**11 ПАТТЕРНЫ ПРОЕКТОРИРОВАНИЯ**

Задание 2. Игровой персонаж с дополнительными навыками

Компонент: IGameCharacter

Метод: string GetAbilities() – возвращает перечень навыков.

Конкретный компонент: BaseCharacter – базовый персонаж.

Декораторы:

o StealthDecorator – добавляет способность к скрытности.

o PowerAttackDecorator – добавляет мощную атаку.

o HealingDecorator – добавляет возможность лечения.

Взаимодействие: Игрок выбирает базового персонажа и расширяет его

возможности декораторами.

Листинг программы:

using GNS;

using System;

class Task2 {

static void Main() {

IGameCharacter character = new BaseCharacter();

Console.WriteLine(character.GetAbilities());

character = new StealthDecorator(character);

Console.WriteLine(character.GetAbilities());

character = new PowerAttackDecorator(character);

Console.WriteLine(character.GetAbilities());

character = new HealingDecorator(character);

Console.WriteLine(character.GetAbilities()); }}

Классы BaseCharacter, StealthDecorator, PowerAttackDecorator, HealingDecorator:

namespace GNS{

public class BaseCharacter : IGameCharacter{

public string GetAbilities(){

return "Стандартные способности.";}}

public class StealthDecorator : IGameCharacter{

private readonly IGameCharacter \_character;

public StealthDecorator(IGameCharacter character){

\_character = character;}

public string GetAbilities(){return $"{\_character.GetAbilities()}, Скрытность.";}}

public class PowerAttackDecorator : IGameCharacter{

private readonly IGameCharacter \_character;

public PowerAttackDecorator(IGameCharacter character){

\_character = character;}

public string GetAbilities(){

return $"{\_character.GetAbilities()}, Мощная атака.";}}

public class HealingDecorator : IGameCharacter{

private readonly IGameCharacter \_character;

public HealingDecorator(IGameCharacter character){

\_character = character;}

public string GetAbilities(){

return $"{\_character.GetAbilities()}, Лечение.";}}}

Интерфейс IGameCharacter:

namespace GNS{

public interface IGameCharacter{

string GetAbilities();}}

Таблица 11.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| - | Стандартные способности.  Стандартные способности., Скрытность.  Стандартные способности., Скрытность., Мощная атака.  Стандартные способности., Скрытность., Мощная атака., Лечение. |

Анализ результатов:

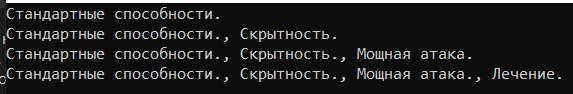


Рисунок 11.1 – Результат работы программы