Kriston Mendenhall

CSE201

Inheritance

Explain the meaning of Inheritance: inheritance is where one class, the child class, and it’s associated instances, inherit attributes and methods from another class, the parent class. This is accomplished in the child class declaration following the child class name with a colon and then the parent class name.

Highlight the benefit of inheritance: inheritance allows for the centralization of attributes and methods that are the same in two or more classes. This centralization simplifies the codeine process by making it so that changes to the attributes or methods only have to happen in one location, the parent class, rather than having to change these attributes and methods in every class that use uses them. The children classes automatically inherit the changes that were done to the parent class.

Provide an application of inheritance: in our mindfulness activity, the parent class, Activity, contained a method called ShowSpinner that was used by three linked children classes; BreathingActivity, ListeningActivity, and ReflectingActivity. Because these three children classes could use this method, the method did not have to be coded for each child class. Also, if the method ever needed to be changed or updated, it only had to be done in the parent class.

Use a code example of inheritance From the program you wrote: the following is how the child BreathingActivity class is linked to the parent Activity class.

public class BreathingActivity : Activity

{

The next code shows the ShowSpinner metod in the Activity/parent class:

public void ShowSpinner(int seconds)

{

List<string> animationStrings = new List<string>();

{

animationStrings.Add("|");

animationStrings.Add("/");

animationStrings.Add("-");

animationStrings.Add("\\");

}

for (int i = 0; i < seconds; i++)

{

foreach (string s in animationStrings)

{

Console.Write($" {s}");

Thread.Sleep(250);

Console.Write("\b \b");

}

}

Console.WriteLine("");

}

This final piece of code shows the call within the BreathingActivity class to the method within the Activity class:

public void Run()

{

Console.Clear();

DisplayStartingMessage();

Console.Write("How long, in seconds, would you like your session to last? ");

int duration = int.Parse(Console.ReadLine());

SetDuration(duration);

Console.Clear();

Console.WriteLine("Get ready...");

ShowSpinner(3);