1. Describe the problem generics address.
   - Allow us to design classes and methods but defer the specification of types until the class or method is declared and called

2. How would you create a list of strings, using the generic List class?
   - List<T> list = new List<T>();

3. How many generic type parameters does the Dictionary class have?
   - Two parameters. One for key and one for value

4. True/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?
   - List.Add()
   - List.Insert()
   - List.AddRange()

6. Name two methods that cause items to be removed from a List.
   - List.Remove(): remove object
   - List.RemoveAt(): use index to remove

7. How do you indicate that a class has a generic type parameter?
   - Class ClassName<T>

8. True/False. Generic classes can only have one generic type parameter.
9. True/False. Generic type constraints limit what can be used for the generic type.
10. True/False. Constraints let you use the methods of the thing you are constraining to.