1. Describe the problem generics address.

Generics address the problem of code redundancy and type safety. They allow you to create classes, methods, and data structures that can operate on any data type without the need to write multiple versions for different types. This reduces code duplication and the risk of runtime errors due to type mismatches.

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

List<string> MyStringList = new List<string>();

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

The Dictionary class has two generic type parameters: ‘TKey’ and ‘TValue’.

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

false

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

The Add method is used to add items to a List object.

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

Remove and RemoveAt are two methods that cause items to be removed from a List.

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

You indicate that a class has a generic type parameter by using angle brackets (<>) after the class name, specifying the type parameter inside the brackets.

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