

Definindo Comandos

O seguinte trecho de código XAML contém um botão para realizar uma compra num aplicativo de e-commerce:

```
<Button x:Name="buttonComprar" Text="Comprar"
Command="{Binding ComprarCommand}"></Button>
```

Você deve implementar no View Model desse aplicativo a configuração desse comando (`ComprarCommand`), fazendo uma chamada a um método de uma biblioteca externa: `ECommerce.SubmeterCompra(DadosDaCompra)` .

Marque a alternativa com o trecho de código C# que representa a implementação correta do comando `ComprarCommand` .

Selecione uma alternativa

A

```
public DadosDaCompra DadosDaCompra { get; set; }
public ICommand ComprarCommand { get; set; }
public ECommerceViewModel()
{
    ComprarCommand = ()=>
    {
        ECommerce.SubmeterCompra(DadosDaCompra);
    };
}
```

B

```
public DadosDaCompra DadosDaCompra { get; set; }
public ICommand ComprarCommand { get; set; }
public ECommerceViewModel()
{
    ComprarCommand = ECommerce.SubmeterCompra;
}
```

C

```
public DadosDaCompra DadosDaCompra { get; set; }
public ICommand ComprarCommand { get; set; }
public ECommerceViewModel()
{
    ComprarCommand = new Command(() =>
    {
        ECommerce.SubmeterCompra(DadosDaCompra);
    });
}
```

D

```
public DadosDaCompra DadosDaCompra { get; set; }
public ICommand ComprarCommand { get; set; }
public ECommerceViewModel()
{
```

```
ComprarCommand = ECommerce.SubmeterCompra(DadosDaCompra);  
}
```

E

```
public DadosDaCompra DadosDaCompra { get; set; }  
public ICommand ComprarCommand { get; set; }  
public ECommerceViewModel()  
{  
    ComprarCommand = new Command(ECommerce.SubmeterCompra(DadosDaCompra));  
}
```