**C# Quizzes**

Quiz 1:

Things to concentrate on.

1. Object and its functions.
2. Variable declarations, primitive types and var.
3. Null coalescing operator, as operator.
4. Expressions, operators and operator precedence.
5. If and while statements.
6. String formatting and string interpolation.
7. What the .NET Framework and CLR are.
8. Strings and the common string functions.

Quiz 2:

Things to focus on for quiz 2.

1. Difference between field, property and method and when to use each one
2. Purpose of a constructor
3. Purpose of constructor chaining
4. Accessibility - public, private, internal, protected
5. Declaring properties using various formats - full, auto and with mixed accessibility
6. Purpose of inheritance and how to reference base type
7. Purpose of *base* and *this* keywords
8. What a Control is and some common examples (e.g. TextBox, Button, CheckBox, ComboBox, ListBox)
9. Core properties of Controls - Name, Text
10. Lifetime of a form - Constructor, OnLoad, OnFormClosing, OnFormClosed
11. Core methods of a Form - ShowDialog, Show, Close, OnLoad
12. Events to handle for common controls - Click, TextChanged
13. How to get the results of ShowDialog
14. Purpose of MessageBox and how to show it

**Quiz 3**

Posted on: Monday, October 29, 2018 7:29:11 PM CDT

1. Purpose of List<T>.
2. Working with List<T> including enumeration, Count, Add, Remove and Clear.
3. What a generic type is and how you would use a generic type (e.g. List<T>).
4. Object and collection initializer syntax
5. Purpose of an interface.
6. How to implement an interface in a class (excludes explicit interface implementations).
7. Purpose of an abstract class.
8. Purpose of a static class and how to implement one.
9. Purpose of extension methods and how to declare them.
10. Purpose of IValidatableObject and IEnumerable<T> interfaces.
11. Common delegates like Func a nd Action.
12. Purpose of LINQ.
13. Basic LINQ syntax (from...where...select).
14. Some common LINQ extension methods like Where, FirstOrDefault, ToArray, ToList and Count().