

1. Dado o seguinte código, o que será impresso no console? Explique sua resposta.

```
var lista = new List { 1, 2, 3, 4, 5 };  
var resultado = lista.Where(x => x % 2 == 0).Select(x => x * 2);  
Console.WriteLine(string.Join(', ', resultado));
```

Resposta:

Primeiro a lista é filtrada por números que sejam múltiplos de 2, pois usa divisão por 2, possui resto 0. Após isso é multiplicado por 2 cada elemento da que é múltiplo por 2. Logo o resultado é “4,8”.

2. Se você rodar a query abaixo no banco de dados, o que acontecerá?

```
SELECT * FROM Usuarios WHERE Nome LIKE '%_Silva%'
```

Resposta:

O resultado da consulta será usuários que contém o nome “Silva” no nome e ignora nomes que começam com “Silva”, pois o caractere “_” pede uma caractere antes do “Silva”. Ou seja, ele vai pegar todos que tenha sobrenome com “Silva”.

Exemplo: João da Silva, João Silva, João Silvania.

3. Considere o código abaixo. Qual será a saída e por quê?

```
public class Exemplo  
{  
    public string Nome { get; set; }  
    public Exemplo(string nome)  
    {  
        Nome = nome ?? "Sem Nome";  
    }  
}  
  
var obj = new Exemplo(null);  
Console.WriteLine(obj.Nome);
```

Resposta:

Será exibido “Sem Nome”. Porque, foi passado nulo para o nome e o operador ?? verificar retorna o valor a esquerda se não nulo, ou caso nulo o valor da direita. Ou seja, mesmo se passado nulo o Nome vai ter um valor diferente de nulo.

4. O que está errado no código abaixo e como corrigir?

```
public async Task<List<Usuario>> BuscarUsuarios()
{
    using (var db = new MeuDbContext())
    {
        return db.Usuarios.ToListAsync();
    }
}
```

Resposta:

O erro está na falta do “await” antes do “db.Usuarios.ToListAsync();”. Porque está usando “ToListAsync”.