

## Valide a classe Produto

Você está desenvolvendo um aplicativo C# que inclui uma classe chamada `Produto`. O seguinte segmento de código define a classe `Produto`:

```
class Produto
{
    public int Id { get; set; }
    public int CategoriaId { get; set; }
    public string Nome { get; set; }
    public bool EhValido { get; set; }
}
```

Você implementa a interface `System.ComponentModel.DataAnnotations.IValidatableObject` para fornecer uma maneira de validar o objeto `Produto`.

O objeto `Product` possui os seguintes requisitos:

- A propriedade `Id` deve ter um valor maior que zero.
- A propriedade `Nome` deve ter um valor diferente de vazio ou nulo.

Você precisa validar o objeto `Produto`. Qual segmento de código você deve usar?

Selecione uma alternativa

**A**

```
public IEnumerable Validate(ValidationContext validationContext)
{
    if (Id <= 0)
        yield return new ValidationResult("Id do produto é obrigatório.", new[] { "Id" });
    if (string.IsNullOrEmpty(Nome))
        yield return new ValidationResult("Nome do produto é obrigatório.", new[] { "Nome" });
}
```

**B**

```
public bool Equals(Produto produtoAValidar)
{
    produtoAValidar.EhValido = produtoAValidar.Id > 0 || !string.IsNullOrEmpty(produtoAValidar.Nome);
    return produtoAValidar.EhValido;
}
```

**C**

```
public ValidationResult Validate()
{
    ValidationResult validationResult = null;
    if (Id <= 0)
    {
        validationResult = new ValidationResult("Id do produto é obrigatório.");
    }
}
```

```
if (string.IsNullOrEmpty(Nome))
{
    validationResult = new ValidationResult("Nome do produto é obrigatório.");
}
return validationResult;
}
```

**D**

```
public bool Validate()
{
    EhValido = Id > 0 || !string.IsNullOrEmpty(Nome);
    return EhValido;
}
```