BridgeLabz

Employability Delivered

SQL & ADO.NET

Outcome

- Ability to create Employee Payroll DB
- Ability to use Normalization and ER Modeling to create a Employee Payroll ER Model
- Ability to implement DB Queries using MSSQL Client (SSMS)
- Ability to manage Employee Payroll data from the .NET Application using ADO.NET

BridgeLabz
Employability Delivered

Section 1: MSSQL DB

BridgeLabz
Employability Delivered

UC 1

Ability to create a payroll service database

- Using MSSQL Client use `create database` query to create a payroll_service database
- Also you can see the DB created by using `show database` query
- And go to the database created by using `use payroll_service` query

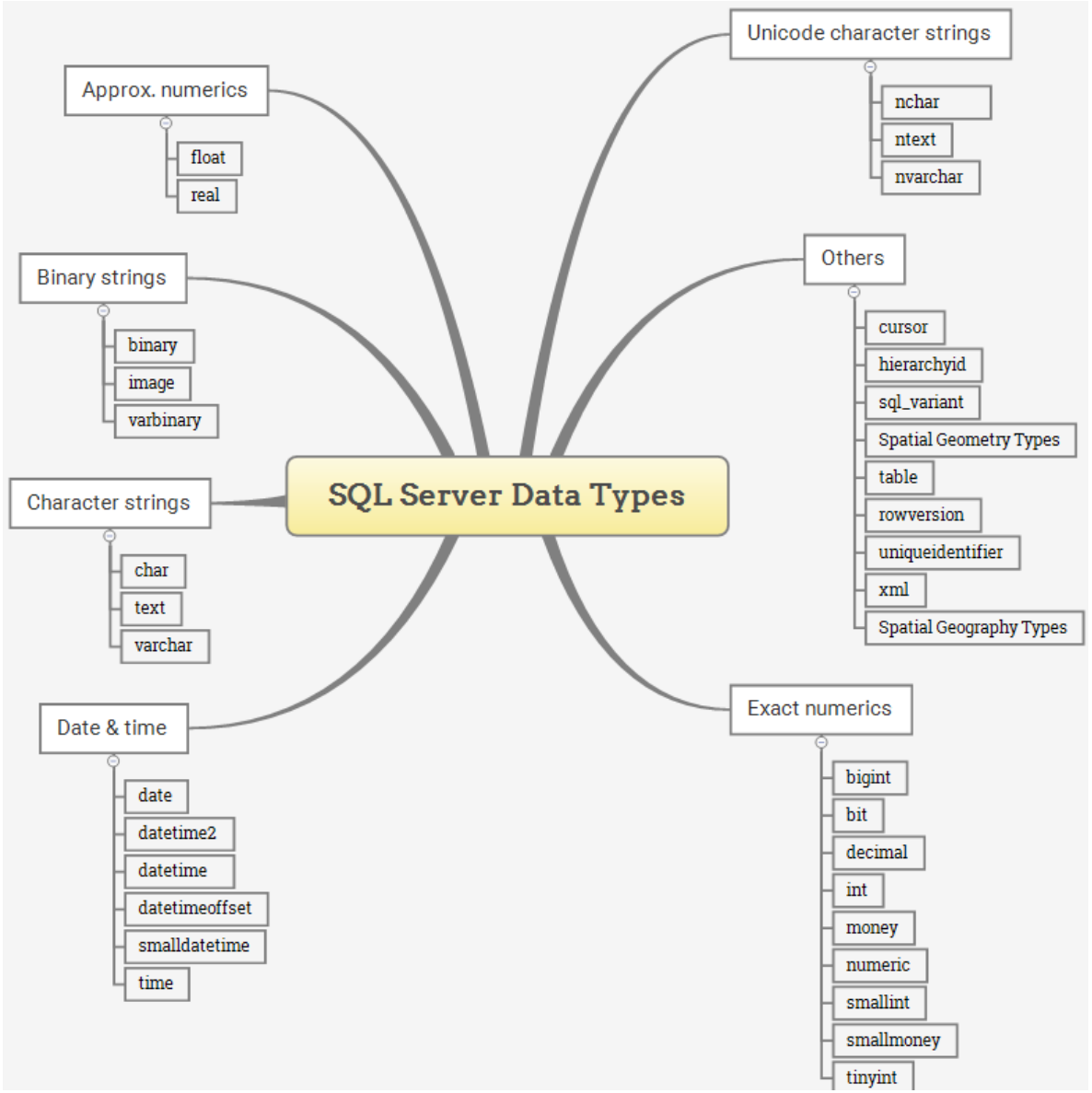
BridgeLabz
Employability Delivered

UC 2

Ability to create a employee payroll table in the payroll service database to manage employee payrolls

- Use payroll_service database in MSSQL Client
- Use `Create Table employee_payroll` Query to create employee payroll table with columns id, name, salary and start date as column. Note Id is set to auto increment.
- Understand the SQL data types to be used for the table
- Note: SQL Queries as case insensitive

BridgeLabz
Employability Delivered



MSSQL Data Types

BridgeLabz
Employability Delivered

UC 3

Ability to create employee payroll data in the payroll service database as part of **CRUD** Operation

- Use payroll_service database in MSSQL Client
- Use `INSERT INTO employee_payroll` Query to create employees payroll data into the employee_payroll table

BridgeLabz
Employability Delivered

UC 4

Ability to retrieve all the employee payroll data that is added to payroll service database

- Use payroll_service database in MSSQL Client
- Use `SELECT * FROM employee_payroll` to retrieve all the data from the employee_payroll table

BridgeLabz
Employability Delivered

UC 5

Ability to retrieve salary data for a particular employee as well as all employees who have joined in a particular data range from the payroll service database

- Use `SELECT salary FROM employee_payroll WHERE name = 'Bill'` Query to View Bill's salary
- Use `Select` query with `Where` condition View employees between start dates
- Query: `WHERE start BETWEEN CAST('2018-01-01' AS DATE) AND DATE(NOW());`
- Note: Where Condition Clause is used to retrieve the row needed from the table
- Note: Use of Database Functions like `CAST()` and `NOW()` in the Query

BridgeLabz
Employability Delivered

UC 6

Ability to add Gender to Employee Payroll Table and Update the Rows to reflect the correct Employee Gender

- Use payroll_service database in MSSQL Client
- Use Alter Table Command to add Field gender after the name field
- Use Update Query to set the gender using where condition with the employee name
- E.g. `UPDATE employee_payroll set gender = 'M' where name = 'Bill' or name = 'Charlie';`

BridgeLabz
Employability Delivered

UC 7

Ability to find sum, average, min, max and number of male and female employees

- Use payroll_service database in MSSQL Client
- Use Database Function `SUM, AVG, MIN, MAX, COUNT` to do analysis by Male or Female.
- Note: You will need to use GROUP BY GENDER grouping to get the result
- E.g. `SELECT SUM(salary) FROM employee_payroll WHERE gender = 'F' GROUP BY gender;`

MSSQL Commands

```sql
CREATE TABLE employee_payroll
(
EmployeeID int identity(1, 1) primary key,
EmployeeName varchar(255),
PhoneNumber varchar(255),
Address varchar(255),
Department varchar(255),
Gender char(1),
BasicPay float,
Deductions float,
TaxablePay float,
Tax float,
NetPay float,
StartDate Date,
City varchar(255),
Country varchar(255)
);

insert into employee_payroll
(EmployeeName,PhoneNumber,Address,Department,Gender,BasicPay,Deductions,TaxablePay,T
values
('Gunjan', '78787878787', 'Mumbai', 'ENGG', 'M', 30000, 2000, 1000, 200, 18000, getd

select * from employee_payroll

select * from employee_payroll where EmployeeName='Terisa'

select schema_name(t.schema_id) as schema_name,
        t.name as table_name,
        t.create_date,
        t.modify_date
from sys.tables t
order by schema_name,
        table_name;
```

BridgeLabz
Employability Delivered

Section 2: ER Diagram

BridgeLabz
Employability Delivered

UC 8

Ability to extend employee_payroll data to store employee information like employee phone, address and department

- Ensure employee department is non nullable fields.
- Add Default Value for address field.

UC 9

Ability to extend employee_payroll table to have Basic Pay, Deductions, Taxable Pay, Income Tax, Net Pay

BridgeLabz
Employability Delivered

UC 10

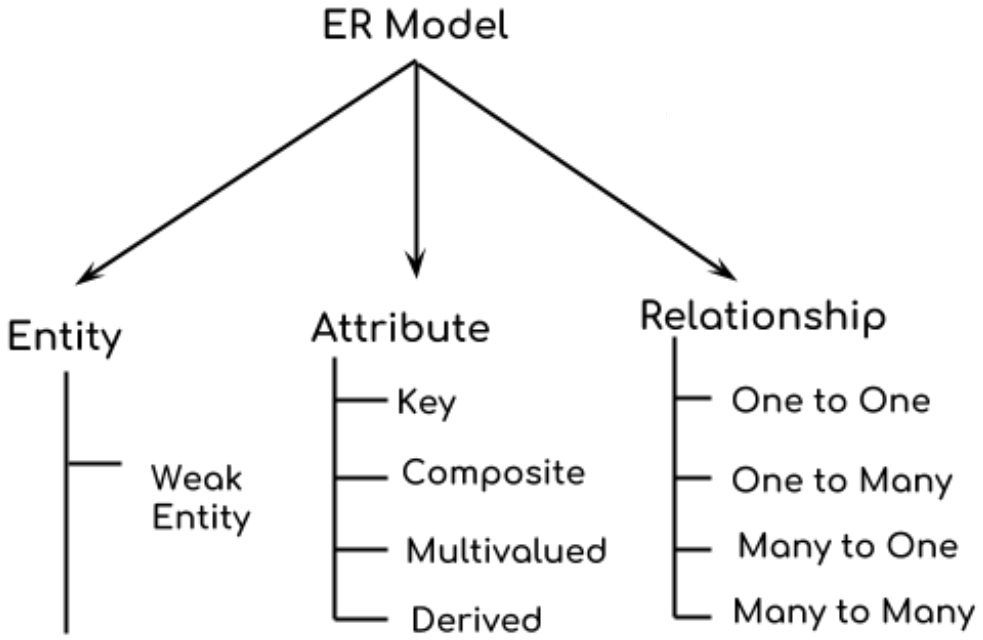
Ability to make Terissa as part of Sales and Marketing Department

- Note: The Complete employee payroll roll need to be Inserted
- If a Salary is now going to be updated multiple rows has to be updated creating unnecessary redundancy
- Further There is 2 Employee ID so according to Database there is two different Terissa

Entity Relationship Diagram – ER Diagram

BridgeLabz
Employability Delivered

- An **Entity–relationship model (ER model)** describes the structure of a database with the help of a diagram, which is known as **Entity Relationship Diagram (ER Diagram)**.
- An ER model is more intuitive approach than following rules in Normalized forms like 1NF, 2NF, etc. and is a design or blueprint of a database that can later be implemented as a database. Very similar to Class Diagram.
- The best way to design the DB is to keep the knowledge of Normalized Forms and use the Intuitive Approach and Domain Knowledge to arrive at the ER Diagram.
- The main components of E-R Diagram are:
  - Entity
  - Attribute
  - Relationship



Components of ER Diagram

BridgeLabz
Employability Delivered

Employee Payroll Table

```
Executed in 0 ms
[mssql> sp_columns 'employee_payroll'
TABLE_QUALIFIER  TABLE_OWNER  TABLE_NAME       COLUMN_NAME   DATA_TYPE  TYPE_NAME     PRECISION  LENGTH  SCALE  RADIX  NULLABLE  REMARKS  COLUMN_DEF  SQL_DATA_TYPE  SQL_DATE
RDINAL_POSITION  IS_NULLABLE  SS_DATA_TYPE
---------------  -----------  ---------------  ------------  ---------  ------------  ---------  ------  -----  -----  --------  -------  ----------  -------------  --------
---------------  -----------  ------------
payroll_service  dbo          employee_payroll  EmployeeID    4          int identity  10         4       0      10     0         null     null        4              null
                 NO           56
payroll_service  dbo          employee_payroll  EmployeeName  12         varchar       255        255     null   null   1         null     null        12             null
                 YES          39
payroll_service  dbo          employee_payroll  PhoneNumber   12         varchar       255        255     null   null   1         null     null        12             null
                 YES          39
payroll_service  dbo          employee_payroll  Address       12         varchar       255        255     null   null   1         null     null        12             null
                 YES          39
payroll_service  dbo          employee_payroll  Department    12         varchar       255        255     null   null   1         null     null        12             null
                 YES          39
payroll_service  dbo          employee_payroll  Gender        1          char          1          1       null   null   1         null     null        1              null
                 YES          39
payroll_service  dbo          employee_payroll  BasicPay      6          float         15         8       null   10     1         null     null        6              null
                 YES          109
payroll_service  dbo          employee_payroll  Deductions    6          float         15         8       null   10     1         null     null        6              null
                 YES          109
payroll_service  dbo          employee_payroll  TaxablePay    6          float         15         8       null   10     1         null     null        6              null
                 YES          109
payroll_service  dbo          employee_payroll  Tax           6          float         15         8       null   10     1         null     null        6              null
0                YES          109
payroll_service  dbo          employee_payroll  NetPay        6          float         15         8       null   10     1         null     null        6              null
1                YES          109
payroll_service  dbo          employee_payroll  StartDate     -9         date          10         20      null   null   1         null     null        -9             null
2                YES          0
payroll_service  dbo          employee_payroll  City          12         varchar       255        255     null   null   1         null     null        12             null
3                YES          39
payroll_service  dbo          employee_payroll  Country       12         varchar       255        255     null   null   1         null     null        12             null
4                YES          39

14 row(s) returned
```

BridgeLabz
Employability Delivered

Employee Payroll Table

```
Executed in 0 ms
[mssql> select * from employee_payroll where EmployeeName='Terisa';
EmployeeID  EmployeeName  PhoneNumber  Address  Department  Gender  BasicPay  Deductions  TaxablePay  Tax  NetPay  StartDate                 City    Country
----------  ------------  -----------  -------  ----------  ------  --------  ----------  ----------  ---  ------  ------------------------  ------  -------
1           Terisa        78787878787  Mumbai   HR          F       20000     2000        1000        200  18000   2020-10-22T00:00:00.000Z  Mumbai  IN

1 row(s) returned
```

```
25
26    select * from employee_payroll where EmployeeName='Terisa'
```

Results Messages

	EmployeeID	EmployeeName	PhoneNumber	Address	Department	Gender	BasicPay	Deductions	TaxablePay	Tax	NetPay	StartDate	City	Country
1	1	Terisa	78787878787	Mumbai	HR	F	20000	2000	1000	200	18000	2020-10-22	Mumbai	IN

BridgeLabz
Employability Delivered

UC 10

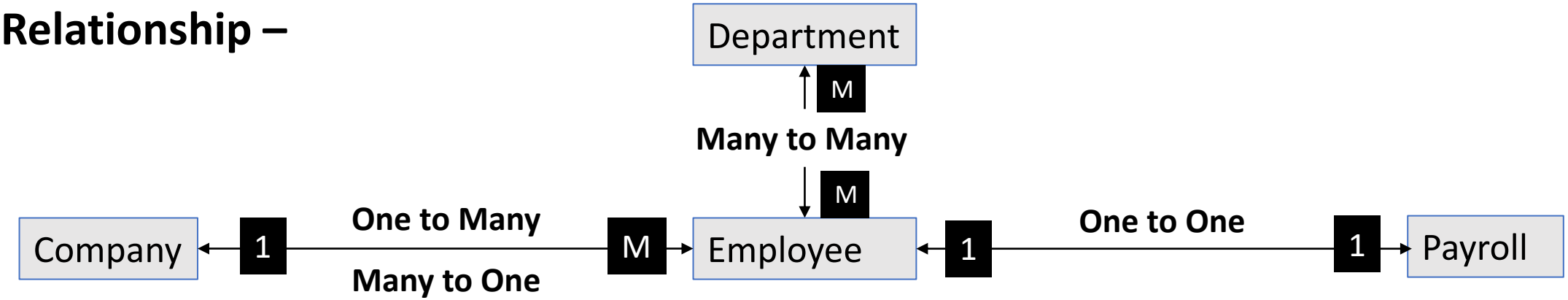
Draw the ER Diagram for Payroll Service DB

- Identifies the Entities – Entities can be Identified using Normalization Technique
- Check each attribute and see if they are Composite or Multi-Valued

Entity Relationship Diagram – ER Diagram

BridgeLabz
Employability Delivered

- **Entity –** An Entity is similar to a Class or a Table in DB and is identified by its attributes and relationship with other entities. An Entity can be a **Week Entity** if it dependent on other Entity.
- **Attribute –** An Attribute is the property of an Entity and are following 4 Types:
  - **Key attribute –** Uniquely Identified the Entity **e.g. employee id**
  - **Composite attribute –** is a combination of other attributes **e.g. employee address**
  - **Multivalued attribute –** hold multiple values like the **phone number**
  - **Derived attribute –** value is dynamic and derived from another attribute **e.g. net pay**
- **Relationship –**



BridgeLabz
Employability Delivered

UC 11

Implement the ER Diagram into Payroll Service DB

- Create the corresponding tables as identified in the ER Diagram along with attributes
- For Many to Many relationship, create new Table called Employee Department having Employee Id and Department ID and redo the UC 7

BridgeLabz
Employability Delivered

UC 12

Ensure all retrieve queries done especially in UC 4, UC 5 and UC 7 are working with new table structure

BridgeLabz
Employability Delivered

Section 3: ADO.NET

UC 1

Ability to create a payroll service database and have java program connect to database

- Use the payroll_service database created in MSSQL
- Download and register the Ado.Net driver to your source in Visual Studio
- Alternatively use NuGet to add Ado.Net driver to your source
- Check if the Driver Class is available for the C# program
- Check all the drivers registered with the Ado.Net driver
- Check if the database connection to payroll_service mssql DB is established

Test Database Connectivity using C# Program

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.Text;

namespace EmployeePayroll
{
    class EmployeeRepo
    {
        public static string connectionString = "Data Source=(localdb)\\MSSQLLocalDB;Initial Catalog=Employee_PayRole;
        SqlConnection connection = new SqlConnection(connectionString);
        public DataSet GetAllEmployee()
        {
            try
            {
                DataSet dataset = new DataSet();
                using (this.connection)
                {
                    this.connection.Open();
                    SqlDataAdapter adapter = new SqlDataAdapter("SPGetEmployeeList", this.connection);
                    adapter.Fill(dataset);
                    this.connection.Close();
                    return dataset;
                }
            }
            catch (Exception e)
            {
                throw new Exception(e.Message);
            }
            finally
            {
                this.connection.Close();
            }
        }
```

- Test Database connectivity. Please note the following:
  - Set the DB URL to local host database you have created.
  - Check the Username and Password set in MSSQL
  - Check if MSSQL Ado.Net driver class is loaded
  - List the MSSQL Ado.Net Drivers Registered
  - Get the SQL Connection from the Ado.Net Driver passing the DB URL, User name and Password
  - Output of the C# Program is Connection Established

ADO.NET API Core Concepts

BridgeLabz
Employability Delivered

- **ADO.NET Statement –** The **Step 3** is to **create ADO.NET Statement or Prepared Statement** using the Connection Object to execute the CRUD operation
  - `public static string connectionString = "Data Source=(localdb)\\MSSQLLocalDB;Initial Catalog=payroll_service;";`
  - `SqlConnection connection = new SqlConnection(connectionString);`
- **ResultSet –** The **Step 4** is to process the result of the database query. For this Ado.Net ResultSet interface is used. The ResultSet contains **set of records** and each record contains values corresponding to the Columns retrieved from the DB

BridgeLabz
Employability Delivered

UC 2

Ability for Employee Payroll Service to retrieve the Employee Payroll from the Database

- Using ODBC read the employee payroll data from the database
- Add Start Data to EmployeePayroll Class and ensure backward compatibility
- Populate the EmployeePayroll Object
- Return the list of Employee Payroll Data

BridgeLabz
Employability Delivered

UC 3

Ability to update the salary i.e. the base pay for Employee Terisa to 3000000.00 and sync it with Database

- Update the employee payroll in the database
- Update the Employee Payroll Object with the Updated Salary
- Compare Employee Payroll Object with DB to pass the MSTest Test.

BridgeLabz
Employability Delivered

UC 4

Ability to update the salary i.e. the base pay for Employee Terisa to 3000000.00 and sync it with Database using JDBC PreparedStatement

- Update the employee payroll in the database
- Update the Employee Payroll Object with the Updated Salary
- Compare Employee Payroll Object with DB to pass the MSTest Test.

BridgeLabz
Employability Delivered

Refactor UC 4

Refactor Code to do the following

- Create PreparedStatement to retrieve payroll data by name
- Cache the PreparedStatement at the Driver and DB Level
- Make Payroll DB Service Object as Singleton so PreparedStatement is cached in the Program
- Reuse the ResultSet to populate EmployeePayrollData Object

BridgeLabz
Employability Delivered

UC 5

Ability to retrieve all employees who have joined in a particular data range from the payroll service database

BridgeLabz
Employability Delivered

UC 6

Ability to find sum, average, min, max and number of male and female employees

- Use Database Function `SUM, AVG, MIN, MAX, COUNT` to do analysis by Male and Female.
- Note: You will need to use GROUP BY GENDER grouping to get the result
- E.g. `SELECT SUM(salary) FROM employee_payroll WHERE gender = 'F' GROUP BY gender;`

BridgeLabz
Employability Delivered

UC 7

Ensure UC 2 – UC 7 works with the new ER Diagram implemented into Payroll Service DB

- Ensure the EmployeePayroll Class incorporates all the Entities identified in the ER Diagram
- Ensure When Adding new Employee Payroll, many tables will be impacted and transaction in the table need to be implemented
- For Department, Employee Payroll class can hold array of Department Name

BridgeLabz
Employability Delivered

UC 8

Implement the complete ER Diagram in the Database

- Refactor the code to ensure all the queries are working
- Ensure the EmployeePayroll Class incorporates all the Entities identified in the ER Diagram
- Ensure When Adding new Employee Payroll, many tables will be impacted and transaction in the table need to be implemented
- For Department, Employee Payroll class can hold array of Department Name

BridgeLabz
Employability Delivered

UC 9

Ensure UC 2 – UC 7 works with the new ER Diagram implemented into Payroll Service DB

BridgeLabz

Employability Delivered

Thank You