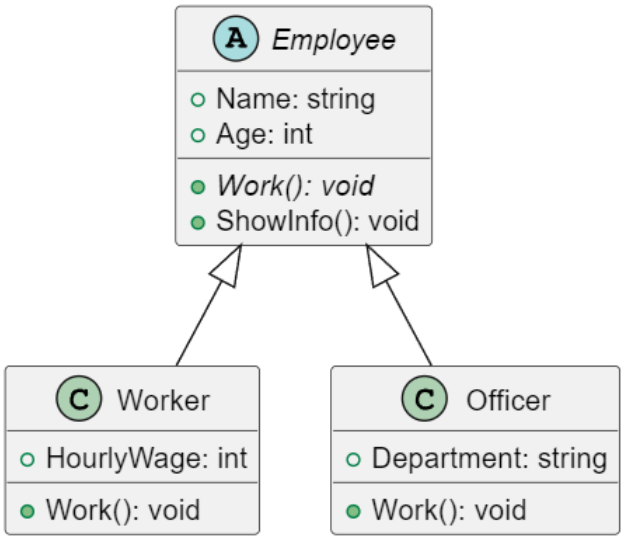
SAMSUN ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
YAZILIM MÜHENDİSLİĞİ BÖLÜMÜ
2023 – 2024 Akademik Yılı Güz Dönemi
MYAZ205 – Nesne Tabanlı Programlama

15/01/2023
Final Sınavı

1. Aşağıda verilen UML diyagramını inceleyiniz. UML diyagramı için ilgili ifadelerde boş bırakılan yerleri cevap kâğıdına geçiriniz.



`Employee` sınıfı ____(a)____ olarak tanımlandığından bu sınıftan türetme **yapılamaz**. Bu sınıfta tanımlanan _____(b)____ üyeler, alt sınıflarda geçersiz kılınmak zorundadır. `Employee` sınıfı içerisinde tanımlanan ___(c)____ üyesi bu üyelerden biri olarak gösterilebilir. Alt sınıflarda ilgili üyeleri geçersiz kılmak üzere ___(d)____ anahtar sözcüğü kullanılır.

(a) abstract / soyut (5p)
(b) abstract (5p)
(c) Work (5p)
(d) Override (5p)

2. Cevap

a) Paketler

a. Microsoft.EntityFrameworkCore (2p)

ORM işlemlerinin yapılabilmesi için temel pakettir. DbContext gibi tanımları içerir.

b. Microsoft.EntityFrameworkCore.Tools (2p)

Code-First yaklaşımını destekleyen araçları içerir. Bu araçlar, veritabanınızı kod üzerinden yönetmenize ve veritabanı işlemlerini gerçekleştirmenize yardımcı olur.

c. Microsoft.EntityFrameworkCore.Design (2p)

.NET Command Line aracılığıyla migrations (migrasyon) oluşturma ve veritabanı şemasını yönetme gibi tasarım zamanı görevlerini destekler.

d. Microsoft.EntityFrameworkCore.Sqlite (2p)

SQLite veritabanı ile ilgili EF Core sağlayıcısı kullanılabilir ve bu veritabanı üzerinde çeşitli işlemler gerçekleştirilebilir.

b) Tablo tanımlarının gerçekleştirilebilmesi için `Id` ya da `WorkerId`, `OfficerId` gibi bir alanın tanımlanması gerekir. Böylelikle veri tabanındaki tablolarda `Primary Key` alanı tanımı oluşturabilir. Sınıf diyagramında `Id`, `WorkerId` ya da `OfficerId` gibi alanların tanım olmadığı görülmektedir. Bu nedenle ya anahtarsız çalışmak istendiği özellikle belirtilmeli ya da `Primary Key` olarak kullanılabilecek alanlar tanımlanmalıdır. (12 p)

c) `RepositoryContext` sınıfını varsayılan yapıcı metot ile türetmek mümkündür. Çünkü bağlantı dizesi için yapıcı metoda herhangi bir parametre geçişi yapılmamıştır. `OnConfiguring(DbContextOptionsBuilder)` metodu da geçersiz kılınarak bağlantı ifadesi bu metot içerisinde verilebilir (5 p).

3. Cevap

```
1  public interface IWorkerRepository
2  {
3      IQueryable<Worker> GetAllWorkers();
4  }
```

(a) 5 p

```
1  public class WorkerRepository : IWorkerRepository
2  {
3      private readonly RepositoryContext _context;
4      public WorkerRepository()
5      {
6          _context = new RepositoryContext();
7      }
8      public IQueryable<Worker> GetAllWorkers()
9      {
10         return _context
11             .Workers
12             .AsQueryable<Worker>();
13     }
14 }
```

(b) 10 p

4. Aşağıda verilen kod parçasına uygun olarak gerekli tanımı yapınız (20 p).

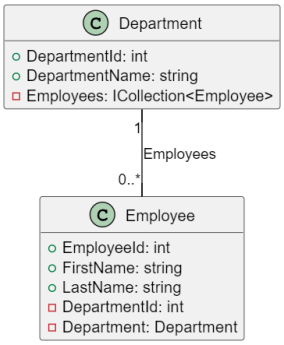
```
using Project.Models;
using Repositories.Contracts;

namespace Repositories.Extensions;

public static class RepositoryExtension
{
    public static IQueryable<Worker> GetAllWorkerWithStart(this IWorkerRepository workers, char c)
    {
        return workers
            .GetAllWorkers()
            .Where(w => w.Name.StartsWith(c));
    }
}
```

5. Cevaplar

a) UML diyagramı



b) Sınıflar

```
public abstract class Employee
{
    public int EmployeeId { get; set; }
    public string Name { get; set; }
    public int Age { get; set; }

    // Her çalışanın bir birimi vardır
    public int DepartmentId { get; set; }
    public Department Department { get; set; }

    // Abstract method
    public abstract void Work();

    // Regular method
    public string ShowInfo()
    {
        return $"Name: {Name}, Age: {Age}";
    }
}
```

```
public class Department
{
    public int DepartmentId { get; set; }
    public string DepartmentName { get; set; }
    public ICollection<Employee> Employees { get; set; } = new List<Employee>();
}
```