|  |  |  |
| --- | --- | --- |
| C-Sharp Language Review | January 1  2021 | |
| (Version 2021.01) Use this quiz to check your understanding of C# code, grammar and terminology. | | Self-Check Quiz |

|  |  |
| --- | --- |
|  | using System; |
|  | using System.Collections.Generic; |
|  | using CSharp.Language.Quiz.Entities; |
|  |  |
|  | namespace CSharp.Language.Quiz |
|  | { |
|  | public class Program |
|  | { |
|  | private static Random rnd = new Random(); |
|  |  |
|  | public static void Main(string[] args) |
|  | { |
|  | Program app = new Program(args); |
|  |  |
|  | app.AssignMarks(30, 80); |
|  |  |
|  | foreach (Student person in app.Students) |
|  | { |
|  | System.Console.WriteLine("Name: " + person.Name); |
|  | foreach (EarnedMark item in person.Marks) |
|  | System.Console.WriteLine("\t" + item); |
|  | } |
|  | } |
|  |  |
|  | private List<Student> \_students = new List<Student>(); |
|  |  |
|  | public List<Student> Students |
|  | { |
|  | get { return \_students; } |
|  | set { \_students = value; } |
|  | } |
|  |  |
|  | public Program(string[] studentNames) |
|  | { |
|  | WeightedMark[] CourseMarks = new WeightedMark[4]; |
|  | CourseMarks[0] = new WeightedMark("Quiz 1", 20); |
|  | CourseMarks[1] = new WeightedMark("Quiz 2", 20); |
|  | CourseMarks[2] = new WeightedMark("Exercises", 25); |
|  | CourseMarks[3] = new WeightedMark("Lab", 35); |
|  | int[] possibleMarks = new int[4] { 25, 50, 12, 35 }; |
|  |  |
|  | foreach (string name in studentNames) |
|  | { |
|  | EarnedMark[] marks = new EarnedMark[4]; |
|  | for (int i = 0; i < possibleMarks.Length; i++) |
|  | marks[i] = new EarnedMark(CourseMarks[i], possibleMarks[i], 0); |
|  | Students.Add(new Student(name, marks)); |
|  | } |
|  | } |
|  |  |
|  | public void AssignMarks(int min, int max) |
|  | { |
|  | foreach (Student person in Students) |
|  | foreach (EarnedMark item in person.Marks) |
|  | item.Earned = (rnd.Next(min, max) / 100.0) \* item.Possible; |
|  | } |
|  | } |
|  | } |
|  |  |
|  | namespace CSharp.Language.Quiz.Entities |
|  | { |
|  | public class Student |
|  | { |
|  | public string Name { get; private set; } |
|  | public EarnedMark[] Marks { get; private set; } |
|  |  |
|  | public Student(string name, EarnedMark[] marks) |
|  | { |
|  | Name = name; |
|  | Marks = marks; |
|  | } |
|  | } |
|  |  |
|  | public class WeightedMark |
|  | { |
|  | public int Weight { get; private set; } |
|  | public string Name { get; private set; } |
|  |  |
|  | public WeightedMark(string name, int weight) |
|  | { |
|  | if (weight <= 0 || weight > 100) |
|  | throw new Exception("Invalid weight: must be greater than zero and at most 100"); |
|  | if (string.IsNullOrEmpty(name) || string.IsNullOrEmpty(name.Trim())) |
|  | throw new Exception("Name cannot be empty for weighted item"); |
|  | Weight = weight; |
|  | Name = name; |
|  | } |
|  | } |
|  |  |
|  | public class EarnedMark : WeightedMark |
|  | { |
|  | public int Possible { get; private set; } |
|  | private double \_Earned; |
|  | public double Earned |
|  | { |
|  | get { return \_Earned; } |
|  | set |
|  | { |
|  | if (value < 0 || value > Possible) |
|  | throw new Exception("Invalid earned mark assigned"); |
|  | \_Earned = value; |
|  | } |
|  | } |
|  |  |
|  | public double Percent |
|  | { get { return (Earned / Possible) \* 100; } } |
|  |  |
|  | public double WeightedPercent |
|  | { get { return Percent \* Weight / 100; } } |
|  |  |
|  | public EarnedMark(WeightedMark markableItem, int possible, double earned) |
|  | : this(markableItem.Name, markableItem.Weight, possible, earned) |
|  | { |
|  | } |
|  |  |
|  | public EarnedMark(string name, int weight, int possible, double earned) |
|  | : base(name, weight) |
|  | { |
|  | if (possible <= 0) |
|  | throw new Exception("Invalid possible marks"); |
|  | Possible = possible; |
|  | Earned = earned; |
|  | } |
|  |  |
|  | public override string ToString() |
|  | { |
|  | return String.Format("{0} ({1})\t - {2}% ({3}/{4}) \t- Weighted Mark {5}%", |
|  | Name, |
|  | Weight, |
|  | Percent, |
|  | Earned, |
|  | Possible, |
|  | WeightedPercent); |
|  | } |
|  | } |
|  | } |

1. Which term best describes the boxed portion of the following code which is copied from line 90 of the attached code?  
    **public class EarnedMark : WeightedMark**



* 1. Constructor



* 1. Base Class



* 1. Interface



* 1. Sub Class



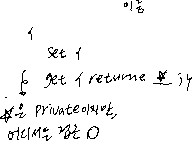
1. Which term best describes the boxed portion of the following code which is copied from line 106 of the attached code?  
    { get { return (Earned / Possible) \* 100; } }



* 1. Local Variable
  2. Field



* 1. Property



* 1. Method Name



1. Which term best describes the boxed portion of the following code which is copied from line 79 of the attached code?  
    public WeightedMark(string name, int weight)



* 1. Local Variable



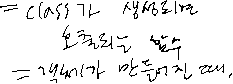
* 1. Field



* 1. Property



* 1. Method Name



* 1. Constructor



1. Which term best describes the boxed portion of the following code which is copied from line 51 of the attached code?  
    public void AssignMarks(int min, int max)
   1. Local Variable
   2. Field



* 1. Property
  2. Method Name



* 1. Parameter

1. Which term best describes the boxed portion of the following code which is copied from line 9 of the attached code?  
    private static Random rnd = new Random();
   1. Local Variable
   2. Field



* 1. Property



* 1. Method Name



* 1. Class



1. Which term best describes the boxed portion of the following code which is copied from line 19 of the attached code?  
    System.Console.WriteLine("Name: " + person.Name);
   1. Class Name



* 1. Fully-Qualified Class Name



* 1. Object Name



* 1. Fully-Qualified Object Name



1. Which term best describes the boxed portion of the following code which is copied from line 13 of the attached code?  
    Program app = new Program(args);
   1. Class Name



* 1. Method Name



* 1. Object Name



* 1. Namespace



1. Which term best describes the boxed portion of the following code which is copied from line 13 of the attached code?  
    Program app = new Program(args);



* 1. Class Name



* 1. Method Name



* 1. Object Name



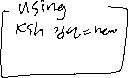
* 1. Namespace



1. Which term best describes the boxed portion of the following code which is copied from line 2 of the attached code?  
   using System.Collections.Generic;



* 1. Class Name



* 1. Method Name



* 1. Object Name



* 1. Namespace



1. Which term best describes the boxed portion of the following code which is copied from line 13 of the attached code?  
    Program app = new Program(args);
   1. Parameter
   2. Return Type



* 1. Argument



* 1. Constructor

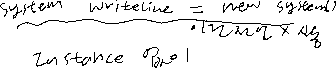
1. Which term best describes the boxed portion of the following code which is copied from line 5 of the attached code?  
   namespace CSharp.Language.Quiz
   1. Class Name
   2. Method Name
   3. Object Name
   4. Namespace



1. Which term best describes the boxed portion of the following code which is copied from line 21 of the attached code?  
    System.Console.WriteLine("\t" + item);
   1. Class Name
   2. Method Name
   3. Object Name
   4. Namespace



1. Which term best describes the boxed portion of the following code which is copied from line 21 of the attached code?  
    System.Console.WriteLine("\t" + item);



* 1. Instance Method Call



* 1. Static Method Call



* 1. Global Method Call



* 1. Constructor Call

1. Which term best describes the boxed portion of the following code which is copied from line 15 of the attached code?  
    app.AssignMarks(30, 80);



* 1. Instance Method Call



* 1. Static Method Call
  2. Global Method Call
  3. Constructor Call

1. Which term best describes the boxed portion of the following code which is copied from line 47 of the attached code?  
    Students.Add(new Student(name, marks));
   1. Instance Method Call
   2. Static Method Call
   3. Global Method Call
   4. Constructor Call



1. Which term best describes the boxed portion of the following code which is copied from line 79 of the attached code?  
    public WeightedMark(string name, int weight)
   1. Argument



* 1. Return Type
  2. Parameter



* 1. Constructor

1. Which term best describes the boxed portion of the following code which is copied from line 51 of the attached code?  
    public void AssignMarks(int min, int max)
   1. Argument
   2. Return Type



* 1. Parameter
  2. Constructor

1. Which term best describes the boxed portion of the following code which is copied from line 125 of the attached code?  
    public override string ToString()
   1. Argument
   2. Return Type



* 1. Parameter
  2. Constructor

1. Which term best describes the boxed portion of the following code which is copied from line 19 of the attached code?  
    System.Console.WriteLine("Name: " + person.Name);
   1. Literal Value
   2. Concatenation

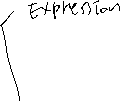
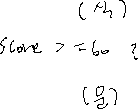


* 1. Parameter
  2. Data Type

1. Which term best describes the boxed portion of the following code which is copied from line 19 of the attached code?  
    System.Console.WriteLine("Name: " + person.Name);
   1. Literal Value
   2. Expression



* 1. Method Call



* 1. Assignment Statement

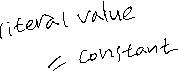


1. Which term best describes the boxed portion of the following code which is copied from line 19 of the attached code?  
    System.Console.WriteLine("Name: " + person.Name);
   1. Call to a Property Getter
   2. Call to a Property Setter
   3. Argument



* 1. Return Type

1. Which term best describes the boxed portion of the following code which is copied from line 36 of the attached code?  
    CourseMarks[0] = new WeightedMark("Quiz 1", 20);
   1. Data Type
   2. Variable



* 1. Literal Value



* 1. Expression

1. Which term best describes the boxed portion of the following code which is copied from line 36 of the attached code?  
    CourseMarks[0] = new WeightedMark("Quiz 1", 20);
   1. Data Type
   2. Variable
   3. Literal Value



* 1. Expression

1. Which term best describes the boxed portion of the following code which is copied from line 51 of the attached code?  
    public void AssignMarks(int min, int max)
   1. Data Type



* 1. Variable
  2. Literal Value



* 1. Expression

1. Which term best describes the boxed portion of the following code which is copied from line 79 of the attached code?  
    public WeightedMark(string name, int weight)
   1. Data Type
   2. Variable



* 1. Literal Value
  2. Expression

1. Which term best describes the boxed portion of the following code which is copied from line 109 of the attached code?  
    { get { return Percent \* Weight / 100; } }
   1. Data Type
   2. Variable
   3. Literal Value
   4. Expression



1. Which term best describes the boxed portion of the following code which is copied from line 35 of the attached code?  
    WeightedMark[] CourseMarks = new WeightedMark[4];
   1. Size of Array



* 1. Position in Array
  2. Array Variable
  3. Array Element

1. Which term best describes the boxed portion of the following code which is copied from line 37 of the attached code?  
    CourseMarks[1] = new WeightedMark("Quiz 2", 20);
   1. Size of Array
   2. Position in Array



* 1. Array Variable
  2. Array Element

1. Which term best describes the boxed portion of the following code which is copied from line 37 of the attached code?  
    CourseMarks[1] = new WeightedMark("Quiz 2", 20);
   1. Size of Array
   2. Position in Array
   3. Array Variable



* 1. Array Element

1. Which term best describes the boxed portion of the following code which is copied from line 36 of the attached code?  
    CourseMarks[0] = new WeightedMark("Quiz 1", 20);
   1. Size of Array
   2. Position in Array
   3. Array Variable
   4. Array Element



1. Which term best describes the boxed portion of the following code which is copied from line 40 of the attached code?  
    int[] possibleMarks = new int[4] { 25, 50, 12, 35 };
   1. Expression
   2. Array Initializer



* 1. Value Body
  2. Placeholder



1. Which term best describes line 64 of the attached code?
   1. Auto-Implemented Property



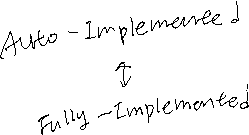
* 1. Fully-Implemented Property
  2. Field
  3. Constructor
  4. Method

1. Which term best describes lines 67-72 of the attached code?
   1. Auto-Implemented Property
   2. Fully-Implemented Property
   3. Field
   4. Constructor



* 1. Method

1. Which term best describes lines 105-106 of the attached code?



* 1. Auto-Implemented Property
  2. Fully-Implemented Property



* 1. Field
  2. Constructor
  3. Method

1. Which term best describes lines 94-103 of the attached code?
   1. Auto-Implemented Property
   2. Fully-Implemented Property

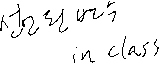


* 1. Field
  2. Constructor
  3. Method

1. Which term best describes line 93 of the attached code?
   1. Auto-Implemented Property



* 1. Fully-Implemented Property



* 1. Field



* 1. Constructor
  2. Method

1. Which term best describes lines 125-134 of the attached code?
   1. Auto-Implemented Property
   2. Fully-Implemented Property
   3. Field
   4. Constructor
   5. Method

