**Database build**

I imported the csv file into the Access and stored it in a new table called “employees”, the first field is not imported. The Primary Key for this table is the field “EmployeeNumber”, since it is obvious that this would be a unique identifier for each employee. Also for this project, I have noticed that the raw data might be combined from the company’s HR database and some surveys. Instead of production environment’s database design, some of the fields are pre-calculated such as “YearsAtCompany”, so there is no need to build such as “departments” and “jobs” table separately. Further, in order to keep the “employees” table as original (and simple) from the raw data as possible, I decided not to omit any field, and not to use Data Types such as “Yes/No” since it will change it from the datasheet view. I only used two Data Types: “Number” for numerical values, and “Short Text” for text values as the default setting done by Access. I have changed the field name “Starting Salary” and “Current Salary” to “StartingSalary” and “CurrentSalary” to avoid space, same as other field names follow the CamelCase.

I also created 7 tables based on the “Descriptors for numeric ratings”, since I will use them in order to provide descriptors for some of the fields in “employees” table when creating some queries. These descriptor tables are quite similar, I manually created them in the Access and use the numerical numbers as their Primary Key, and named after its “[table name] + ID”. The descriptor’s field name named “[table name] + Desc”. Take PerformaceRating’s design view as an example:

|  |  |
| --- | --- |
| **Field Name** | **Data Types** |
| PerformanceRatingID (PK) | Number |
| PerformanceRatingDesc | Short Text |

After the descriptor tables created, I populate the tables with the records as shown in the excel file.

**Relationships**

I have built relationships between “employees” table and each descriptor table:

(PK = Primary Key, FK = Foreign Key)

|  |  |  |  |
| --- | --- | --- | --- |
| **Table** | **Field** | **Table** | **Field** |
| employees | Education (FK) | Education | EducationID (PK) |
| EnvironmentSatisfaction (FK) | EnvironmentSatisfaction | EnvironmentSatisfactionID (PK) |
| JobInvolvement (FK) | JobInvolvement | JobInvolvementID (PK) |
| JobSatisfaction (FK) | JobSatisfaction | JobSatisfactionID (PK) |
| PerformanceRating (FK) | PerformanceRating | PerformanceRatingID (PK) |
| RelationshipSatisfaction (FK) | RelationshipSatisfaction | RelationshipSatisfactionID (PK) |
| WorkLifeBalance (FK) | WorkLifeBalance | WorkLifeBalanceID (PK) |

So we can create a query when we need to know the descriptor for a certain criteria.

**Queries**

Query1: Shows the employee number with corresponding starting salary, ordered by the Employee number as default, expected.

Query2: Shows the employee number with corresponding current salary, ordered by the Employee number as default, expected.

Query3: Shows the employee number with corresponding gender of each employee, ordered by the Employee number (primary key) as default, expected.

Query4: Shows the employee number with corresponding age of each employee, ordered by the primary key as default.

Query5: Shows the average starting salary given the sample of 800 employees, expected. Note that average starting salary could vary from many aspects.

Query6: Shows the maximum age of employees where the criteria YearsAtCompany < 1.

Query7: Shows the minimum starting salary of the sample employees group by gender. Noticed Female has higher minimum starting salary (4516) than Male (4036), this could because of special individual cases, might not apply overall.

Query8: I rounded the average number by 2 decimal points which gives me a better look. Noticed it applies to both male and female, when job level increases, the average current salary not necessarily increases. And for each job level, female average current salaries are higher than male, except for level 4 with slightly lower than male.

Query9: In order to show the descriptor of the average performanceRating of each department, I used the Round() function to round the average number into integer then match with the descriptors. In order to see the difference between departments, I used Round() function to round the average in 2 decimal point. Lastly sorted lowest to highest.

Query10: Set a parameter query under the EmployeeNumber field [Please type employee number], I decided to show EmployeeNumber field too along with DistanceFromHome and BusinessTravel since it would present the query result better. As expected, if run the query, Access will ask the user to type in the employee number prior showing the result.