CSE 210: Programming with Classes Week 5

# Inheritance

Questions that I hope to answer:

* Explain the meaning of Inheritance
* Highlight a benefit of Inheritance
* Provide an application of Inheritance
* Use a code example of Inheritance from the program you wrote
* Thoroughly explain these concepts

Inheritance is a fundamental principle of object-oriented programming that allows a class to inherit attributes and behaviors from another class This promotes code reusability and enables a hierarchy of related classes. Inheritance helps developers avoid duplicating code, maintain consistency across multiple classes, and extend functionality easily.

For me, the most important benefit of inheritance is code reuse, which allows us to centralize logic in a base class and modify or extend behavior where necessary. This reduces the chance of introducing bugs and makes code easier to maintain.

In the mindfulness app, I applied inheritance by creating a base Activity class that holds shared attributes and methods, like ShowSpinner() and a starting message. Specific activities, such as BreathingActivity and ReflectionActivity, inherit from this class. These child classes reuse and extend the shared functionality but also define their own specific behaviors, like displaying unique prompts or questions.

Example:

| using System; using System.Threading;  public abstract class Activity {  private string name;  private string description;  private int duration;   public Activity(string name, string description)  {  this.name = name;  this.description = description;  }   public void Start()  {  Console.WriteLine($"Starting {name} Activity...");  Console.WriteLine(description);  Console.Write("Enter the duration in seconds: ");  duration = int.Parse(Console.ReadLine());  Console.WriteLine("Prepare to begin...");  ShowSpinner(3);  }   public void End()  {  Console.WriteLine("\nWell done! You've completed the activity.");  Console.WriteLine($"\nYou finished the {name} Activity for {duration} seconds.");  ShowSpinner(3);    LogManager.LogActivity(GetType().Name);   LogManager.DisplayLog();   }   protected int GetDuration() => duration;   protected void ShowSpinner(int seconds)  {   List<string> animationStrings = new List<string>();  animationStrings.Add("|");  animationStrings.Add("/");  animationStrings.Add("-");  animationStrings.Add("\\");  animationStrings.Add("|");  animationStrings.Add("/");  animationStrings.Add("-");  animationStrings.Add("\\");  Console.WriteLine();   foreach (string s in animationStrings)  {   Console.Write(s);  Thread.Sleep(200);  Console.Write("\b \b");  }   Console.WriteLine();  }   public abstract void PerformActivity(); } |
| --- |