

# Employee database Sample

Current schema by Giuseppe Maxia

Data conversion from XML to relational by Patrick Crews

This work is licensed under the

Creative Commons Attribution-Share Alike 3.0 Unported License.

To view a copy of this license, visit

<http://creativecommons.org/licenses/by-sa/3.0/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

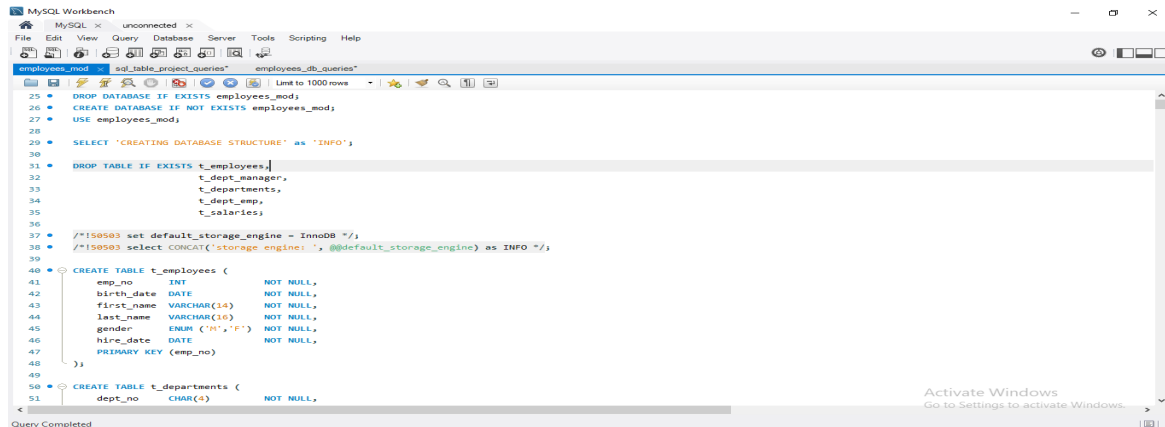
## DISCLAIMER

To the best of my knowledge, this data is fabricated, and

It does not correspond to real people.

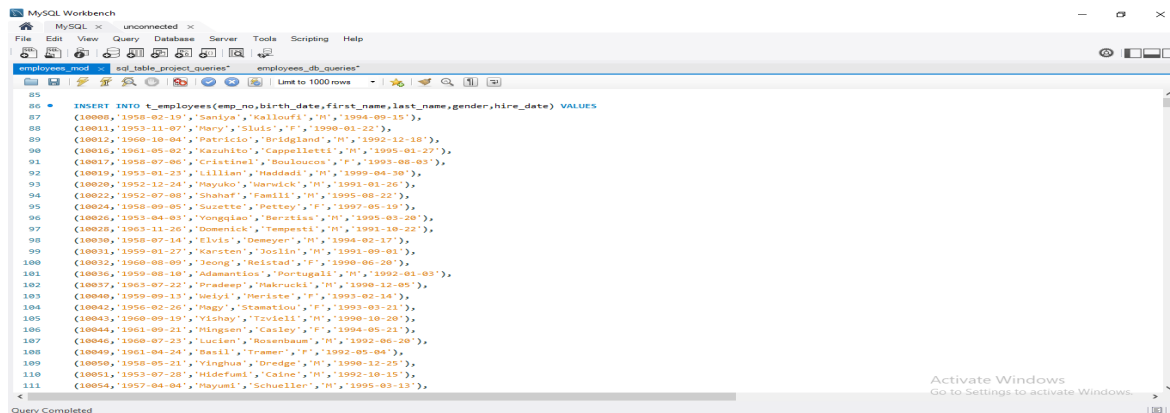
Any similarity to existing people is purely coincidental.

## Creating the employee\_mod Database



```
MySQL Workbench
MySQL x unconnected x
File Edit View Query Database Server Tools Scripting Help
employees_mod sql_table_project_queries employees_db_queries
25 DROP DATABASE IF EXISTS employee_mod;
26 CREATE DATABASE IF NOT EXISTS employee_mod;
27 USE employee_mod;
28
29 SELECT 'CREATING DATABASE STRUCTURE' as 'INFO';
30
31 DROP TABLE IF EXISTS t_employees,
32 t_dept_manager,
33 t_departments,
34 t_dept_emp,
35 t_salaries;
36
37 /*!50503 set default_storage_engine = InnoDB */
38 /*!50503 select CONCAT('storage engine: ', @@default_storage_engine) as INFO */
39
40 CREATE TABLE t_employees (
41   emp_no INT NOT NULL,
42   birth_date DATE NOT NULL,
43   first_name VARCHAR(14) NOT NULL,
44   last_name VARCHAR(16) NOT NULL,
45   gender ENUM ('M','F') NOT NULL,
46   hire_date DATE NOT NULL,
47   PRIMARY KEY (emp_no)
48 );
49
50 CREATE TABLE t_departments (
51   dept_no CHAR(4) NOT NULL,
```

## Inserting Data into the Database



```
MySQL Workbench
MySQL x unconnected x
File Edit View Query Database Server Tools Scripting Help
employees_mod sql_table_project_queries employees_db_queries
85 INSERT INTO t_employees(emp_no,birth_date,first_name,last_name,gender,hire_date) VALUES
86 (10000,'1958-02-19','Saniya','Kalloufi','M','1994-09-15'),
87 (10011,'1953-11-07','Harry','Sluis','F','1998-01-22'),
88 (10012,'1968-10-04','Patricio','Bridgland','M','1992-12-18'),
89 (10016,'1961-05-02','Kazuhiro','Cappelletti','M','1995-01-27'),
90 (10017,'1958-07-06','Cristinel','Bouloucos','F','1993-08-05'),
91 (10019,'1953-01-23','Lillian','Heddes','M','1990-04-30'),
92 (10020,'1952-12-24','Hayuko','Warwick','M','1991-01-26'),
93 (10022,'1952-07-08','Shahar','Famili','M','1995-08-22'),
94 (10024,'1958-09-09','Suzette','Petty','F','1997-05-19'),
95 (10026,'1953-04-03','Yongqiao','Berstias','M','1995-03-20'),
96 (10028,'1963-11-26','Domenick','Temeesti','M','1991-10-22'),
97 (10030,'1958-07-14','Elvis','Deemeyer','M','1994-02-17'),
98 (10031,'1959-01-27','Karsten','Joslin','M','1991-09-01'),
99 (10032,'1960-08-09','Jeong','Reistad','F','1990-06-20'),
100 (10036,'1959-08-10','Adamantios','Portugali','M','1992-01-03'),
101 (10037,'1963-07-22','Pradeep','Makrunki','M','1998-12-00'),
102 (10040,'1959-09-13','Weiyl','Meriste','F','1993-02-14'),
103 (10042,'1956-02-26','Hagy','Stamatlou','F','1993-03-21'),
104 (10043,'1960-09-19','Ylchay','Tzevili','M','1990-10-20'),
105 (10044,'1961-09-21','Ringsen','Casley','F','1994-05-21'),
106 (10046,'1968-07-23','Lucien','Rosenbaum','M','1992-06-20'),
107 (10049,'1961-04-24','Basil','Tramer','M','1992-05-04'),
108 (10050,'1958-05-21','Yinghua','Dredge','M','1990-12-25'),
109 (10051,'1953-07-28','Hidefumi','Caine','M','1992-10-15'),
110 (10054,'1957-04-04','Hayumi','Schueller','M','1995-03-13'),
111
```

## Exploring the content of each table

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL code:

```
USE employees_mod;

SELECT * FROM t_employees;

SELECT * FROM t_dept_emp;

SELECT * FROM t_departments;

SELECT * FROM t_dept_manager;

SELECT * FROM t_salaries;
```

The results grid displays the first 13 rows of the `t_employees` table:

emp_no	birth_date	first_name	last_name	gender	hire_date
10008	1958-02-19	Gerya	Kabouf	M	1994-09-15
10011	1953-11-07	Mary	Skie	F	1990-01-22
10012	1960-05-04	Pietro	Bridgend	M	1992-11-18
10016	1961-05-02	Kazuhiro	Cappelletti	M	1995-01-27
10017	1958-07-06	Cristinel	Boudoucos	F	1993-08-03
10018	1953-01-23	Lillian	Heddadi	M	1999-04-30
10020	1952-12-24	Mayuko	Warwick	M	1991-01-26
10022	1952-07-08	Shahaf	Fendi	M	1995-08-22
10024	1958-09-05	Suzette	Petty	F	1997-05-19
10026	1953-04-03	Yongsoo	Bertias	M	1995-03-20
10028	1963-11-26	Domenick	Tempesti	M	1991-10-22
10030	1958-07-14	Elvis	Demeyer	M	1994-02-17
10031	1959-01-27	Karsten	Jahn	M	1991-09-01
10032	1960-08-09	Jeong	Reisstad	F	1990-06-20

**TASK1:** Create a visualization that provides a breakdown between the male and female employees working in the company each year, starting from 1990.

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL code:

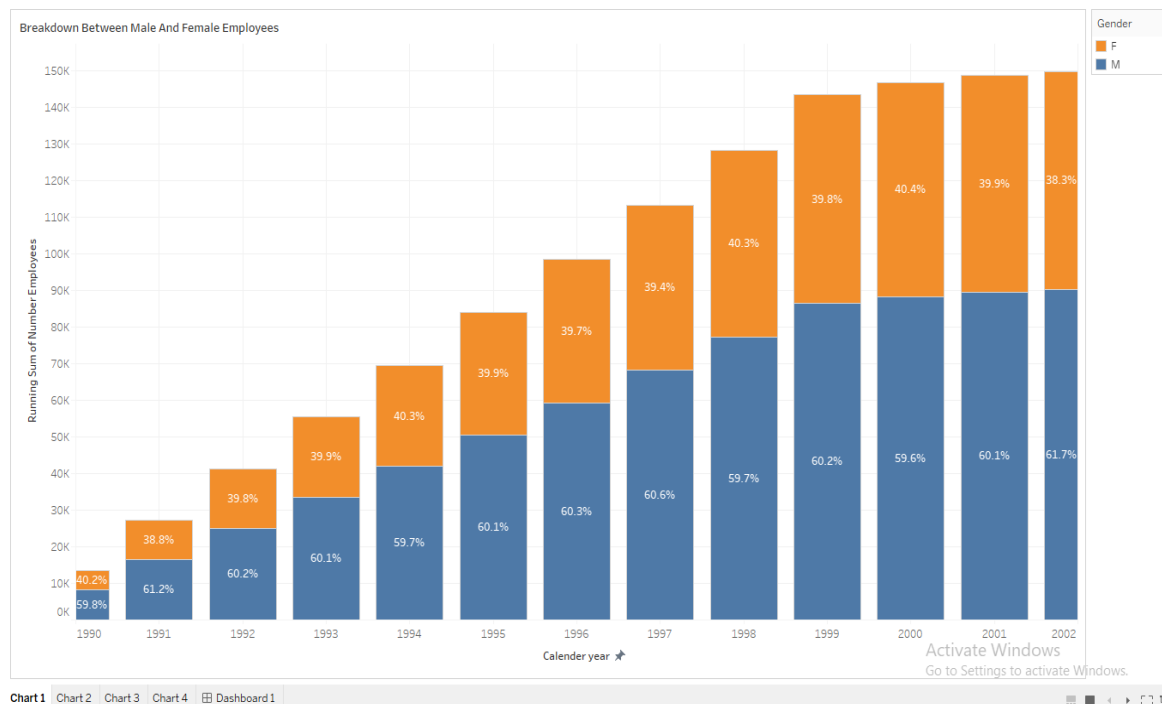
```
# TASK 1
/* Create a visualization that provides a breakdown between the male
and female employees working in the company each year, starting from 1990. */

# Extracting the data based on the above business question.

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

The results grid displays the first 13 rows of the query results:

Calendar_year	gender	number_employees
1990	M	4929
1990	F	5470
1991	M	5480
1991	F	5423
1992	M	5499
1992	F	5430
1993	M	5523
1993	F	5533
1994	M	5533
1994	F	5533
1995	M	5533
1995	F	5533
1996	M	5533
1996	F	5533
1997	M	5533
1997	F	5533
1998	M	5533
1998	F	5533
1999	M	5533
1999	F	5533
2000	M	5533
2000	F	5533
2001	M	5533
2001	F	5533
2002	M	5533
2002	F	5533



**TASK2:** Compare the number of male managers to the number of female managers from different departments for each year, starting from 1990.

MySQL Workbench

```

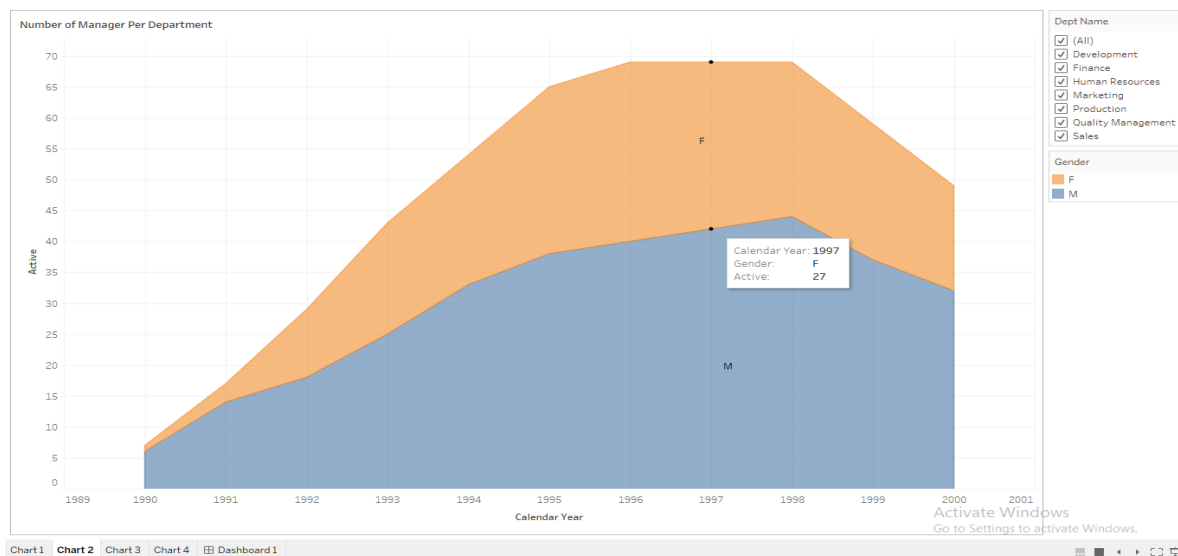
33
34
35 /* Compare the number of male managers to the number of
36 female managers from different departments for each year, starting from 1990.
37 */
38
39 SELECT
40     d.dept_name,
41     ee.gender,
42     ee.emp_no,
43     de.from_date,
44     de.to_date,
45     e.calendar_year,
46     CASE

```

dept_name	gender	emp_no	from_date	to_date	calendar_year	active
Marketing	M	110022	1995-12-30	1998-12-29	1990	0
Marketing	M	110022	1995-12-30	1998-12-29	1991	0
Marketing	M	110022	1995-12-30	1998-12-29	1992	0
Marketing	M	110022	1995-12-30	1998-12-29	1993	0
Marketing	M	110022	1995-12-30	1998-12-29	1994	0
Marketing	M	110022	1995-12-30	1998-12-29	1995	1
Marketing	M	110022	1995-12-30	1998-12-29	1996	1
Marketing	M	110022	1995-12-30	1998-12-29	1997	1
Marketing	M	110022	1995-12-30	1998-12-29	1998	1
Marketing	M	110022	1995-12-30	1998-12-29	1999	0
Marketing	M	110022	1995-12-30	1998-12-29	2000	0
Marketing	M	110039	1997-04-09	9999-01-01	1990	0
Marketing	M	110039	1997-04-09	9999-01-01	1991	0
Marketing	M	110039	1997-04-09	9999-01-01	1992	0

Result Grid | Filter Rows | Export | Wrap Cell Contents | Fetch rows: 1000

Query Completed



**TASK3:** Compare the average salary of female versus male employees in the entire company until the year 2002, and add a filter allowing you to see that per each department.

MySQL Workbench

```

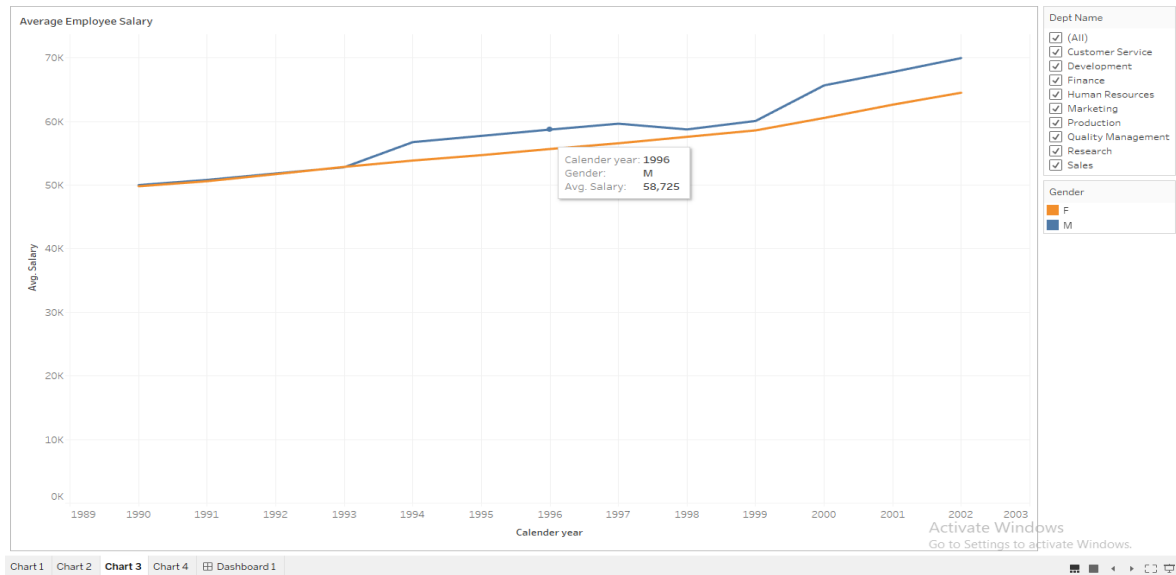
69
70 SELECT
71     e.gender,
72     d.dept_name,
73     ROUND(AVG(s.salary), 2) AS Salary,
74     YEAR(s.from_date) AS Calendar_year
75 FROM
76     t_salaries s
77 JOIN
78     t_employees e ON s.emp_no = e.emp_no
79 JOIN
80     t_dept_emp de ON e.emp_no = de.emp_no

```

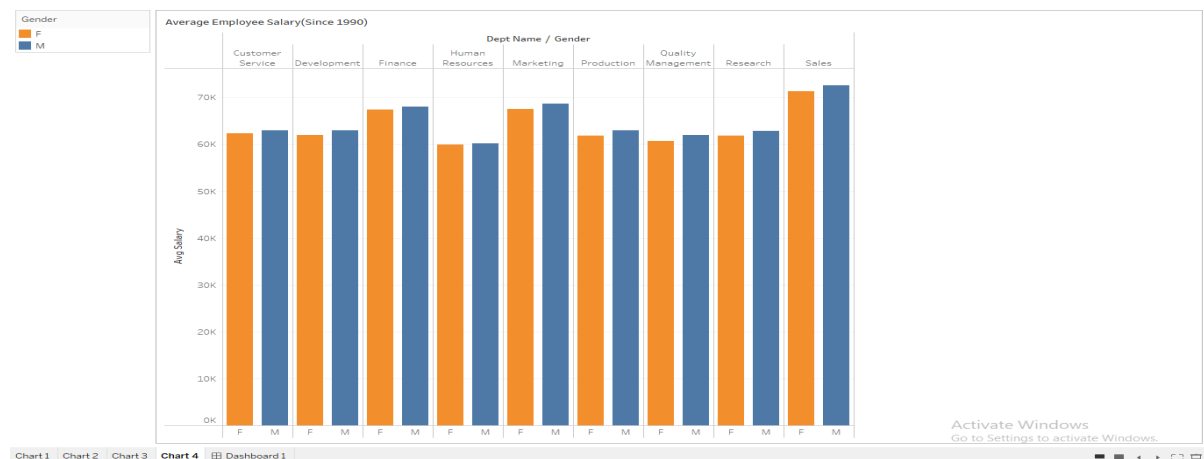
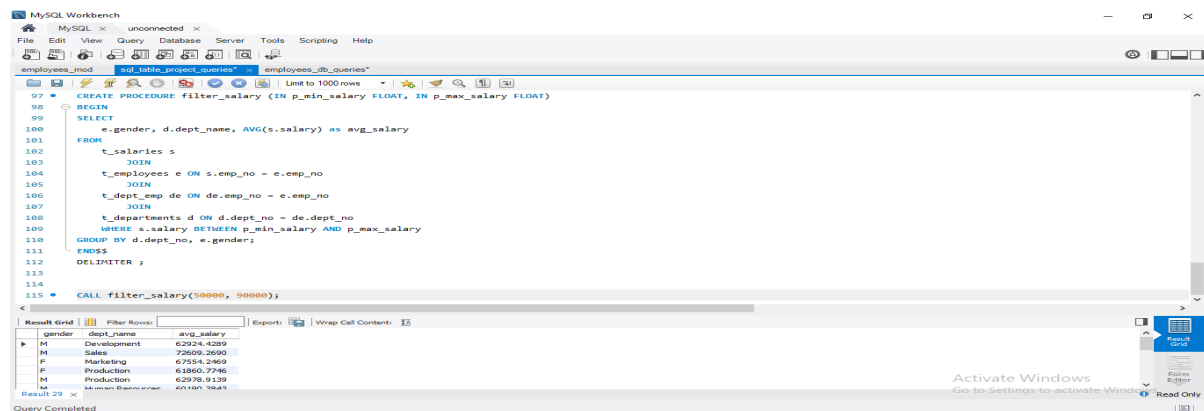
dept_name	gender	emp_no	from_date	to_date	calendar_year	active
Marketing	M	110022	1995-12-30	1998-12-29	1990	0
Marketing	M	110022	1995-12-30	1998-12-29	1991	0
Marketing	M	110022	1995-12-30	1998-12-29	1992	0
Marketing	M	110022	1995-12-30	1998-12-29	1993	0
Marketing	M	110022	1995-12-30	1998-12-29	1994	0
Marketing	M	110022	1995-12-30	1998-12-29	1995	1
Marketing	M	110022	1995-12-30	1998-12-29	1996	1
Marketing	M	110022	1995-12-30	1998-12-29	1997	1
Marketing	M	110022	1995-12-30	1998-12-29	1998	1
Marketing	M	110022	1995-12-30	1998-12-29	1999	0
Marketing	M	110022	1995-12-30	1998-12-29	2000	0
Marketing	M	110039	1997-04-09	9999-01-01	1990	0
Marketing	M	110039	1997-04-09	9999-01-01	1991	0
Marketing	M	110039	1997-04-09	9999-01-01	1992	0

Result Grid | Filter Rows | Export | Wrap Cell Contents | Fetch rows: 1000

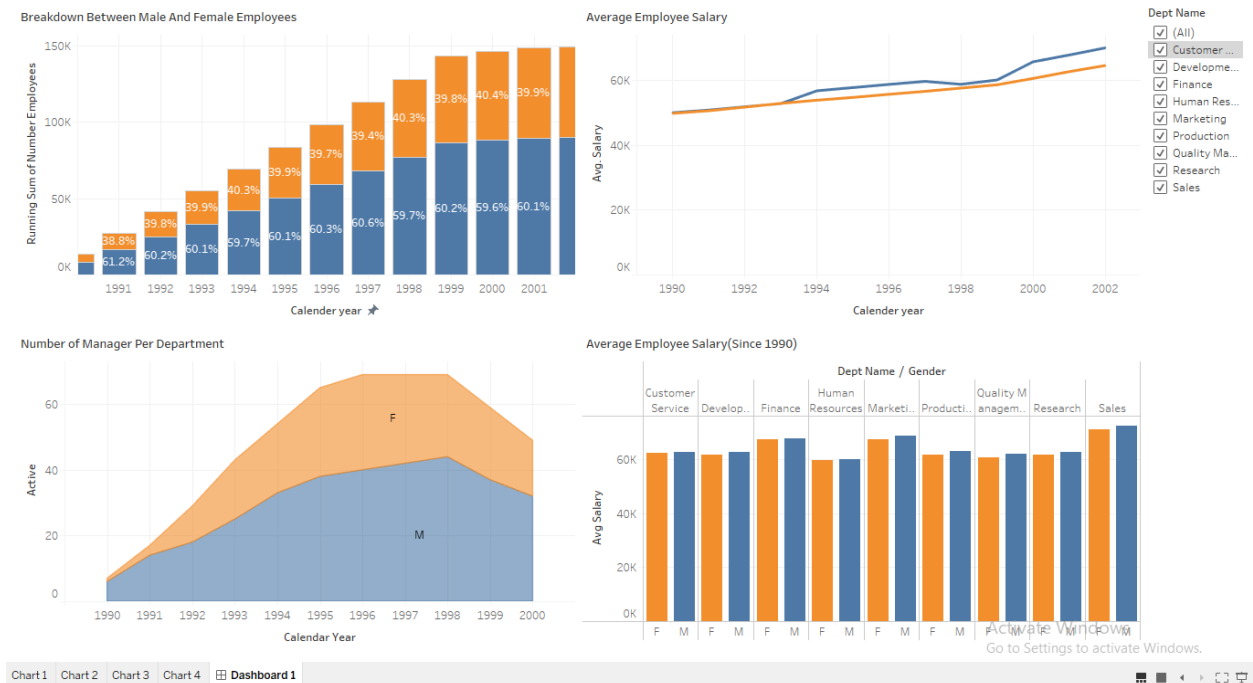
Query Completed



**TASK4:** 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. Finally, visualize the obtained result-set in Tableau as a double bar chart.



## FINAL DASHBOARD:



PS: The final dashboard copy is available on Tableau Public, [click here](#) to access it. The raw code is available on GitHub, [click here](#) to view it.