COS20007: Object Oriented Programming

Hurdle Task 1: Semester Test

Show Wai Yan/105293041

# FileSystem.cs

namespace Task1

{

public class FileSystem

{

// Field

private List<Thing> \_contents;

// Constructor

public FileSystem()

{

\_contents = new List<Thing> { };

}

// Methods

public void Add(Thing toAdd)

{

\_contents.Add(toAdd);

}

public void PrintContents()

{

Console.WriteLine("This File System contains:");

foreach (Thing item in \_contents)

{

item.Print();

}

}

// Properties

}

}

# Thing.cs

namespace Task1

{

public abstract class Thing

{

// Fields

private string \_name;

// Constructors

public Thing(string name)

{

\_name = name;

}

// Method

public abstract int Size();

public abstract void Print();

// Properties

public string Name

{

get { return \_name; }

}

}

}

# Folder.cs

namespace Task1

{

public class Folder : Thing

{

// Field

private List<Thing> \_contents;

// Constructor

public Folder(string name)

: base(name)

{

\_contents = new List<Thing> { };

}

// Methods

public void Add(Thing toAdd)

{

\_contents.Add(toAdd);

}

public override int Size()

{

int totalSize = 0;

foreach (Thing item in \_contents)

{

totalSize += item.Size();

}

return totalSize;

}

public override void Print()

{

if (\_contents.Count() == 0)

{

Console.WriteLine($"The Folder: '{Name}' is empty!");

return;

}

// Couting folders and files

int foldersCount = 0;

int filesCount = 0;

foreach (Thing item in \_contents)

{

if (item.GetType() == typeof(Folder)) foldersCount++;

else if (item.GetType() == typeof(File)) filesCount++;

}

string folderStatus = foldersCount > 1 ? $"{foldersCount} folders" : $"{foldersCount} folder";

string fileStatus = filesCount > 1 ? $"{filesCount} files" : $"{filesCount} file";

Console.WriteLine(

$"The Folder: '{Name}' contains {folderStatus} and {fileStatus} totalling {Size()} bytes:"

);

foreach (Thing item in \_contents)

{

item.Print();

}

}

}

}

# File.cs

namespace Task1

{

public class File : Thing

{

// Fields

private string \_extension;

private int \_size;

//Constructors

public File(string name, string extension, int size)

: base(name)

{

\_extension = extension;

\_size = size;

}

// Methods

public override int Size()

{

return \_size;

}

public override void Print() {

Console.WriteLine($"File '{Name}.{\_extension}' Size: {Size()} bytes");

}

}

}

# Program.cs

namespace Task1

{

public class Program

{

public static void Main(string[] args)

{

// First 10 prime number

int[] A = { 2, 3, 5, 7, 11, 13, 17, 19, 23, 29 };

// Last four digit of my student id - 3041

int[] B = { A[3], A[0], A[4], A[1] };

// Create File Systems

FileSystem midTest = new FileSystem();

// Adding B[0] files to midTest

string myStudentId = "105293041";

for (int i = 0; i < B[0]; i++)

{

midTest.Add(

new File(

$"{myStudentId}-{i.ToString("D2")}",

"txt",

new Random().Next(1000, 10000)

)

);

}

// Adding a folder that contains B[1] files to midTest

Folder Test1 = new Folder("Test1");

// Add B[1] files to Test1

for (int i = 0; i < B[1]; i++)

{

Test1.Add(

new File(

$"{myStudentId}-{i.ToString("D2")}",

"txt",

new Random().Next(1000, 10000)

)

);

}

midTest.Add(Test1); // Add that folder to midTest;

// Adding a folder that contains a folder that contains B[2] files

Folder Test2 = new Folder("Test2"); // Creating a parent folder

Folder Test2Child = new Folder("Test2Child"); // Creating child folder of Parent

// Add B[2] files to Test2Child

for (int i = 0; i < B[2]; i++)

{

Test2Child.Add(

new File(

$"{myStudentId}-{i.ToString("D2")}",

"txt",

new Random().Next(1000, 10000)

)

);

}

// Add Child folder to parent

Test2.Add(Test2Child);

// Add parent folder to midTest

midTest.Add(Test2);

// Adding a number of B[3] empty folders to the file system

for (int i = 0; i < B[3]; i++)

{

midTest.Add(new Folder($"Test{i + 3}"));

}

midTest.PrintContents();

}

}

}

# UML class diagram

A diagram of a computer

AI-generated content may be incorrect.

# A screenshot of the program output

A screenshot of a computer

AI-generated content may be incorrect.