W06 Prove: Polymorphism - Articulate

# What is Polymorphism and why is it important?

## Explain the meaning of Polymorphism

Polymorphism is when a parent class has a function or a method and the child classes have that method’s inner works different amongst themselves.

## Highlight a benefit of Polymorphism

A benefit of Polymorphism is that even though the structure (or perhaps signature) of a method in the class remains the same, the actual behavior of such method can be different for a different circumstance. This then allows the method to be called in the same fashion regardless of the fact that the behavior inside of the method is different for different classes.

## Provide an application of Polymorphism

One application I can think of the use of Polymorphism is in the calculation of volumes for different container shapes. You can have different formulas and different variables provide the volume, but still call the same method using defined as abstract in the parent class and defined as override for each of the child classes.

## Use a code example of Polymorphism from the program you wrote

// the parent class showing the "abstract" keyword included

public abstract class Goal

{

    protected string \_shortName;

    protected string \_description;

    protected int \_points;

    public Goal(string name, string description, int points)

    {

        \_shortName = name;

        \_description = description;

        \_points = points;

    }

    public string GetName()

    {

        return \_shortName;

    }

    public int GetPoints()

    {

        return \_points;

    }

    public abstract int RecordEvent();

    public abstract bool IsComplete();

    public virtual string GetDetailsString()

    {

        string detail = $"{\_shortName} ({\_description})";

        if (IsComplete())

        {

            detail = "[X] " + detail;

        }

        else

        {

            detail = "[ ] " + detail;

        }

        return detail;

    }

    public abstract string GetStringRepresentation();

}

// a child class

public class SimpleGoal : Goal

{

    private bool \_isComplete;

    public SimpleGoal(string name, string description, int points) : base(name, description, points)

    {

        \_isComplete = false;

    }

    public void SetComplete(bool isComplete)

    {

        \_isComplete = isComplete;

    }

    public override int RecordEvent()

    {

        \_isComplete = true;

        return \_points;

    }

    public override bool IsComplete()

    {

        return \_isComplete;

    }

    public override string GetStringRepresentation()

    {

        return $"SimpleGoal:{\_shortName},{\_description},{\_points},{\_isComplete}";

    }

}