1. // what is the purpose or effect of the "@" prefix

2. // before the string literal assigned to stringTwo?

3.

4. var stringOne = "abc";

5. var stringTwo = @"abc”;

6.

7. // Bonus: What is the effect of the "@" prefix in a variable name?

8.

9. var @stringThree = "def";

1. // What values will be assigned to c and d?

2.

3. int a = 1;

4. int b = 4;

5.

6. int c = a | b;

7.

8. int d = a & b;

1. // Suppose the following two fields in a class used in a shared library.

2. // What, if anything, are the differences between A and B?

3.

4. public const int A = 1;

5. public readonly int B = 1;

1. // What, if anything, is the difference between the two following

2. // error handing implementations?

3.

4.

5. // Implementation #1:

6.

7. try { /\* … \*/ }

8. catch (Exception caughtException)

9. {

10. throw caughtException;

11. }

12.

13.

14. // Implementation #2:

15.

16. try { /\* … \*/ }

17. catch (Exception caughtException)

18. {

19. throw;

20. }

1. // Assume we have a public Task DoSomethingAsync(Uri uri) method.

2. // This method performs a network operation.

4. // It is expected that the method takes some time to complete.

5. // What are some changes you might suggest be made to this code?

6.

7.

8. public async void DoOneThingAsync()

9. {

10. DoSomethingAsync(new Uri("http://some.url"));

11. }

12.

13.

14. public async Task DoTwoThingsAsync()

15. {

16. await DoSomethingAsync(new Uri("http://first.url"));

17. await DoSomethingAsync(new Uri("http://second.url"));

18. }

19.

20.

21. public async Task DoManyThingsAsync(IEnumerable<Uri> uris)

22. {

23. foreach (var uri in uris)

24. {

25. await DoSomethingAsync(uri);

26. }

27. }