[HttpDelete("{id:int}")]

public async Task<ActionResult<Employee>> DeleteEmployee(int id)

{

try

{

var employeeToDelete = await employeeRepository.GetEmployee(id);

if (employeeToDelete == null)

{

return NotFound($"Employee with Id = {id} not found");

}

return await employeeRepository.DeleteEmployee(id);

}

catch (Exception)

{

return StatusCode(StatusCodes.Status500InternalServerError,

"Error deleting data");

}

}

}

}

Copy

EmployeeRepository

EmployeeRepository uses entity framework core to store and retrieve data from SQL Server database.

public class EmployeeRepository : IEmployeeRepository

{

private readonly AppDbContext appDbContext;

public EmployeeRepository(AppDbContext appDbContext)

{

this.appDbContext = appDbContext;

}

public async Task<Employee> AddEmployee(Employee employee)

{

var result = await appDbContext.Employees.AddAsync(employee);

await appDbContext.SaveChangesAsync();

return result.Entity;

}

public async Task<Employee> DeleteEmployee(int employeeId)

{

var result = await appDbContext.Employees

.FirstOrDefaultAsync(e => e.EmployeeId == employeeId);

if (result != null)

{

appDbContext.Employees.Remove(result);

await appDbContext.SaveChangesAsync();

return result;

}

return null;

}

public async Task<Employee> GetEmployee(int employeeId)

{

return await appDbContext.Employees

.FirstOrDefaultAsync(e => e.EmployeeId == employeeId);

}

public async Task<Employee> GetEmployeeByEmail(string email)

{

return await appDbContext.Employees

.FirstOrDefaultAsync(e => e.Email == email);

}

public async Task<IEnumerable<Employee>> GetEmployees()

{

return await appDbContext.Employees.ToListAsync();

}

public async Task<Employee> UpdateEmployee(Employee employee)

{

var result = await appDbContext.Employees

.FirstOrDefaultAsync(e => e.EmployeeId == employee.EmployeeId);

if (result != null)

{

result.FirstName = employee.FirstName;

result.LastName = employee.LastName;

result.Email = employee.Email;

result.DateOfBrith = employee.DateOfBrith;

result.Gender = employee.Gender;

result.DepartmentId = employee.DepartmentId;

result.PhotoPath = employee.PhotoPath;

await appDbContext.SaveChangesAsync();

return result;

}

return null;

}

}