1.

```
public class LoanCollection : IEnumerable
{
    private readonly Loan[] _loanCollection;
    public LoanCollection(Loan[] loanArray)
    {
        _loanCollection = new Loan[loanArray.Length];

        for (int i = 0; i < loanArray.Length; i++)
        {
            _loanCollection[i] = loanArray[i];
        }
    }

    public IEnumerator GetEnumerator()
    {
        return _loanCollection.GetEnumerator();
    }
}
```

2. D

3. C

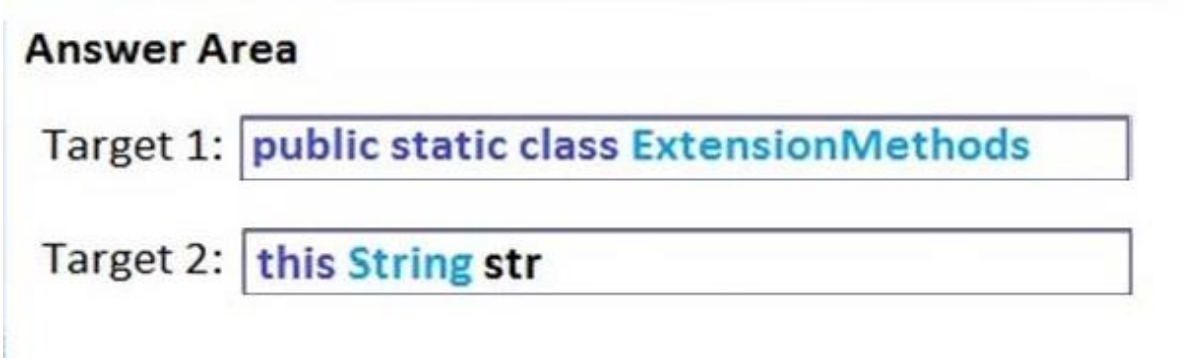
4.

```
public static class ExtensionMethods
{
    public static bool IsUrl(
        this String str
        )
    {
        var regex = new Regex(
            "(https?://)?([A-Za-z9-0-]*\\.)?([A-Za-z0-9-]*)" +
            "\\.[A-Za-z0-9]*/?.*");
        return regex.IsMatch(str);
    }
}
```

5. B

6. BD

7.



8. A

9. D

10.

```
string response;
[ ] (letter)
    case
    if
    switch
{
    [ ] 'a':
        case
        default
        else
        if
        response = "animal";
        break;
    [ ] 'm':
        case
        default
        else
        if
        response = "mineral";
        break;
    [ ] :
        case
        default
        else
        if
```

11.

12.

Target 1:

```
: IEnumerable
```

Target 2:

```
public IEnumerator
GetEnumerator()
```

Target 3:

```
return _loanCollection.
GetEnumerator();
```

13. D

14. C

15. D

16.

```
var sb = new StringBuilder();
sb.Append("First Line");
sb.AppendLine();
sb.Append("Second Line");
```

17. AC

18.

```
public static class ExtensionMethods
{
    public static bool IsEmail(
        this String str
        )
    {
        var regex = new Regex(@"^([\w\.\-]+)@([\w\-]+)((\.(\w){2,3})+)$");
        return regex.IsMatch(str);
    }
}
```

19. D

20. B

21. D

22. B

23. D

24. EF

25.

```
if (other == null) return false;

if (this.ID != other.ID) return false;

if (!Object.Equals
(this.Name, other.Name)) return false;
```

26.

	Yes	No
A user can be a member of more than one of the groups.	○	○
If the user belongs to only the Administrators group, the following code will return a value of true: user.UserGroup == Group.Administrators	○	○
If the user belongs to only the Supervisors group, the following code will return a value of true: user.UserGroup lt; Group.Administrators	○	○

27. B

28.

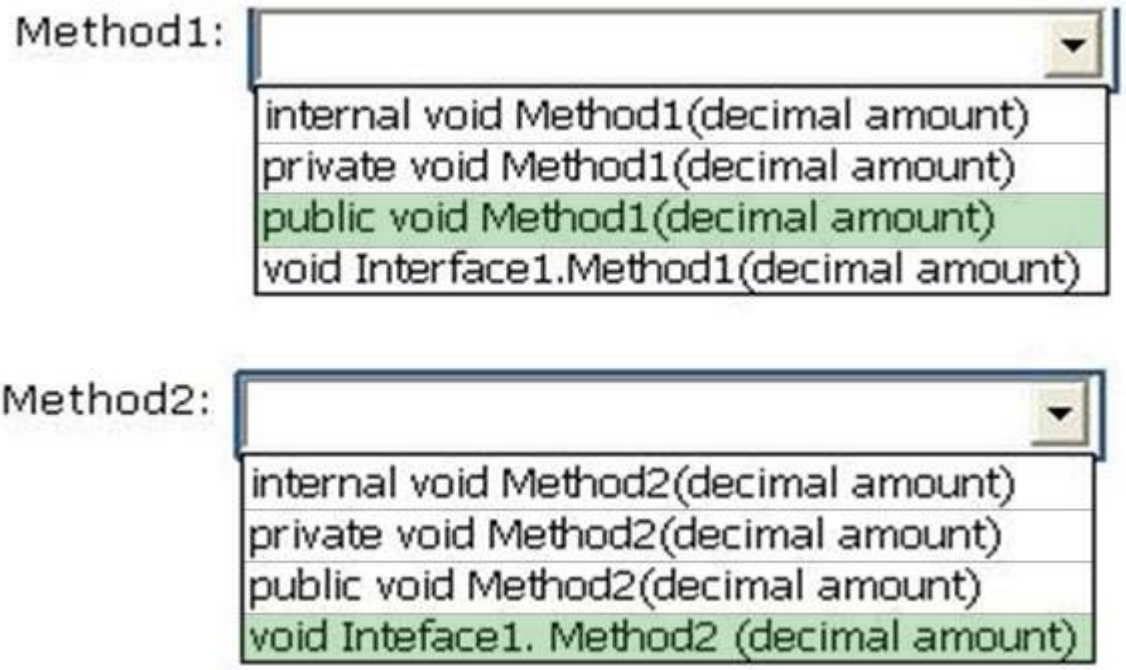
```
output = string.Format("Temperature at [ ] on [ ] : [ ] ", date, temp);
```

{0:t}	{0:d}	{0}
{1:t}	{1:d}	{1}
{0:hh:mm}	{1:dd/mm/yyyy}	{0:N2}
{1:hh:mm}	{0:mm/dd/yyyy}	{1:N2}

29. D

30. B

31.



32. A

33. C

34. A

35. C

36. BC

37. A

38.

	Yes	No
If dbDataType is DateTime, today's date is returned.	○	●
If dbDatatype is Int64, Null is returned.	●	○
If dbDatatype is Double, 0 is returned.	○	●

39.

	Yes	No
All of the objects derived from MyCustomerClass have CustomerID as a property.	○	●
All of the objects derived from MyCustomerClass have CompanyName as a property.	●	○
All of the objects derived from MyCustomerClass have State as a property.	●	○

40. C

41. D

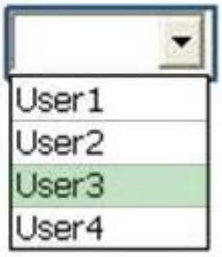
42.

```
BaseLogger logger = new Logger();

logger.Log("Log started");

logger.Log("Base: Log continuing");

logger.LogCompleted();
```

43.

Line 07 of the method will display ...

Line 09 of the method will display ...



44. AC

45. D

46.

```
Int32 returnStatus = Int32.MinValue;
switch (status) {
    case "Active":
        returnStatus = 1;
        break;
    case "Inactive"
        returnStatus = 0;
        break;
    default:
        returnStatus = -1;
        break;
}
return returnStatus;
```

47. CD

48. C

49. B

50. A