**14** **Работа с базами данных. Технологии доступа к данным**

Задание 1. Автоматизация учета сотрудников

Листинг задачи:

using Microsoft.EntityFrameworkCore;

using SQLitePCL;

namespace WpfApp

{

public class HrContext : DbContext

{

public DbSet<DepartmentModel> Departments { get; set; }

public DbSet<EmployeeModel> Employees { get; set; }

public HrContext()

{

SQLitePCL.Batteries.Init();

}

public HrContext(DbContextOptions<HrContext> options) : base(options)

{

SQLitePCL.Batteries.Init();

}

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

{

if (!optionsBuilder.IsConfigured)

{

optionsBuilder.UseSqlite("Data Source=hr.db");

}

}

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

modelBuilder.Entity<EmployeeModel>()

.HasOne(e => e.DepartmentModel)

.WithMany(d => d.Employees)

.HasForeignKey(e => e.DepartmentId);

modelBuilder.Entity<DepartmentModel>().HasData(

new DepartmentModel { Id = 1, Name = "HR", Description = "Отдел найма" },

new DepartmentModel { Id = 2, Name = "IT", Description = "Информационные технологии" },

new DepartmentModel { Id = 3, Name = "Маркетинг", Description = "Отдел маркетинга" },

new DepartmentModel { Id = 4, Name = "Финансы", Description = "Отдел продаж" }

);

modelBuilder.Entity<EmployeeModel>().HasData(

new EmployeeModel { Id = 1, DepartmentId = 1, FullName = "Елена", Position = "Менеджер", IsAvailable = true },

new EmployeeModel { Id = 2, DepartmentId = 2, FullName = "Алексей", Position = "Разработчик", IsAvailable = true },

new EmployeeModel { Id = 3, DepartmentId = 3, FullName = "Мария", Position = "Аналитик", IsAvailable = true }

);

}

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Сотрудник | Отображение сотрудника |

Анализ результатов:

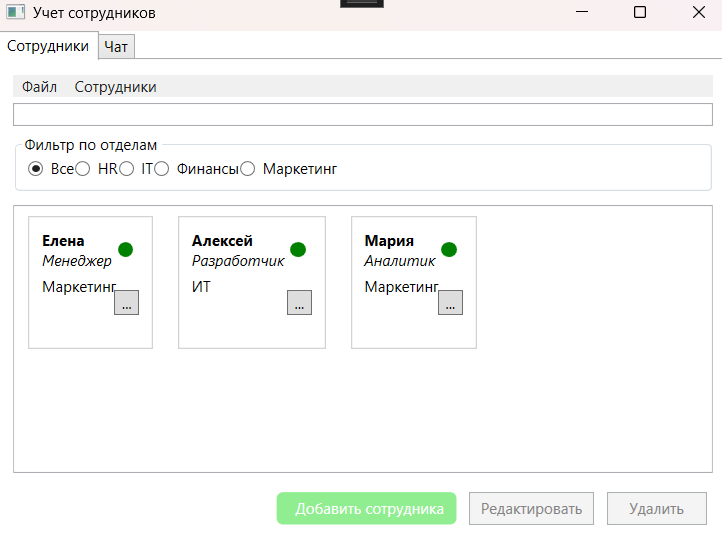


Рисунок 1.1 – Результат работы программы