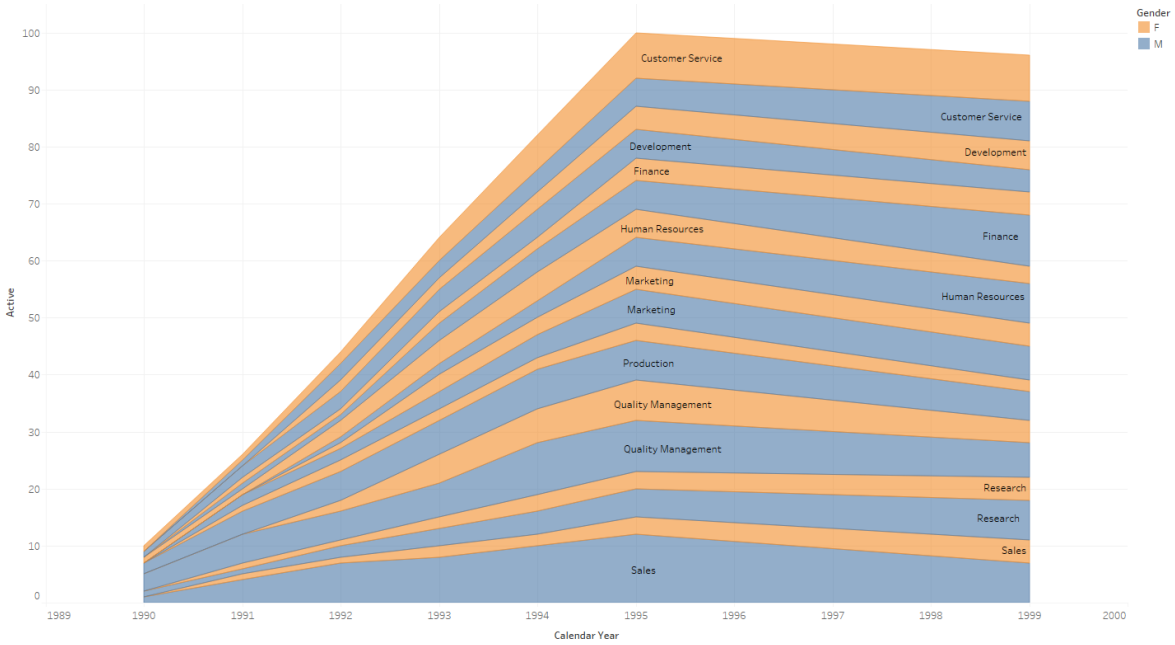


Example 2:

compare the number of male managers with female managers from different departments for each year, starting from 1990.

```
SELECT
    d.dept_name,
    ee.gender,
    dm.emp_no,
    dm.from_date,
    dm.to_date,
```

```
    e.calendar_year,
    CASE
        WHEN e.calendar_year >= YEAR(dm.from_date)
                                AND e.calendar_year <= YEAR(dm.to_date)
        THEN 1
        ELSE 0
    END AS active
FROM
        (SELECT
                YEAR(hire_date) as calendar_year
        FROM
                t_employees
        GROUP BY calendar_year
    ) e
        CROSS JOIN
    t_dept_manager dm
    JOIN
    t_employees ee ON dm.emp_no = ee.emp_no
    JOIN
    t_departments d
    ON
                dm.dept_no = d.dept_no
ORDER BY dm.emp_no AND e.calendar_year;
```



Example 3:

Compare the average salary of female and male employees in the entire company until 2002, and add a filter allowing you to see that per each department.

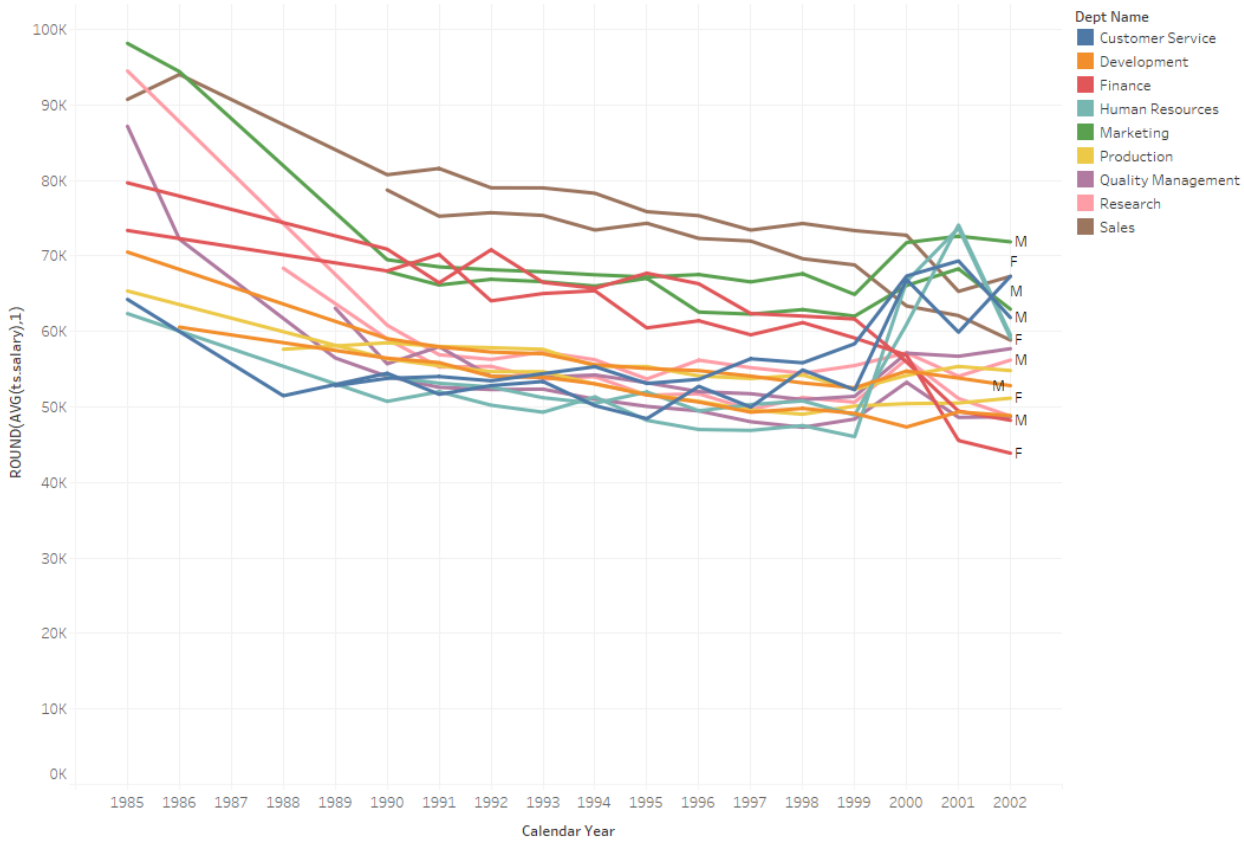
```
SELECT
    te.gender,
        ROUND(AVG(ts.salary),1),
    td.dept_name,
    YEAR(tde.from_date) as calendar_year
FROM
        t_employees te
        JOIN
    t_salaries ts ON te.emp_no = ts.emp_no
    JOIN
    t_dept_emp tde ON ts.emp_no = tde.emp_no
    JOIN
```

t_departments td ON td.dept_no = tde.dept_no

GROUP BY td.dept_name, te.gender,calendar_year

HAVING calendar_year <= 2002

ORDER BY tde.dept_no;



Example 3:

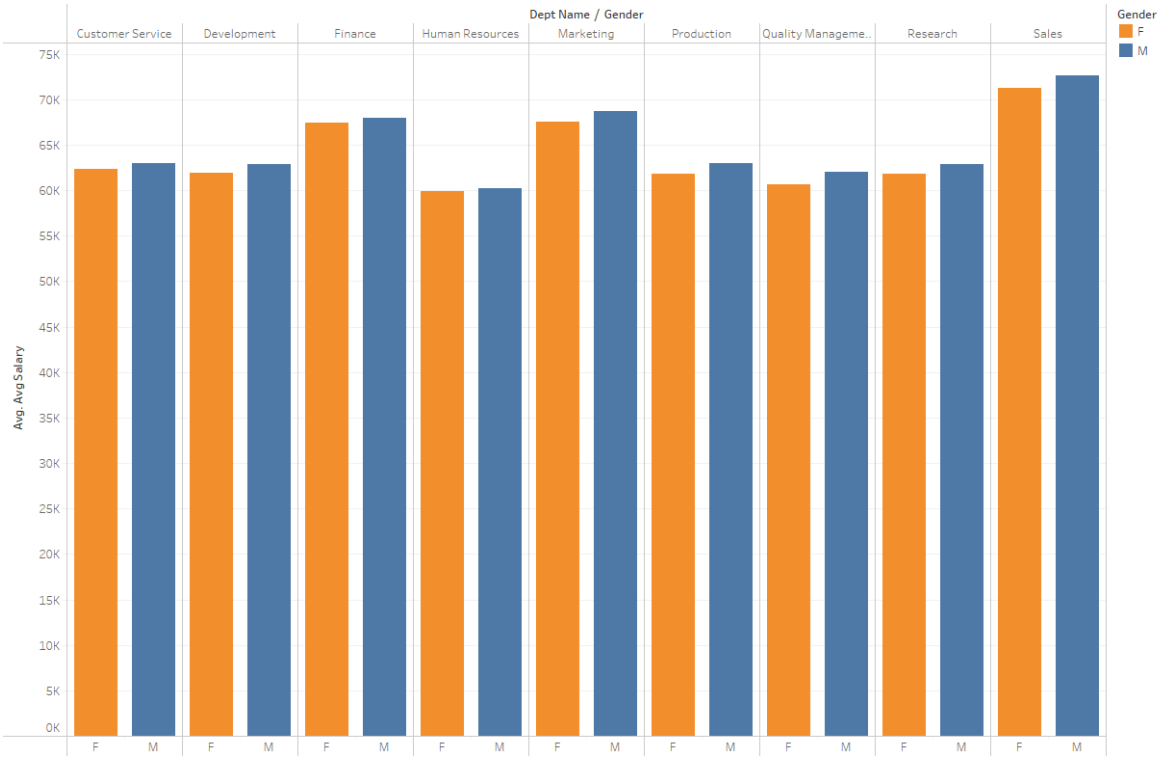
Create an SQL stored procedure that will allow you to obtain the average male and female salary per department within a certain salary range. Let this range be defined by two values the user can insert when calling the procedure (using double-bar charts).

DROP PROCEDURE IF EXISTS filter_salary;

```
DELIMITER $$
CREATE PROCEDURE filter_salary (IN min FLOAT, IN max FLOAT)
BEGIN
SELECT
    e.gender, d.dept_name, AVG(s.salary) as avg_salary
FROM
    t_salaries s
        JOIN
    t_employees e ON s.emp_no = e.emp_no
        JOIN
    t_dept_emp de ON de.emp_no = e.emp_no
        JOIN
    t_departments d ON d.dept_no = de.dept_no
    WHERE s.salary BETWEEN min AND max
GROUP BY d.dept_no, e.gender;
END$$

DELIMITER ;

CALL filter_salary(50000, 90000);
```



Dashboard view: