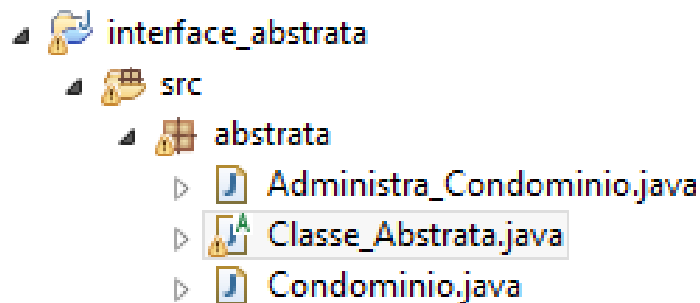


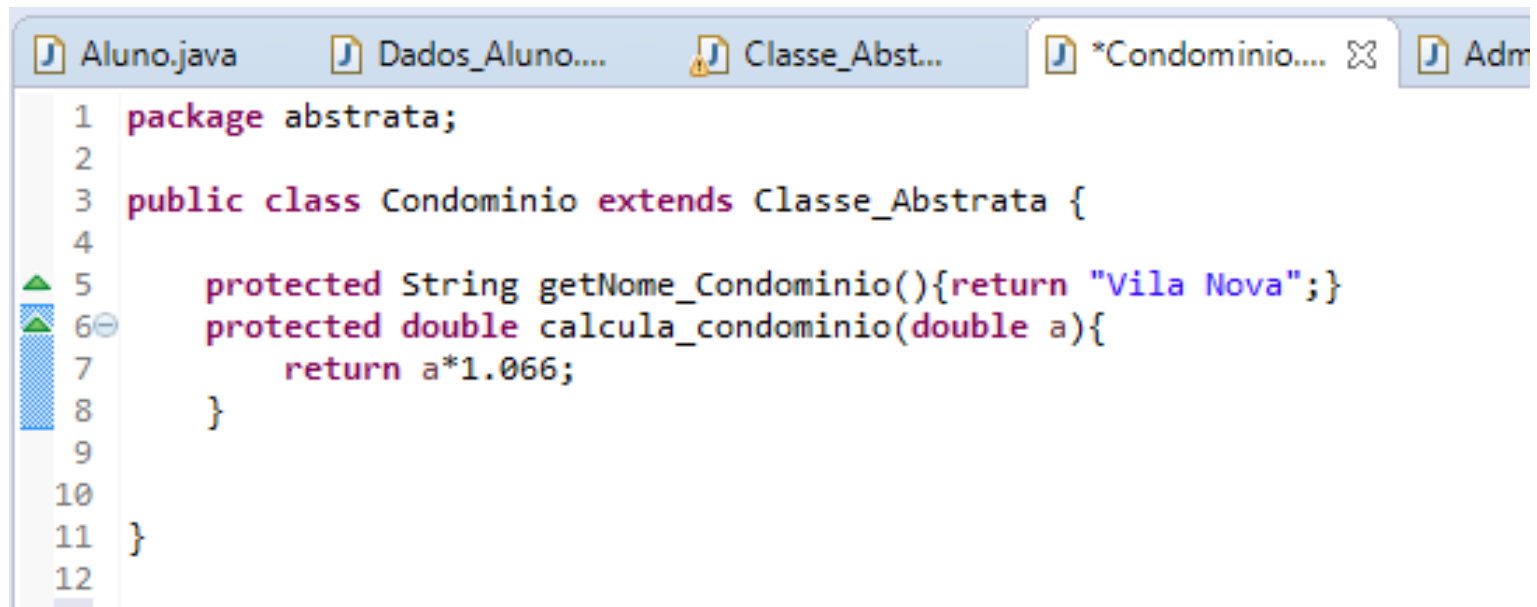
Exemplo do uso de classe abstrata.



```
Aluno.java  Dados_Aluno....  Aluno_2.java  Classe_Abst...  Administr...
```

```
1 package abstrata;
2
3 public abstract class Classe_Abstrata {
4
5     private String nome_condominio;
6     private double despesa_condominio;
7
8     protected String getNome_Condominio(){return nome_condominio;}
9     protected double calcula_condominio(double despesa_condominio)
10    {this.despesa_condominio = despesa_condominio;
11     return despesa_condominio; }
12 }
13
```

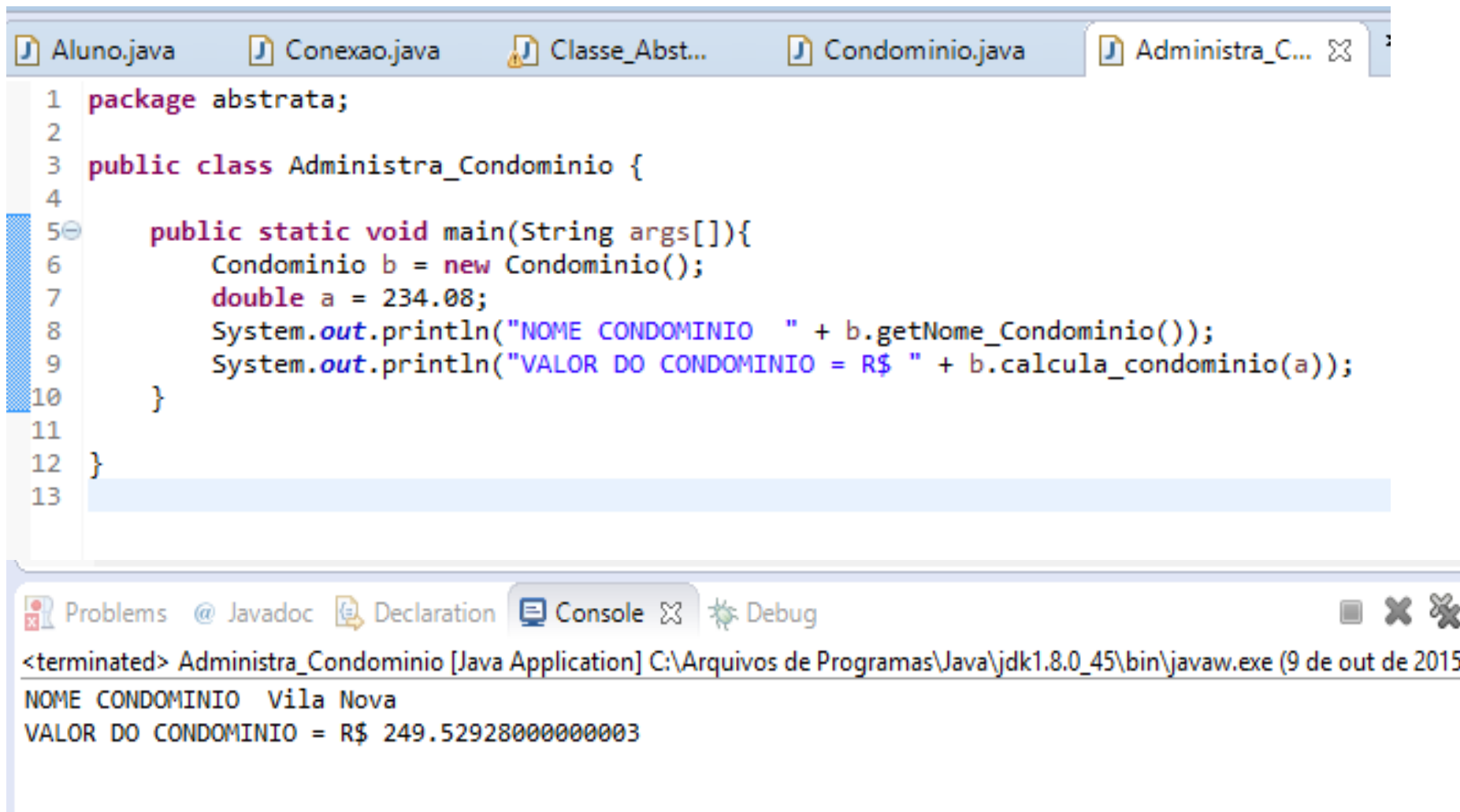
Exemplo do uso de classe abstrata.



The image shows a screenshot of a Java IDE with several tabs open: 'Aluno.java', 'Dados_Aluno....', 'Classe_Abst...', '*Condominio....', and 'Adm...'. The active tab is '*Condominio....', which contains the following Java code:

```
1 package abstrata;
2
3 public class Condominio extends Classe_Abstrata {
4
5     protected String getNome_Condominio(){return "Vila Nova";}
6     protected double calcula_condominio(double a){
7         return a*1.066;
8     }
9
10
11 }
12
```

Exemplo do uso de classe abstrata.



The image shows a screenshot of an IDE with several tabs at the top: `Aluno.java`, `Conexao.java`, `Classe_Abst...`, `Condominio.java`, and `Administra_C...`. The `Administra_C...` tab is active, displaying the following Java code:

```
1 package abstrata;
2
3 public class Administra_Condominio {
4
5     public static void main(String args[]){
6         Condominio b = new Condominio();
7         double a = 234.08;
8         System.out.println("NOME CONDOMINIO " + b.getNome_Condominio());
9         System.out.println("VALOR DO CONDOMINIO = R$ " + b.calcula_condominio(a));
10    }
11
12 }
13
```

Below the code editor, the IDE's output window is visible, showing the execution results:

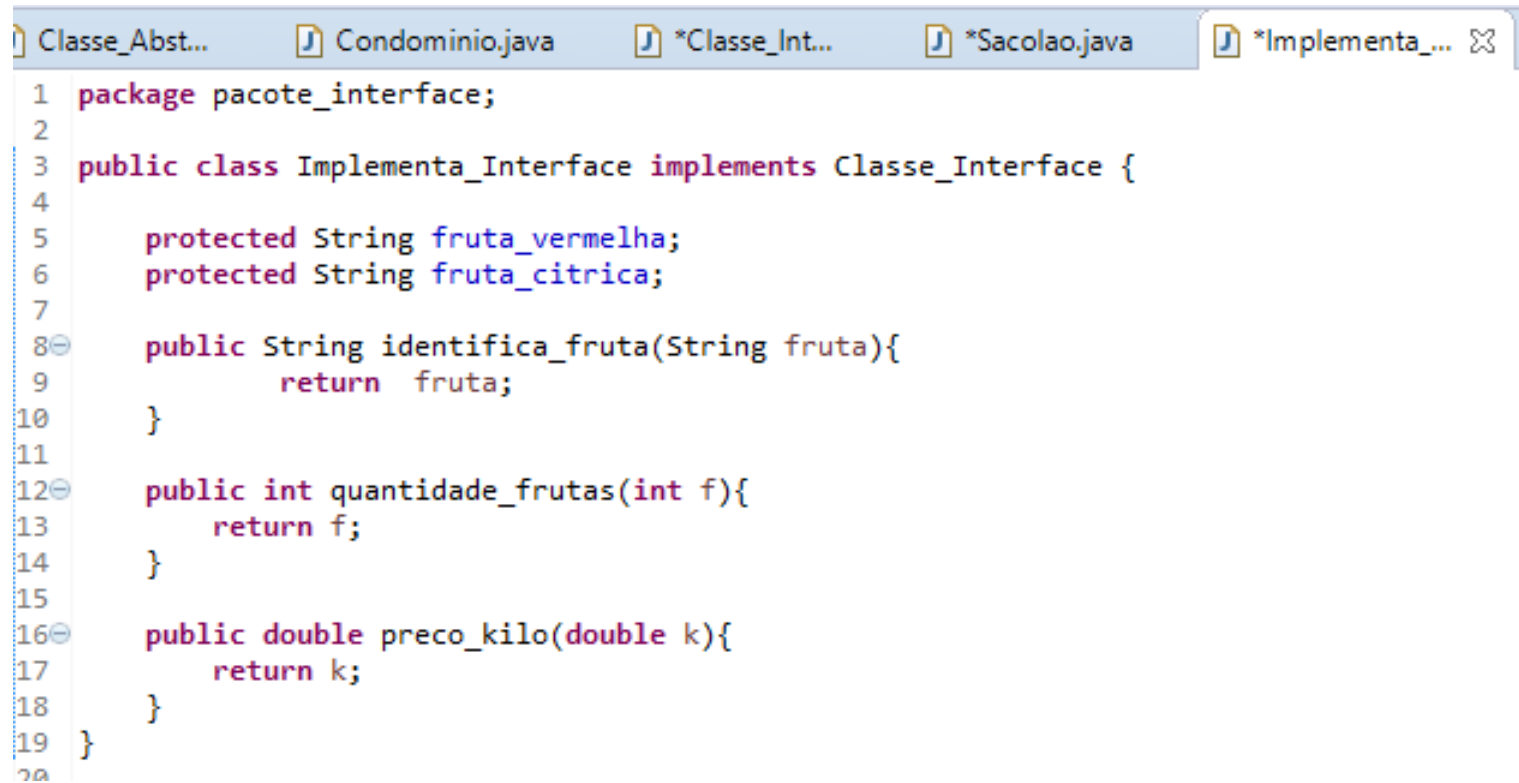
```
<terminated> Administra_Condominio [Java Application] C:\Arquivos de Programas\Java\jdk1.8.0_45\bin\javaw.exe (9 de out de 2015)
NOME CONDOMINIO Vila Nova
VALOR DO CONDOMINIO = R$ 249.52928000000003
```

Exemplo do uso de interface.

- pacote_interface
 - Classe_Interface.java
 - Implementa_Interface.java
 - Sacolao.java

```
1 package pacote_interface;
2
3 public interface Classe_Interface {
4
5     public String identifica_fruta(String fruta);
6     public int quantidade_frutas(int f);
7     public double preco_kilo(double k );
8 }
9
10
```

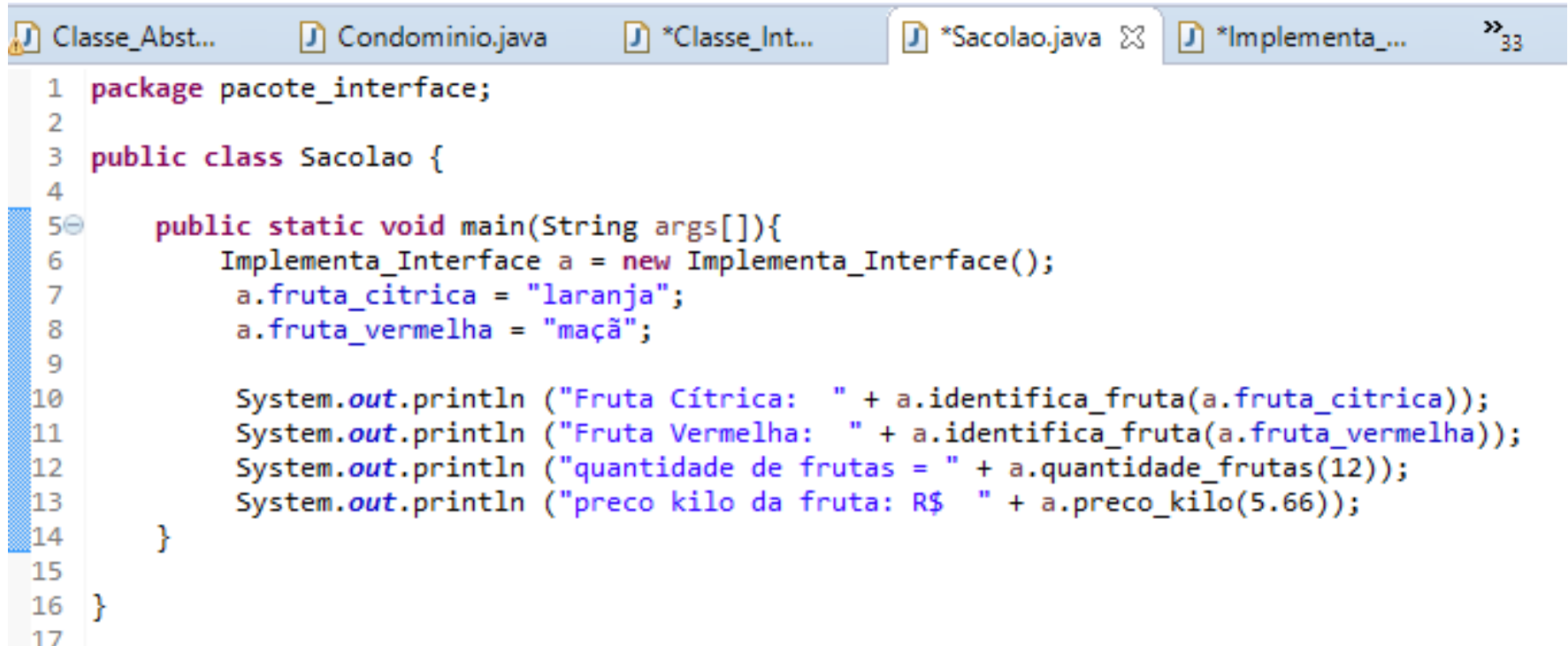
Exemplo do uso de interface.



The screenshot shows an IDE with several tabs at the top: 'Classe_Abst...', 'Condominio.java', '*Classe_Int...', '*Sacolao.java', and '*Implementa_...' (which is the active tab). The code in the active tab is as follows:

```
1 package pacote_interface;
2
3 public class Implementa_Interface implements Classe_Interface {
4
5     protected String fruta_vermelha;
6     protected String fruta_citrica;
7
8     public String identifica_fruta(String fruta){
9         return fruta;
10    }
11
12    public int quantidade_frutas(int f){
13        return f;
14    }
15
16    public double preco_kilo(double k){
17        return k;
18    }
19 }
```

Exemplo do uso de interface.



The screenshot shows an IDE with several tabs: 'Classe_Abst...', 'Condominio.java', '*Classe_Int...', '*Sacolao.java', and '*Implementa_...'. The active tab is '*Sacolao.java'. The code is as follows:

```
1 package pacote_interface;
2
3 public class Sacolao {
4
5     public static void main(String args[]){
6         Implementa_Interface a = new Implementa_Interface();
7         a.fruta_citrica = "laranja";
8         a.fruta_vermelha = "maçã";
9
10        System.out.println ("Fruta Cítrica: " + a.identifica_fruta(a.fruta_citrica));
11        System.out.println ("Fruta Vermelha: " + a.identifica_fruta(a.fruta_vermelha));
12        System.out.println ("quantidade de frutas = " + a.quantidade_frutas(12));
13        System.out.println ("preco kilo da fruta: R$ " + a.preco_kilo(5.66));
14    }
15
16 }
17
```

- **Exercício classe abstrata:** Construir e executar o exemplo de aplicação com classe abstrata.
- **Exercício interface:** Construir e executar o exemplo de aplicação com interface.
- **Lista do 2º bimestre**
- **1.** Considerar uma aplicação que represente as atividades de uma academia de esportes. Criar a aplicação utilizando o conceito de classe abstrata.
- **2.** Considerar uma aplicação que represente um campeonato de futebol (série C do campeonato brasileiro, por exemplo). Criar a aplicação utilizando o conceito de interface.