using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace OOP9

{

public class Program

{

public static void RemoveSeparators(Action<string> action, string a) => action(a);

public static void AddSymbol(Action<string> action, string a) => action(a);

public static void VowelsToUpper(Action<string> action, string a) => action(a);

public static void RemoveExtraSpaces(Action<string> action, string a) => action(a);

public static void SpaceToLowLine(Action<string> action, string a) => action(a);

static void Main(string[] args)

{

Game game = new Game();

Orc orc = new Orc("orc", 150);

Elf elf = new Elf("elf", 100);

game.Attack += new Doing(orc.OnAttack);

game.Heal += new Doing(orc.OnHeal);

game.Attack += new Doing(elf.OnAttack);

game.Heal += new Doing(elf.OnHeal);

game.Go();

////////////////////////////////////

String k = "a?d yn, 2e!2 ;:w22 r2.5";

Console.WriteLine(k);

Action<string> action1, action2, action3, action4, action5;

action1 = (string a) => {

StringBuilder sb = new StringBuilder(a);

for (int i = 0; i < sb.Length; i++)

{

if (sb[i] == '.' || sb[i] == ',' || sb[i] == '!' || sb[i] == '?' || sb[i] == ':' || sb[i] == ';')

{

sb.Remove(i, 1);

i--;

}

}

k = sb.ToString();

};

action2 = (string a) => {

StringBuilder sb = new StringBuilder(a);

sb.Append('+');

k = sb.ToString();

};

action3 = (string a) => {

StringBuilder sb = new StringBuilder(a);

for (int i = 0; i < sb.Length; i++)

{

if (sb[i] == 'a' || sb[i] == 'e' || sb[i] == 'i' || sb[i] == 'o' || sb[i] == 'u' || sb[i] == 'y')

{

sb[i] = sb[i].ToString().ToUpper().ToCharArray()[0];

}

}

k = sb.ToString();

};

action4 = (string a) => {

StringBuilder sb = new StringBuilder(a);

for (int i = 0; i < sb.Length; i++)

{

sb.Replace(" ", " ");

}

k = sb.ToString();

};

action5 = (string a) => {

StringBuilder sb = new StringBuilder(a);

for (int i = 0; i < sb.Length; i++)

{

sb.Replace(' ', '\_');

}

k = sb.ToString();

};

RemoveSeparators(action1, k);

Console.WriteLine(k);

AddSymbol(action2, k);

Console.WriteLine(k);

VowelsToUpper(action3, k);

Console.WriteLine(k);

RemoveExtraSpaces(action4, k);

Console.WriteLine(k);

SpaceToLowLine(action5, k);

Console.WriteLine(k);

}

}

public delegate void Doing();

public class Game

{

public string Name;

public int HP;

public event Doing Attack;

public event Doing Heal;

public void Go()

{

if (Attack != null && Heal != null)

{

Attack();

Heal();

}

else if (Attack != null)

{

Attack();

}

else

{

Heal?.Invoke();

}

}

}

public class Orc : Game

{

public Orc(string name, int HP)

{

Name = name;

this.HP = HP;

}

public void OnAttack()

{

HP -= 40;

Console.WriteLine(Name + " " + HP);

}

public void OnHeal()

{

HP += 20;

Console.WriteLine(Name + " " + HP);

}

}

public class Elf : Game

{

public Elf(string name, int HP)

{

Name = name;

this.HP = HP;

}

public void OnAttack()

{

HP -= 10;

Console.WriteLine(Name + " " + HP);

}

public void OnHeal()

{

HP += 50;

Console.WriteLine(Name + " " + HP);

}

}

}