# Employee Database Project

By: Bob Palmer

Applied Database Management IS362DLS2A2016

Week 8 - Final Project

## Purpose

For this project, I have developed a database which contains employee information. Use of this database can be universally applied to time-clock, payroll, tax, and human resource applications. Included in this file is the data dictionary, the logical model, the "create" table, the physical model, an input form, an output report, an update/modify example, and three sample queries with results. Please see the included file “Palmer\_EMP\_DATA.accdb” to review actual implementation.

Each employee must have exactly one Pay\_Type. Salaried employees do not clock in, and therefore, cannot have a variable Pay\_Rate. Hourly employees, on the other hand, can hanve a variable Pay\_Rate based on the position which they are working. This has practical application in many restaurant and retail settings, and also allows for the implementation of a shift differential for certain positions (such as overnight stocker or prep cook).

Bonuses are calculated based on a percentage of total sales, as long as the total sales exceed a defined minimum. This minimum can be defined based on any number of factors by an external application.

## Data Dictionary

Employee\_ID: (AutoNumber) The internal identifier for an employee. Used across all tables which reference that employee.

First\_Name: (Short\_Text) Employee’s first name.

Middle\_Name: (Short\_Text) Employee’s middle name.

Last\_Name: (Short\_Text) Employee’s last name.

Street\_Address: (Short\_Text) Employee’s street address, including house number and street name.

City: (Short\_Text) Employee’s city of residence.

State: (Short\_Text) Employee’s state of residence, non-abbreviated with first letter capitalized.

Zip: (Short\_Text) Employee’s Zip Code

SSN: (Short\_Text) Employee’s Social Security Number

Pay\_Type: (Number) How the employee is paid. Descriptions are found in the Pay\_Type table field “PT\_Name”.

PT\_Name: (Short\_Text) A text description of the numerical pay type.

Pay\_Rate: (Currency) Dollar amount of an hourly employee’s hourly wage.

Job\_Type: (Number) The employee’s job type. Descriptions are found in the Job\_Type table field “Job\_Description”

Job\_Description: (Short\_Text) A text description of the numerical job type.

Salary: (Currency) Dollar amount of a salaried employee’s annual salary.

Min\_Weekly\_Sales: (Currency) The minimum store sales required for a Salaried With Bonus employee to receive their bonus.

Percentage: (Number) The percentage of sales which the Salaried With Bonus employee receives as a bonus. Represented as a whole number.

## Logical Model

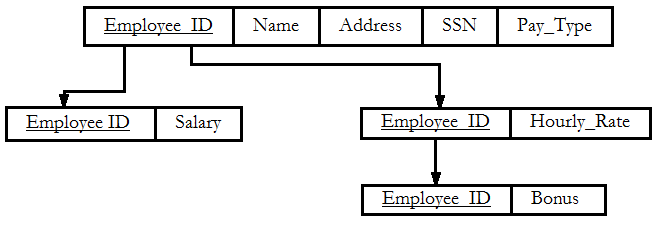
## C:\Users\Bob\AppData\Local\Microsoft\Windows\INetCache\Content.Word\EER.PNG

EER Diagram

Total Specialization Rule (Hoffer, et al, pp.121 fig. 3-6a)

Partial Specialization Rule (Hoffer, et al, pp.123 fig. 3-6b)

Disjoint Rule (Hoffer, et al, pp.124 fig. 3-7a)



3rd Normal Form (Hoffer, et al., pp 186, fig 4-23 )

## “Create” Table

**CREATE TABLE** Pay\_Type

(ID NUMBER(11,0) **NOT NULL**,

PT\_Name VARCHAR(25),

**CONSTRAINT** Pay\_Type \_PK **PRIMARY KEY** (ID));

**CREATE TABLE** Employee\_Base

(Employee\_ID NUMBER(11,0) **NOT NULL**,

First\_Name VARCHAR(25) **NOT NULL**,

Middle\_Name VARCHAR(25),

Last\_Name VARCHAR(25),

Street\_Address VARCHAR(50),

City VARCHAR(25),

State VARCHAR(25),

Zip VARCHAR(10),

SSN VARCHAR(11),

Pay\_Type NUMBER (11,0) **NOT NULL,**

**CONSTRAINT** Employee\_Base\_PK **PRIMARY KEY** (Employee\_ID),

**CONSTRAINT** Employee\_Base\_FK **FOREIGN KEY** (Pay\_Type) REFERENCES Pay\_Type (ID));

**CREATE TABLE** Job\_Type

(ID NUMBER(11,0) **NOT NULL**,

Job\_Description VARCHAR(50),

**CONSTRAINT** Job\_Type \_PK **PRIMARY KEY** (ID));

**CREATE TABLE** Hourly\_Rates

(ID NUMBER(11,0) **NOT NULL**,

Pay\_Rate DECIMAL(6,2) **NOT NULL**,

Employee\_ID NUMBER(11,0) **NOT NULL**,

Job\_Type NUMBER (11,0) **NOT NULL,**

**CONSTRAINT** Pay\_Type \_PK **PRIMARY KEY** (ID),

**CONSTRAINT** Hourly\_Rates\_FK **FOREIGN KEY** (Employee\_ID) REFERENCES Employee\_Base (Employee\_ID),

**CONSTRAINT** Hourly\_Rates\_FK **FOREIGN KEY** (Job\_Type) REFERENCES Job\_Type (ID));

**CREATE TABLE** Salaries

(ID NUMBER(11,0) **NOT NULL**,

Salary DECIMAL(11,2) **NOT NULL**,

Employee\_ID NUMBER(11,0) **NOT NULL**,

Job\_Type NUMBER (11,0) **NOT NULL,**

**CONSTRAINT** Salaries \_PK **PRIMARY KEY** (ID),

**CONSTRAINT** Salaries \_FK **FOREIGN KEY** (Employee\_ID) REFERENCES Employee\_Base (Employee\_ID),

**CONSTRAINT** Salaries \_FK **FOREIGN KEY** (Job\_Type) REFERENCES Job\_Type (ID));

**CREATE TABLE** Bonus

(ID NUMBER(11,0) **NOT NULL**,

Min\_Weekly\_Sales DECIMAL(11,2),

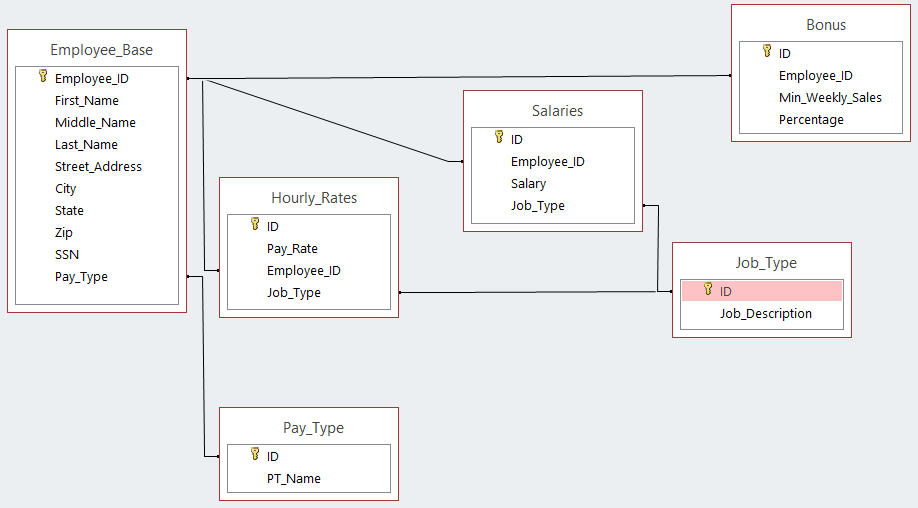
Employee\_ID NUMBER(11,0) **NOT NULL**,

Percentage NUMBER (11,0) **NOT NULL,**

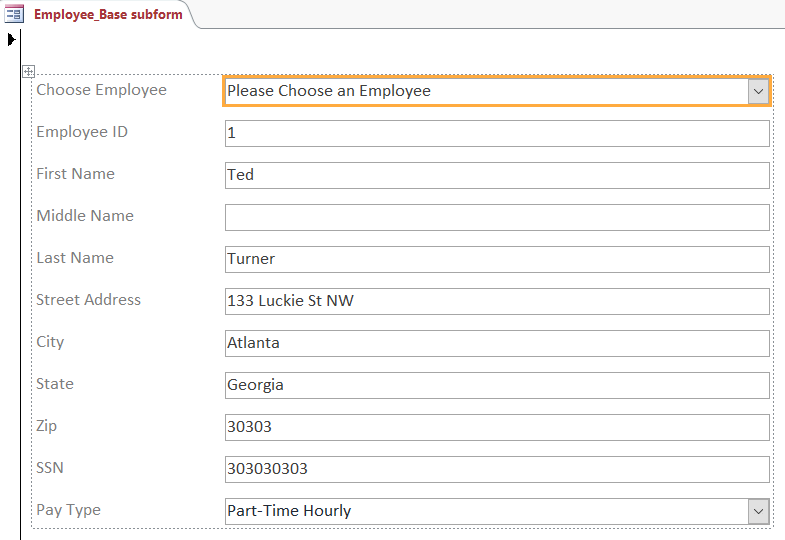
**CONSTRAINT** Bonus \_PK **PRIMARY KEY** (ID),

**CONSTRAINT** Bonus \_FK **FOREIGN KEY** (Employee\_ID) REFERENCES Employee\_Base (Employee\_ID));

## Physical Model

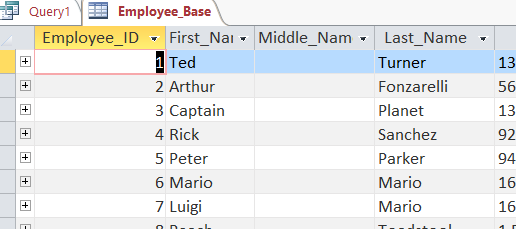


## Input Form and Output Report

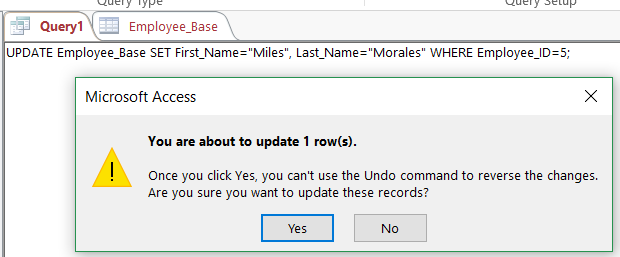


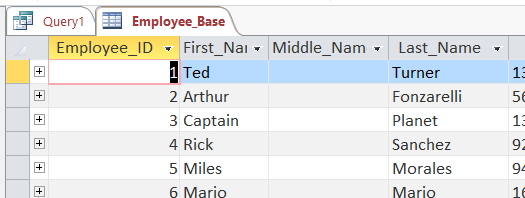


## Update/Modify Example



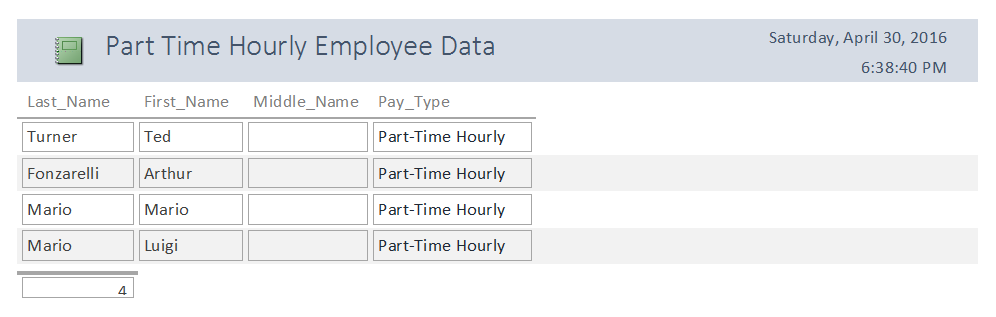
UPDATE Employee\_Base SET First\_Name="Miles", Last\_Name="Morales" WHERE Employee\_ID=5;





## Three Sample Queries w/ Results

### 1

SELECT \* FROM Employee\_Base WHERE Pay\_Type = (SELECT ID FROM Pay\_Type WHERE PT\_Name = "Part-Time Hourly") ORDER BY Last\_Name;

### 2

SELECT Employee\_Base.Last\_Name, Employee\_Base.First\_Name, Employee\_Base.Middle\_Name, Pay\_Type.PT\_Name, Salaries.Salary, Bonus.Percentage, Bonus.Min\_Weekly\_Sales

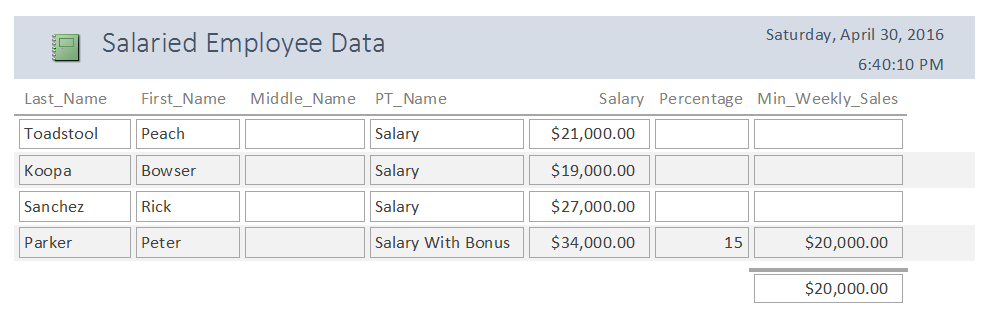
FROM ((Employee\_Base INNER JOIN Salaries

ON Employee\_Base.Employee\_ID=Salaries.Employee\_ID)

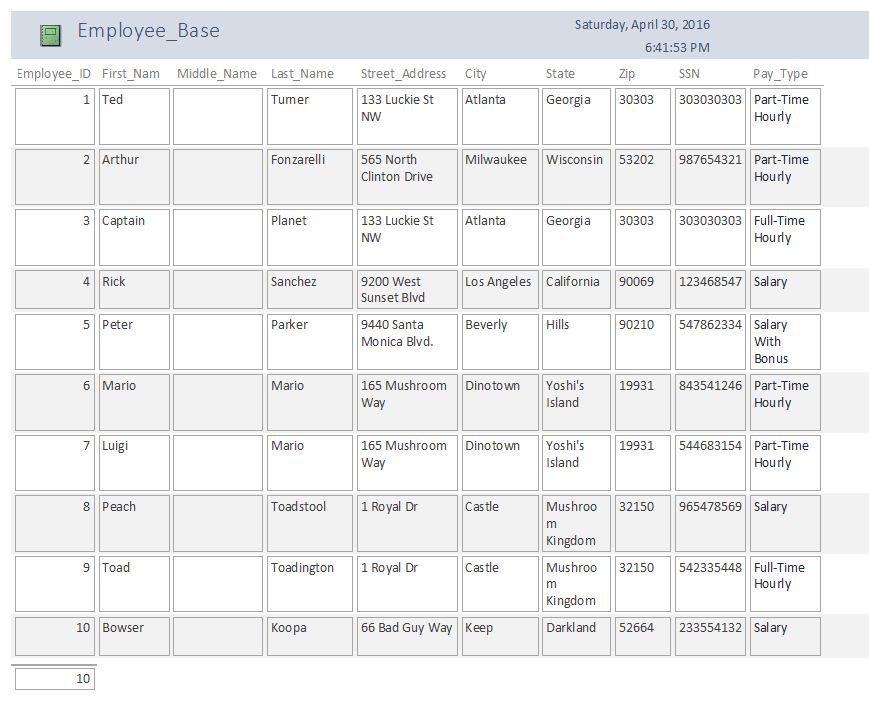
INNER JOIN Pay\_Type

ON Employee\_Base.Pay\_Type=Pay\_Type.ID)

LEFT JOIN Bonus

ON Employee\_Base.Employee\_ID=Bonus.Employee\_ID;(SQL INNER JOIN Keyword, 1999), (SQL COUNT() Function, 1999)

### 3

SELECT \* FROM Employee\_Base ORDER BY Employee\_ID;

## Sources:

Hoffer, J. A., Ramesh, V., & Topi, H. (2016). Modern database management (12th ed.). Pearson.

SQL INNER JOIN Keyword. (1999). Retrieved April 7, 2016, from <http://www.w3schools.com/sql/sql_join_inner.asp>