**Topic 2: Interfaces**

Top of Form

Bottom of Form

**Content**

**Reading**

[Reading](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286730_1)

Interfaces

* + Microsoft:docs.microsoft.com
    - [Interfaces (C# programming guide)](https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/interfaces/)
  + GeeksForGeeks: geeksforgeeks.com
    - [C# | Interface](https://www.geeksforgeeks.org/c-sharp-interface/)
  + TutorialsPoint: tutorialspoint.com
    - [C# - Interfaces](https://www.tutorialspoint.com/csharp/csharp_interfaces.htm)
  + W3Schools: w3schools.com
    - [C# Interface](https://www.w3schools.com/cs/cs_interface.asp)
  + NOTES
    - [interface ICloneable.pdf](https://dmacc.blackboard.com/bbcswebdav/pid-7286738-dt-content-rid-101467801_1/xid-101467801_1) [interface ICloneable.pdf - Alternative Formats](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286730_1)

**C# Interfaces Examples**

[C# Interfaces Examples](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286730_1)

https://youtu.be/LM\_fRixX1v0

using System;

using System.Collections.Generic;

using System.Text;

namespace Module10

{

interface IGame

{

public void play();

}

}

using System;

using System.Collections.Generic;

using System.Text;

namespace Module10

{

class GoFish : IGame

{

public void play()

{

Console.WriteLine("DEAL CARDS!");

}

}

}

using System;

using System.Collections.Generic;

using System.Text;

namespace Module10

{

class DnD : IGame

{

void IGame.play()

{

Console.WriteLine("Develop your character! Roll the dice");

}

}

}

**Using an Interface**

[Using an Interface](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286730_1)

https://youtu.be/Li5wFG6x8fo

How do you USE an interface?

Take a look at why we need to use ICloneable in a Person class.

So how do you use ICloneable?

https://youtu.be/5qJifSkaI0A

namespace Module10

{

public class Person : ICloneable

{

// Properties

private string \_firstName;

private string \_lastName;

public string FirstName { get => \_firstName; set => \_firstName = value; }

public string LastName { get => \_lastName; set => \_lastName = value; }

// Constructors

public Person()

{

FirstName = "unknown";

LastName = "unknown";

}

public Person(string fName, string lName)

{

FirstName = fName;

LastName = lName;

}

// Interface ICloneable requires this method be implemented

public object Clone()

{

Person person = new Person();

person.FirstName = this.FirstName;

person.LastName = this.LastName;

return person;

// OR

// person.FirstName = \_firstName;

// person.LastName = \_lastName;

// return person;

// OR

// return (new Person(this.FirstName, this.LastName));

}

public override string ToString()

{

return LastName + ", " + FirstName;

}

}

}

namespace Module10

{

class PersonDriver

{

static void Main(string[] args)

{

Person instructor = new Person("Michelle", "Ruse");

Person teachingAssistant = new Person("Maria", "Gonzalez");

Console.WriteLine("Created instructor and teaching assistant:");

Console.WriteLine("Original instructor: " + instructor.ToString());

Console.WriteLine("Original teaching assistant: " + teachingAssistant.ToString());

Person newInstructor = new Person();

newInstructor = (Person) instructor.Clone();

Console.WriteLine("\nCreated new instructor with instructor.Clone()");

Console.WriteLine("New instructor: " + newInstructor.ToString());

newInstructor.FirstName = "Michael";

Console.WriteLine("\nChanged new Instructor only:");

Console.WriteLine("New instructor: " + newInstructor.ToString());

Console.WriteLine("Original instructor: " + instructor.ToString());

}

}

}​

[**Use an Interface**](https://dmacc.blackboard.com/webapps/assignment/uploadAssignment?content_id=_7286748_1&course_id=_102593_1&group_id=&mode=view)

Overview: For your Book class from the previous Module, make any corrections needed from instructor feedback add use of the [ICloneable](https://docs.microsoft.com/en-us/dotnet/api/system.icloneable?view=netcore-3.1) interface.

Files to update:

* + Book class (named Book.cs)
    - Update per any feedback from instructor
    - Add use of ICloneable
  + Driver file (which is the file with Main method, often named Program.cs) If your Main method is in Book.cs move to a Program.cs file
    - Create a book object, book1
    - Clone book1 into a second object book2
    - Display both the objects.

This is 5 points

**Abstract Class vs. Interface?**

[Abstract Class vs. Interface?](https://dmacc.blackboard.com/webapps/blackboard/content/listContent.jsp?course_id=_102593_1&content_id=_7286730_1)

What are the main differences you see about Abstract Classes and Interfaces? Discuss it on the weekly discussion board!