

1. Describe the problem generics address.

Generics address the problem of type safety and code reuse in C# by allowing developers to create classes and methods that can work with different data types without sacrificing type safety.

2. How would you create a list of strings, using the generic List class?

```
List<string> stringList = new List<string>();
```

3. How many generic type parameters does the Dictionary class have?

two generic type parameters: TKey and TValue.

4. True/False. When a generic class has multiple type parameters, they must all match.

YES

5. What method is used to add items to a List object?

Add()

6. Name two methods that cause items to be removed from a List.

Remove(), RemoveAt().

7. How do you indicate that a class has a generic type parameter?

```
class MyGenericClass<T>
{
    body
}
```

8. True/False. Generic classes can only have one generic type parameter.

False

9. True/False. Generic type constraints limit what can be used for the generic type.

True

10. True/False. Constraints let you use the methods of the thing you are constraining
To

True