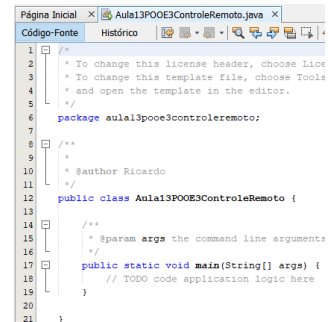
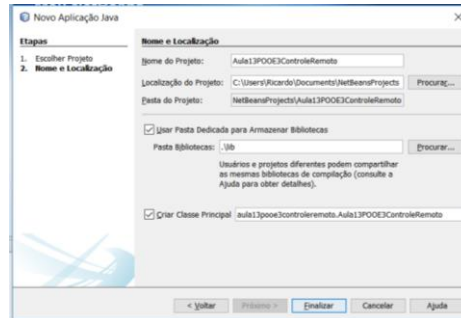
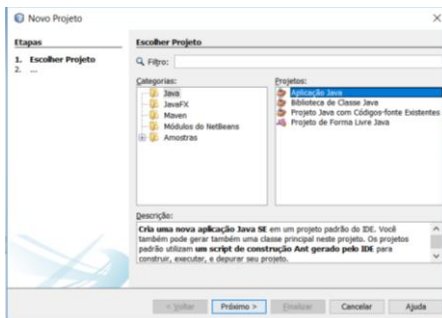


Encapsulamento - Controle Remoto

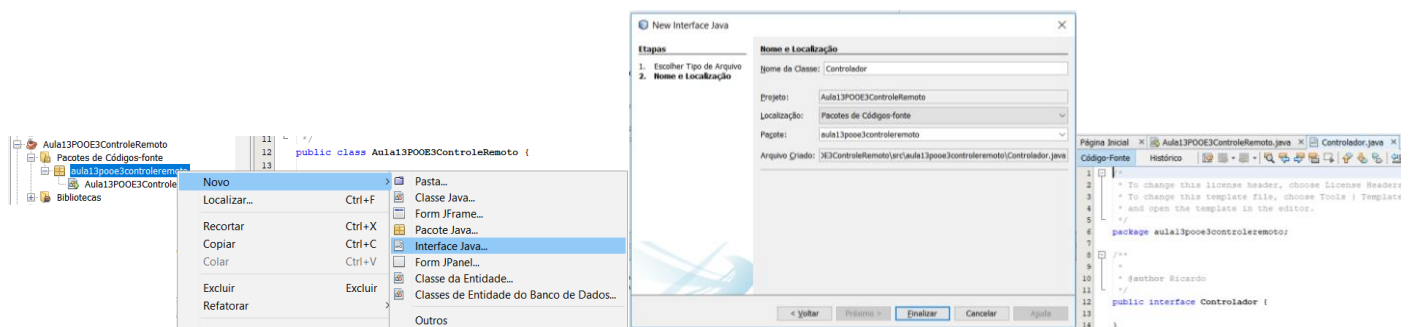
Nesta tarefa, estaremos criando uma aplicação com as funções de um Controle Remoto.



Abra o NetBeans e inicie um Novo Projeto Java/Aplicação Java.



Crie uma Interface. Esta facilitará a comunicação do usuário com as funcionalidades do controle remoto.



Insira o código da Interface.

```

<<interface>>
Controlador

+ ligar()
+ desligar()
+ abrirMenu()
+ fecharMenu()
+ maisVolume()
+ menosVolume()
+ ligarMudo()
+ desligarMudo()
+ play()
+ pause()

interface Controlador
// Métodos Abstratos
public abstrato Metodo ligar()
public abstrato Metodo desligar()
public abstrato Metodo abrirMenu()
public abstrato Metodo fecharMenu()
public abstrato Metodo maisVolume()
public abstrato Metodo menosVolume()
public abstrato Metodo ligarMudo()
public abstrato Metodo desligarMudo()
public abstrato Metodo play()
public abstrato Metodo pause()
FimInterface

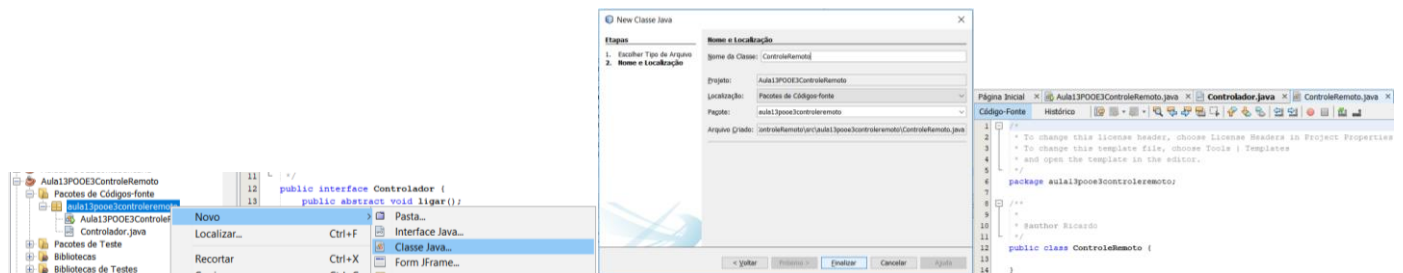
```

```

12 public interface Controlador {
13     public abstract void ligar();
14     public abstract void desligar();
15     public abstract void abrirMenu();
16     public abstract void fecharMenu();
17     public abstract void maisVolume();
18     public abstract void menosVolume();
19     public abstract void ligarMudo();
20     public abstract void desligarMundo();
21     public abstract void play();
22     public abstract void pause();
23 }

```

Crie a Classe “ControleRemoto”.



Defina os Atributos.

```

<<interface>>
Controlador

classe ControleRemoto

// Atributos
privado inteiro volume
privado logico ligado
privado logico tocando

```

```

1  /*
2   * To change this license header, c
3   * To change this template file, c
4   * and open the template in the edi
5   */
6  package aula13pooe3controleremoto;
7
8  /**
9   *
10  * @author Ricardo
11  */
12  public class ControleRemoto {
13
14      //Atributos
15      private int volume;
16      private boolean ligado;
17      private boolean tocando;
18  }

```

Terminado os Atributos, iniciaremos a construção dos Métodos Especiais, sendo: o Construtor e os Getters e Setters.



Desenvolva o Construtor clicando com o botão direito do mouse => Inserir Código => Construtor.

The screenshot shows an IDE with the **ControleRemoto** class open. A right-click context menu is displayed over the class, with the path **Inserir Código... > Construtor...** highlighted. The code for the constructor is being inserted into the class:

```
public class ControleRemoto {
    //Atributos
    private int volume;
    private boolean ligado;
    private boolean tocando;

    //Métodos Especiais

    public ControleRemoto(int volume, boolean ligado, boolean tocando) {
        this.volume = volume;
        this.ligado = ligado;
        this.tocando = tocando;
    }
}
```

Edite o código do Construtor, conforme o planejamento.

The screenshot shows the **ControleRemoto** class with the constructor code being edited. The code is as follows:

```
// Métodos Especiais
public Metodo Construtor()
    volume = 50
    ligado = falso
    tocando = falso
FimMetodo

//Métodos Especiais
public ControleRemoto() {
    this.volume = 50;
    this.ligado = false;
    this.tocando = false;
}
```

Vamos incluir os Métodos Getters e Setters de todos.

Clique com o botão direito do mouse => Inserir Código => Getters e Setters, gerando código para todos.

The screenshot shows the **ControleRemoto** class with the **Gerar Getters e Setters** dialog box open. The dialog box lists the fields `ligado : boolean`, `tocando : boolean`, and `volume : int`. The **Gerar** button is highlighted. The code for the getters and setters is being generated into the class:

```
public ControleRemoto() {
    this.volume = 50;
    this.ligado = false;
    this.tocando = false;
}

//Métodos Especiais
public boolean getLigado() {
    return ligado;
}

public void setLigado(boolean ligado) {
    this.ligado = ligado;
}

public boolean getTocando() {
    return tocando;
}

public void setTocando(boolean tocando) {
    this.tocando = tocando;
}

//Métodos Especiais
public int getVolume() {
    return volume;
}

public void setVolume(int volume) {
    this.volume = volume;
}
FimMetodo
```

Compare o código com o que estava planejado.

<<interface>> Controlador	ControleRemoto
<pre>classe ControleRemoto // Atributos privado inteiro volume privado logico ligado privado logico tocando // Métodos Especiais public Metodo Construtor() volume = 50 ligado = falso tocando = falso FimMetodo privado Metodo getVolume() retorne volume FimMetodo privado Metodo getLigado() retorne volume FimMetodo privado Metodo getTocando() retorne volume FimMetodo privado Metodo setVolume(v: Inteiro) volume = v FimMetodo privado Metodo setLigado(l: logico) ligado = l FimMetodo privado Metodo setTocando(t: logico) tocando = t FimMetodo FimClasse</pre>	<pre>- volume - ligado - tocando + ligar() + desligar() + abrirMenu() + fecharMenu() + maisVolume() + menosVolume() + ligarMudo() + desligarMudo() + play() + pause() - setVolume() - getVolume() - setLigado() - getLigado() - setTocando() - getTocando()</pre>

```
19 //Métodos Especiais
20
21 public ControleRemoto() {
22     this.volume = 50;
23     this.ligado = false;
24     this.tocando = false;
25 }
26
27 public int getVolume() {
28     return volume;
29 }
30
31 public void setVolume(int volume) {
32     this.volume = volume;
33 }
34
35 public boolean isLigado() {
36     return ligado;
37 }
38
39 public void setLigado(boolean ligado) {
40     this.ligado = ligado;
41 }
42
43 public boolean isTocando() {
44     return tocando;
45 }
46
47 public void setTocando(boolean tocando) {
48     this.tocando = tocando;
49 }
```

Conforme o exercício anterior, as variáveis do Tipo Boolean recebem o código “is” ao invés de “get”. Como o exercício trata dos Métodos Getters e Setters, edite os códigos.

```
35 public boolean isLigado() {
36     return ligado;
37 }
38
39 public void setLigado(boolean ligado) {
40     this.ligado = ligado;
41 }
42
43 public boolean isTocando() {
44     return tocando;
45 }
```

```
35 privado Metodo getLigado()
36     retorne volume
37 FimMetodo
38 privado Metodo getTocando()
39     retorne volume
40 FimMetodo
```

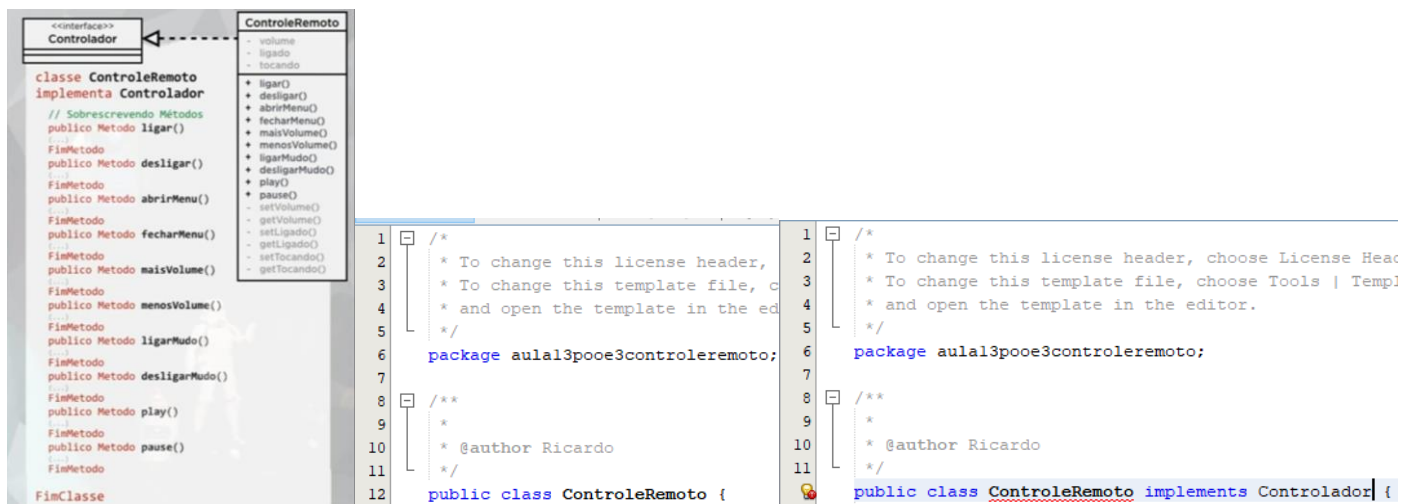
```
35 public boolean getLigado() { //is para get
36     return ligado;
37 }
38
39 public void setLigado(boolean ligado) {
40     this.ligado = ligado;
41 }
42
43 public boolean getTocando() { //is para get
44     return tocando;
45 }
```

Altere os Getters e os Setters de “public” para “private”, conforme planejado.

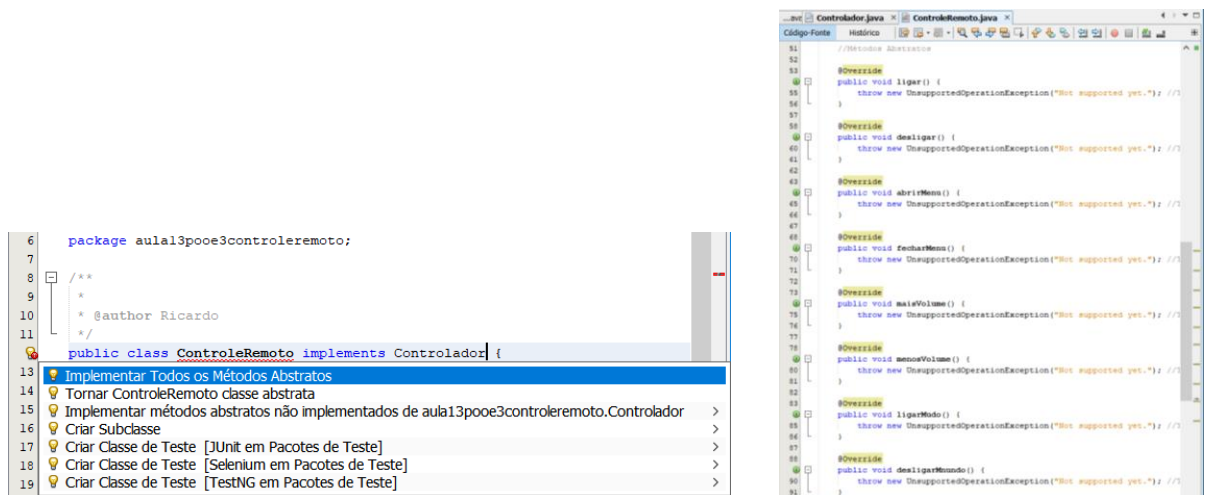
Código-Fonte	Histórico
<pre>27 public int getVolume() { 28 return volume; 29 } 30 31 public void setVolume(int volume) { 32 this.volume = volume; 33 } 34 35 public boolean getLigado() { //is para get 36 return ligado; 37 } 38 39 public void setLigado(boolean ligado) { 40 this.ligado = ligado; 41 } 42 43 public boolean getTocando() { //is para get 44 return tocando; 45 } 46 47 public void setTocando(boolean tocando) { 48 this.tocando = tocando; 49 }</pre>	<pre>27 privado Metodo getVolume() 28 retorne volume 29 FimMetodo 30 privado Metodo getLigado() 31 retorne volume 32 FimMetodo 33 privado Metodo getTocando() 34 retorne volume 35 FimMetodo 36 privado Metodo setVolume(v: Inteiro) 37 volume = v 38 FimMetodo 39 privado Metodo setLigado(l: logico) 40 ligado = l 41 FimMetodo 42 privado Metodo setTocando(t: logico) 43 tocando = t 44 FimMetodo</pre>

Código-Fonte	Histórico
<pre>27 private int getVolume() { 28 return volume; 29 } 30 31 private void setVolume(int volume) { 32 this.volume = volume; 33 } 34 35 private boolean getLigado() { //is para get 36 return ligado; 37 } 38 39 private void setLigado(boolean ligado) { 40 this.ligado = ligado; 41 } 42 43 private boolean getTocando() { //is para get 44 return tocando; 45 } 46 47 private void setTocando(boolean tocando) { 48 this.tocando = tocando; 49 }</pre>	

Para inserir os códigos necessários para os controladores bastará digitar “implements Controlador” e o NetBeans irá reconhecer a necessidade da inserção dos demais códigos automaticamente, bastando editá-los. O alerta do NetBeans acende.

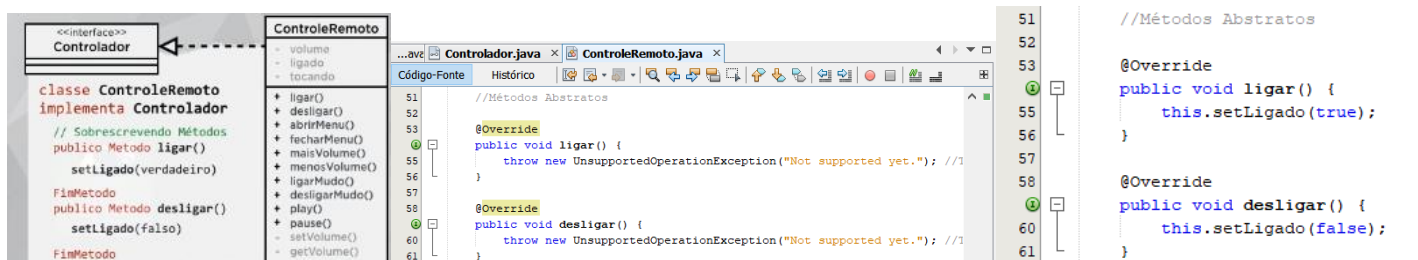


Clique sobre o alerta e após na opção “Implementar Todos os Métodos Abstratos” e os códigos são inseridos automaticamente.



Ligado e Desligado

Programa o Ligado e Desligado apagando o código “throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.” e digitando o código necessário, conforme imagem.



Abrir Menu e Fechar Menu



```
63 @Override
64 public void abrirMenu() {
65     throw new UnsupportedOperationException("Not supported yet."); //
66 }
67
68 @Override
69 public void fecharMenu() {
70     throw new UnsupportedOperationException("Not supported yet."); //
71 }
```

```
63 @Override
64 public void abrirMenu() {
65     System.out.println("Está Ligado? " + this.getLigado());
66     System.out.println("Está tocando? " + this.getTocando());
67     System.out.println("Volume: " + this.getVolume());
68     for (int i = 0; i <= this.getVolume(); i+=10) {
69         System.out.print("|");
70     }
71 }
72
73 @Override
74 public void fecharMenu() {
75     System.out.println("Fechando Menu...");
76 }
```

Mais Volume e Menos Volume, onde SE estiver ligada some ou subtraia 1 do volume.

```
public Metodo maisVolume()
Se (getLigado()) entao
    setVolume(getVolume() + 1)
FimSe
FimMetodo
public Metodo menosVolume()
Se (getLigado()) entao
    setVolume(getVolume() - 1)
FimSe
FimMetodo
```

```
78 @Override
79 public void maisVolume() {
80     if (this.getLigado()) {
81         //SE ligado, aumenta o volume em 5 unidades
82         this.setVolume(this.getVolume() + 5);
83     }
84 }
85
86 @Override
87 public void menosVolume() {
88     if (this.getLigado()) {
89         //SE ligado, diminui o volume em 5 unidades
90         this.setVolume(this.getVolume() - 5);
91     }
92 }
```

Ligar Mudo e Desligar Mudo

```
public Metodo ligarMudo()
Se (getLigado() e getVolume() > 0) entao
    setVolume(0)
FimSe
FimMetodo
public Metodo desligarMudo()
Se (getLigado() e getVolume() = 0) entao
    setVolume(50)
FimSe
FimMetodo
```

```
94 @Override
95 public void ligarMudo() {
96     if (this.getLigado() && this.getVolume() > 0) {
97         //SE ligado e volume acima de 0, zerar volume
98         this.setVolume(0);
99     }
100 }
101
102 @Override
103 public void desligarMudo() {
104     if (this.getLigado() && this.getVolume() == 0) {
105         this.setVolume(50);
106     }
107 }
```

Play e Pause

```
public Metodo play()
Se (getLigado() e nao getTocando()) entao
    setTocando(verdadeiro)
FimSe
FimMetodo
public Metodo pause()
Se (getLigado() e getTocando()) entao
    setTocando(falso)
FimSe
FimMetodo
```

```
109 @Override
110 public void play() {
111     if (this.getLigado() && !(this.getTocando())) {
112         this.setTocando(true);
113     }
114 }
115
116 @Override
117 public void pause() {
118     if (this.getLigado() && this.getTocando()) {
119         this.setTocando(false);
120     }
121 }
```

Terminado o Controlador e o ControleRemoto, agora vamos configurar a Classe main.

Teste 1. Ligar o controle e Abrir o menu

```
1  /*
2  * To change this license header, choose License Headers in Project
3  * To change this template file, choose Tools | Templates
4  * and open the template in the editor.
5  */
6  package aula13pooe3controleremoto;
7
8  /**
9   *
10  * @author Ricardo
11  */
12  public class Aula13POOE3ControleRemoto {
13
14      /**
15       * @param args the command line arguments
16       */
17      public static void main(String[] args) {
18          // TODO code application logic here
19
20          ControleRemoto controle = new ControleRemoto ();
21          controle.ligar();
22          controle.abrirMenu();
23      }
```

```
Saída - Aula13POOE3ControleRemoto (run) X
run:
Está Ligado? true
Está tocando? false
Voluma: 50
|||||CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)
```

Teste 2. Aumentar o volume.

```
17  public static void main(String[] args) {
18      // TODO code application logic here
19
20      ControleRemoto controle = new ControleRemoto ();
21      controle.ligar();
22      controle.maisVolume();
23      controle.abrirMenu();
24  }
```

```
Saída - Aula13POOE3ControleRemoto (run) X
run:
Está Ligado? true
Está tocando? false
Voluma: 55
|||||CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
```

Teste 3. Aumente mais uma vez o volume.

```
17  public static void main(String[] args) {
18      // TODO code application logic here
19
20      ControleRemoto controle = new ControleRemoto ();
21      controle.ligar();
22      controle.maisVolume();
23      controle.maisVolume();
24      controle.abrirMenu();
25  }
```

```
Saída - Aula13POOE3ControleRemoto (run) X
run:
Está Ligado? true
Está tocando? false
Voluma: 60
|||||CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
```

Teste 4. Colocar no mudo.

```
17  public static void main(String[] args) {
18      // TODO code application logic here
19
20      ControleRemoto controle = new ControleRemoto ();
21      controle.ligar();
22      controle.maisVolume();
23      controle.maisVolume();
24      controle.ligarMudo();
25      controle.abrirMenu();
26  }
```

```
Saída - Aula13POOE3ControleRemoto (run) X
run:
Está Ligado? true
Está tocando? false
Voluma: 0
|CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
```

Teste 5. Ligar novamente o volume.

```
17 public static void main(String[] args) {
18     // TODO code application logic here
19
20     ControleRemoto controle = new ControleRemoto ();
21     controle.ligar();
22     controle.maisVolume();
23     controle.maisVolume();
24     controle.ligarMudo();
25     controle.play();
26     controle.abrirMenu();
27 }
```

Saída - Aula13POOE3ControleRemoto (run) X

run:
Está Ligado? true
Está tocando? true
Volume: 0
| CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)

Teste 6. Fechar menu.

```
17 public static void main(String[] args) {
18     // TODO code application logic here
19
20     ControleRemoto controle = new ControleRemoto ();
21     controle.ligar();
22     controle.maisVolume();
23     controle.maisVolume();
24     controle.ligarMudo();
25     controle.play();
26     controle.abrirMenu();
27     controle.fecharMenu();
28 }
```

Saída - Aula13POOE3ControleRemoto (run) X

run:
Está Ligado? true
Está tocando? true
Volume: 0
| Fechando Menu...
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)

Teste 7. Não é possível utilizar acesso diretamente aos atributos “controle.volume = 80;” ou “controle.setVolume(50)” pelo fato de estarem como privados; o que muitas vezes não é bom em um ambiente de produção, mas foi ótimo para exercitarmos.

```
17 public static void main(String[] args) {
18     // TODO code application logic here
19
20     ControleRemoto controle = new ControleRemoto ();
21     controle.ligar();
22     controle.maisVolume();
23     controle.maisVolume();
24     controle.ligarMudo();
25     controle.play();
26     controle.abrirMenu();
27     controle.fecharMenu();
28     controle.volume = 80;
29     controle.setVolume();
30 }
```