**1) Create .net core project and Install below packages in .net core project**

**Microsoft.EntityFrameworkCore.SqlServer**

**Microsoft.EntityFrameworkCore.Tools**

**2) create a folder name as ‘Context’ and create a Model class as ‘Employee.cs’ in this**

**With the following properties as follows.**

namespace CodeFirstMigration.Context

{

**public class** Employee

{

**public int** EmployeeId { get; set; }

**public** string Name { get; set; }

**public** string Address { get; set; }

**public** string CompanyName { get; set; }

**public** string Designation { get; set; }

}

}

**3) We will create another class inside the Context folder as ‘EmployeeDbContext’ that will inherit to DbContext class.**

using Microsoft.EntityFrameworkCore;

namespace CodeFirstMigration.Context

{

**public class** EmployeeDbContext : DbContext

{

**public** EmployeeDbContext(DbContextOptions options) : base(options)

{

}

DbSet<Employee> Employees { get; set; }

}

}

**4) Add connection string inside the appsetting.json file as follows.**

{

"Logging": {

"LogLevel": {

"Default": "Information",

"Microsoft": "Warning",

"Microsoft.Hosting.Lifetime": "Information"

}

},

"ConnectionStrings": {

"EmployeeDBConnection": "Server=DESKTOP-3DF0FM6\\SQLEXPRESS;Database=EmployeeDBTemp;Trusted\_Connection=True"

},

"AllowedHosts": "\*"

}

**5) Add the below code in ConfigureServices method in startup.cs file**

services.AddDbContext<EmployeeDbContext>(item => item.UseSqlServer

(Configuration.GetConnectionString("EmployeeDBConnection")));

**6) Now open the “Package Manager Console”, and run the below commands**

PM> add-migration initialmigration

PM> update-database initialmigration

**7) Go to EmployeeDbContext class and override the DbContext method 'OnModelCreating'. Using the help of the ModelBuilder, we can add some dummy data for the Employees table as follows.**

**So, above we have already prepared the data for seeding. Now, let's add it to migration**

using Microsoft.EntityFrameworkCore;

namespace CodeFirstMigration.Context

{

**public class** EmployeeDbContext : DbContext

{

**public** EmployeeDbContext(DbContextOptions options) : base(options)

{

}

DbSet<Employee> Employees { get; set; }

**protected** override **void** OnModelCreating(ModelBuilder modelBuilder)

{

modelBuilder.Entity<Employee>().HasData(

**new** Employee() { EmployeeId = 1, Name = "John", Designation="IT"

**new** Employee() { EmployeeId = 2, Name = "Chris", Designation="IT"

**new** Employee() { EmployeeId = 3, Name = "Mukesh", Designation="IT"});

}

}

**8) Open Package Manager Console and run the below command to add dumy data into Employee table**

PM> add-migration seedEmpData

PM> update-database seedEmpData

**It will add the dummy data to Employee table**

**9) If we want to add Department table to database then you need to add Department.cs class in context folder**

public class Department

{

[Key]

public int DeptId { get; set; }

public string DeptName { get; set; }

}

**10) run the below command in package manager console to add table into database**

PM> add-migration addTblDept

PM> update-database addTblDept

**It will add the Department table in SQL server**

**11) To insert sample data into Department table add below code in EmployeeDbContext.cs -> OnModelCreating() method**

namespace NetCoreCodeFirstApproach.Context

{

public class EmployeeDbContext : DbContext

{

public EmployeeDbContext(DbContextOptions options) : base(options)

{

}

DbSet<Employee> Employees { get; set; }

DbSet<Department> Departments { get; set; }

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

modelBuilder.Entity<Employee>().HasData(

new Employee() { EmployeeId = 1, Name = "John", Designation = "Developer", Address = "New York", CompanyName = "XYZ Inc" },

new Employee() { EmployeeId = 2, Name = "Chris", Designation = "Manager", Address = "New York", CompanyName = "ABC Inc" },

new Employee() { EmployeeId = 3, Name = "Mukesh", Designation = "Consultant", Address = "New Delhi", CompanyName = "XYZ Inc" });

modelBuilder.Entity<Department>().HasData(

new Department() { DeptId = 1, DeptName = "IT" });

}

}

}

**12) Open Package Manager Console and run the below command to add dumy data into Department table**

PM> add-migration seedDeptdata

PM> update-database seedDeptdata

**This will add the data into Department table**

**13) If we want to add DeptLocation column into Department table then we need to add filed into Department.cs file as below**

namespace NetCoreCodeFirstApproach.Context

{

public class Department

{

[Key]

public int DeptId { get; set; }

public string DeptName { get; set; }

public string DeptLocation { get; set; }

}

}

**13) Open Package manager console and run the below command to add DeptLocation column to database**

**PM>** add-migration addColDeptLocation

PM> update-database addColDeptLocation

**It will add the DeptLocation in Department table**

**14) Add api controller and write the below code**

namespace NetCoreCodeFirstApproach.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class EmployeeController : ControllerBase

{

private EmployeeDbContext employeeDbContext = null;

public EmployeeController(EmployeeDbContext \_employeeDbContext)

{

this.employeeDbContext = \_employeeDbContext;

}

public IActionResult GetEmployeeList()

{

var result = this.employeeDbContext.Employees.ToList();

return Ok(result);

}

}

}