1. D

2. BD

3. A

4.

```
private static decimal CalculateInterest(decimal loanAmount, int loanTerm, decimal loanRat
{
    decimal interestAmount = loanAmount * loanRate * loanTerm;
    #if DEBUG
    LogLine("Interest Amount : ", interestAmount.ToString("c"));
    #endif
    return interestAmount;
}
public static void LogLine(string message, string detail)
{
    Console.WriteLine("Log: {0} = {1}", message, detail);
}
```

5. DE

6. A

7. –

8. A

9.

Create a **CounterCreationDataCollection** collection. Then create the counters as **CounterCreationData** objects and set the necessary properties.

Add the **CounterCreationData** objects to the collection by calling the **Add()** method of the collection.

Call the **Create()** method of the **PerformanceCounterCategory** class and pass the collection to the method.

10. –

11. C

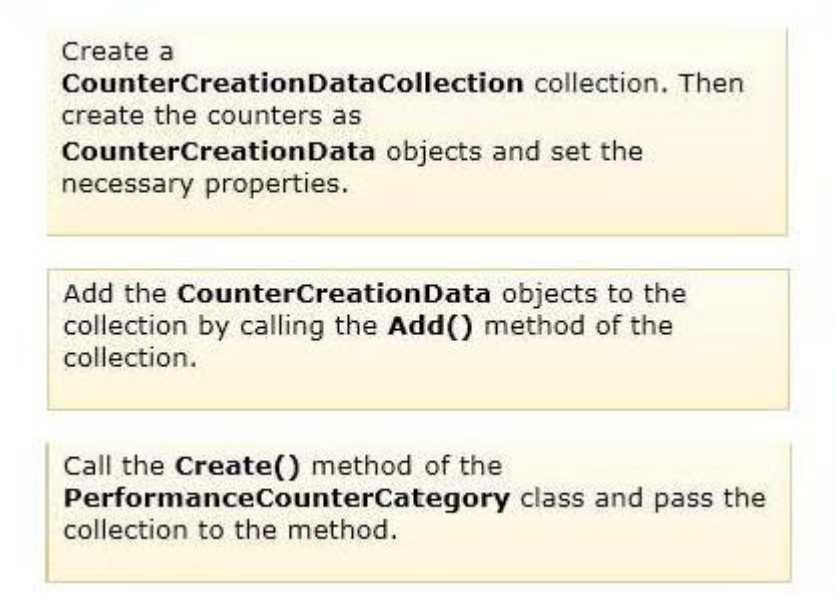
12. BC

13. B

14. D

15. A

16.



17. D

18. –

19.

```
var countersDC = new [ CounterCreationData(); | CounterCreationDataCollection(); | PerformanceCounter(); ]

var IOTDateRate = new [ CounterCreationData(); | CounterCreationDataCollection(); | PerformanceCounter(); ]

IOTDateRate.CounterName = "Data Trans/Sec";

IOTDateRate.CounterHelp = "Data transactions per second";

IOTDateRate.CounterType = PerformanceCounterType.[ CountPerTimeInterval64; | NumberOfItems64; | RateOfCountsPerSecond64; ]

countersDC.Add(IOTDateRate);

PerformanceCounterCategory.Create("Application1", "Application1 category for
IOT data", PerformanceCounterCategoryType.SingleInstance, countersDC);
```

20. C

21. D

22. B

23. BC

24. D

25. A

26. D

27. B

28. D

29. D

30. A

31. B

32. B

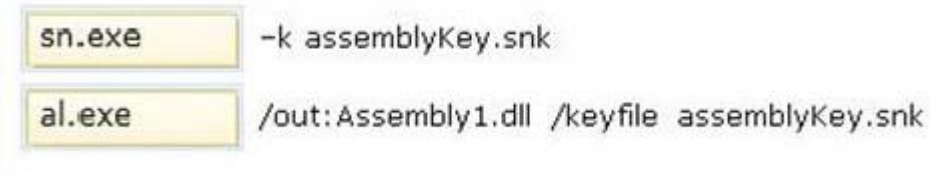
33. B

34. C

35. B

36. C

37.



38. C

39. D

40. D

41. B

42. A

43. A

43(2). C

44. C

45. A

46. B

47.