

// Interfaz para la capa de acceso a la base de datos

```
public interface DatabaseConnection {  
    Connection getConnection();  
}
```

// Implementación de la interfaz para una base de datos específica (por ejemplo, MySQL)

```
public class MySQLDatabaseConnection implements DatabaseConnection {  
    public Connection getConnection() {  
        // Lógica para obtener la conexión a la base de datos MySQL  
    }  
}
```

// Clase de negocios que utiliza el repositorio de empleados a través de la capa de abstracción

```
public class EmployeeService {  
    private EmployeeRepository employeeRepository;  
  
    public EmployeeService(DatabaseConnection databaseConnection) {  
        Connection connection = databaseConnection.getConnection();  
        // Crear una instancia del repositorio de empleados utilizando la conexión a la base de datos  
        this.employeeRepository = new EmployeeRepositoryImpl(connection);  
    }  
  
    // Métodos de negocio que utilizan el repositorio de empleados  
    public List<Employee> getAllEmployees() {  
        return employeeRepository.getAllEmployees();  
    }  
  
    public Employee getEmployeeById(int id) {  
        return employeeRepository.getEmployeeById(id);  
    }  
}
```

```
}
```

```
public void addEmployee(Employee employee) {  
    employeeRepository.addEmployee(employee);  
}
```

```
public void updateEmployee(Employee employee) {  
    employeeRepository.updateEmployee(employee);  
}
```

```
public void deleteEmployee(Employee employee) {  
    employeeRepository.deleteEmployee(employee);  
}
```

```
}
```