**Паттерны проектирования**

Задание 1 Реализация паттернов Фабричный метод, Декоратор и Команд. Персонажи игры (маг, воин, лучник). Классы и их взаимодействие: ICharacter – интерфейс с Attack(). Mage, Warrior, Archer – классы персонажей. CharacterFactory – абстрактная фабрика. MageFactory, WarriorFactory, ArcherFactory – конкретные фабрики.

Листинг программы:

public abstract class CharacterFactory

{

public abstract ICharacter CreateCharacter();

}

public class Mage : ICharacter

{

public void Attack()

{

Console.WriteLine("Маг атакует заклинанием!");

}

}

public class Warrior : ICharacter

{

public void Attack()

{

Console.WriteLine("Воин атакует мечом!");

}

}

public class Archer : ICharacter

{

public void Attack()

{

Console.WriteLine("Лучник стреляет из лука!");

}

}

public class MageFactory : CharacterFactory

{

public override ICharacter CreateCharacter()

{

return new Mage();

}

}

public class WarriorFactory : CharacterFactory

{

public override ICharacter CreateCharacter()

{

return new Warrior();

}

}

public class ArcherFactory : CharacterFactory

{

public override ICharacter CreateCharacter()

{

return new Archer();

}

}

public interface ICharacter

{

void Attack();

}

public class Program

{

public static void Main(string[] args)

{

CharacterFactory mageFactory = new MageFactory();

ICharacter mage = mageFactory.CreateCharacter();

mage.Attack();

CharacterFactory warriorFactory = new WarriorFactory();

ICharacter warrior = warriorFactory.CreateCharacter();

warrior.Attack();

CharacterFactory archerFactory = new ArcherFactory();

ICharacter archer = archerFactory.CreateCharacter();

archer.Attack();

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| characterType = "Mage"; character = Mage  characterType = "Warrior"; character = Warrior  characterType = "Archer"; character = Archer | mage.Attack() -> "Маг атакует заклинанием!"  warrior.Attack() -> "Воин атакует мечом!"  archer.Attack() -> "Лучник стреляет из лука!" |

Анализ результатов:

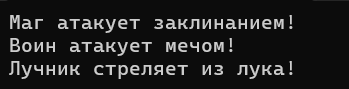


Рисунок 1.1 – Результат работы программы