Pass Task 1 - Spell

Related Learning Outcomes

ULO1 – Explain the OO Principles

This exercise demonstrated object encapsulation, this exercise have been learning about the Solid principle, also the

ULO2 – Use OO Language and Library

Demonstrated class and constructor declaration, the use of conditional statements (use of if else statement, case statement), and assigning values to parameters, pass by referencing.

ULO3 – Design, Develop and Test using an IDE

The code was developed using Xamarin Studio to build and run the program, as well as integrated debugging features to step and inspect values. From the Xamarin studio which gave a clear view of code and the error, the full access to the library, etc.

ULO4 – Communicate using UML Diagrams

interpret a UML class diagram and write the related code, by using UML class diagram and develop a logic to code.

ULO5 – Describe Elements of Good OO Design

The exercise demonstrated correct use of C# coding conventions. OOP uses objects as its fundamental building blocks. Each object is an instance of some class

Screenshots
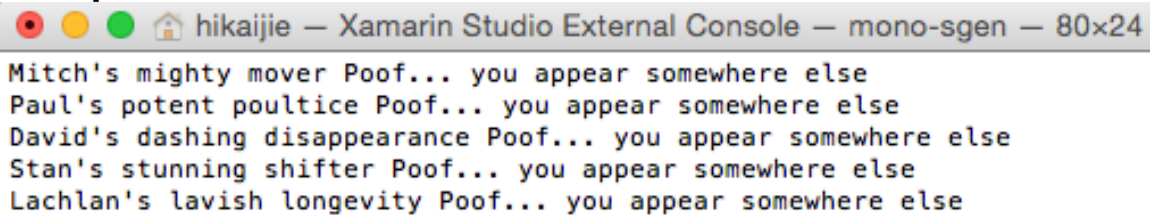
[code running]

```
1 using System;
2
3 namespace SwinwardsSchoolMagic
4 {
5     class MainClass
6     {
7         public static void Main (string[] args)
8         {
9             Spell[] Castspell = new Spell[5];
10            Castspell [0] = new Spell ("Mitch's mighty mover", SpellKind.Teleport);
11            Castspell [1] = new Spell ("Paul's potent poultice", SpellKind.Teleport);
12            Castspell [2] = new Spell ("David's dashing disappearance" , SpellKind.Teleport);
13            Castspell [3] = new Spell ("Stan's stunning shifter", SpellKind.Teleport);
14            Castspell [4] = new Spell ("Lachlan's lavish longevity", SpellKind.Teleport);
15
16            CastAllSpell (Castspell);
17            Console.ReadLine ();
18
19        }
20        public static void CastAllSpell(Spell[] Castspells)
21        {
22            foreach (Spell s in Castspells) {
23                Console.WriteLine (s.Spellname + " " + s.Castspell());
24            }
25        }
26    }
27
28
29 }
30 |
```

```csharp
using System;

namespace SwinwardsSchoolMagic
{
    public class Spell
    {
        private string _name;
        private SpellKind _kind;

        public Spell(string name, SpellKind kind)
        {
            _name = name;
            _kind = kind;
        }

        public string Spellname
        {
            get {return _name;}

        }

        public string Castspell()
        {
            switch (_kind)
            {
            case SpellKind.Teleport:
                return "Poof... you appear somewhere else";
            case SpellKind.Heal:
                return "Ahhh... you feel better";
            case SpellKind.Invisibility:
                return "Zippp... where am I?";
            default:
                return "Invalid";

            }

        }
    }
}
```

```
 1 using System;
 2
 3 namespace SwinwardsSchoolMagic
 4 {
 5     public enum SpellKind
 6     {
 7         Teleport,
 8         Heal,
 9         Invisibility
10     }
11 }
12
13 |
```

[use of IDE]

```
Mitch's mighty mover Poof... you appear somewhere else
Paul's potent poultice Poof... you appear somewhere else
David's dashing disappearance Poof... you appear somewhere else
Stan's stunning shifter Poof... you appear somewhere else
Lachlan's lavish longevity Poof... you appear somewhere else
▮
```