C# 04 Generics

1. Describe the problem generics address.
    - solve the problem of having to use loosely typed objects
    - also enable types (classes and interfaces) to be parameters when defining classes, interfaces and methods
    - let the data type to be later determined at run time
2. How would you create a list of strings, using the generic List class?
    - List<T> newTList= new List<T>()
3. How many generic type parameters does the Dictionary class have?
    - two, Dictionary<TKey,TValue>
4. False. When a generic class has multiple type parameters, they must all match.
5. What method is used to add items to a List object?
    - Insert()
6. Name two methods that cause items to be removed from a List.
    - List<T>RemoveAt()
    - List<T>Remove()
7. How do you indicate that a class has a generic type parameter
    - add <T> after class name and may add where T: Class for example to specify the data type can be used for generics type
8. False. Generic classes can only have one generic type parameter.
9. True. Generic type constraints limit what can be used for the generic type.
10. True. Constraints let you use the methods of the thing you are constraining to