

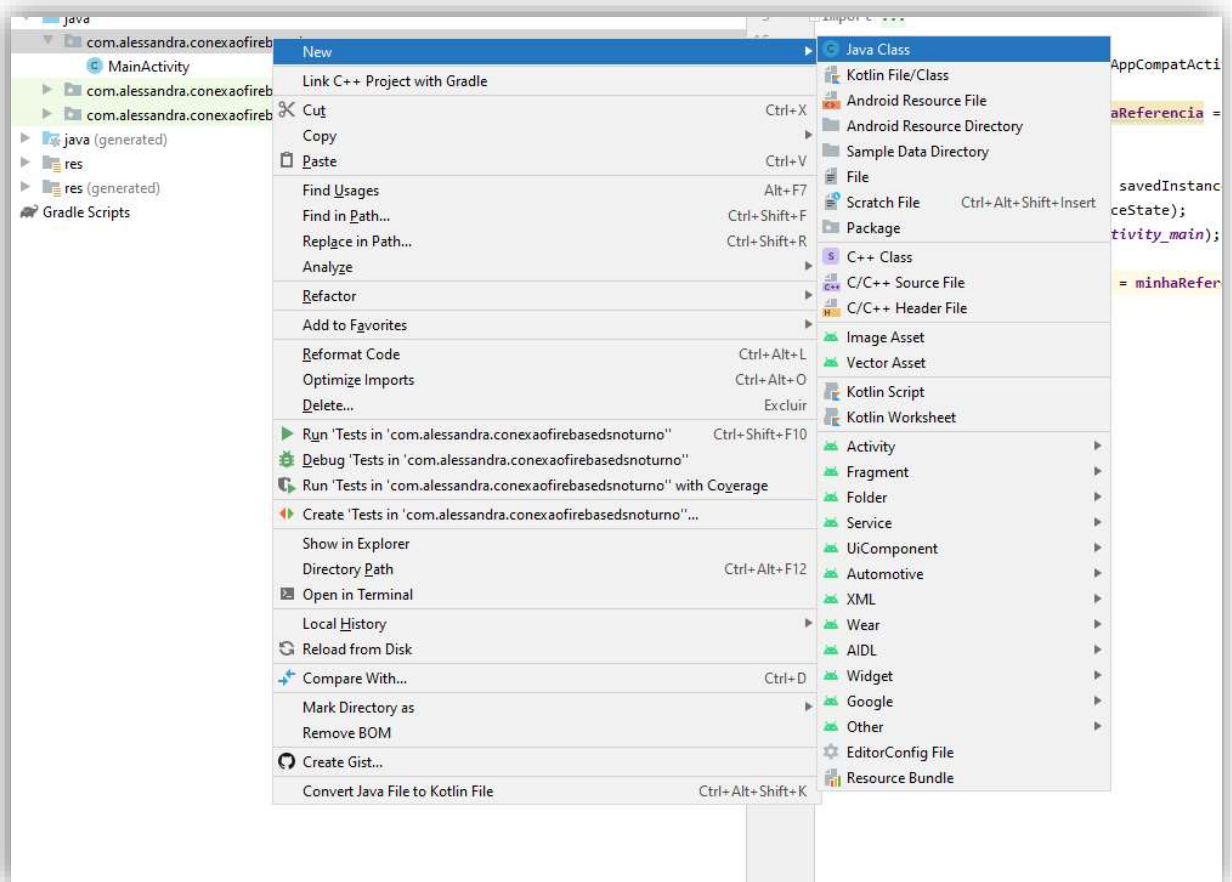
Eu poderia também criar uma referência para produtos, da seguinte maneira:

```
16 public class MainActivity extends AppCompatActivity {
17
18     private DatabaseReference minhaReferencia = FirebaseDatabase.getInstance().getReference();
19
20     @Override
21     protected void onCreate(Bundle savedInstanceState) {
22         super.onCreate(savedInstanceState);
23         setContentView(R.layout.activity_main);
24
25         DatabaseReference produtos = minhaReferencia.child("produtos");
26
27     }
28 }
```

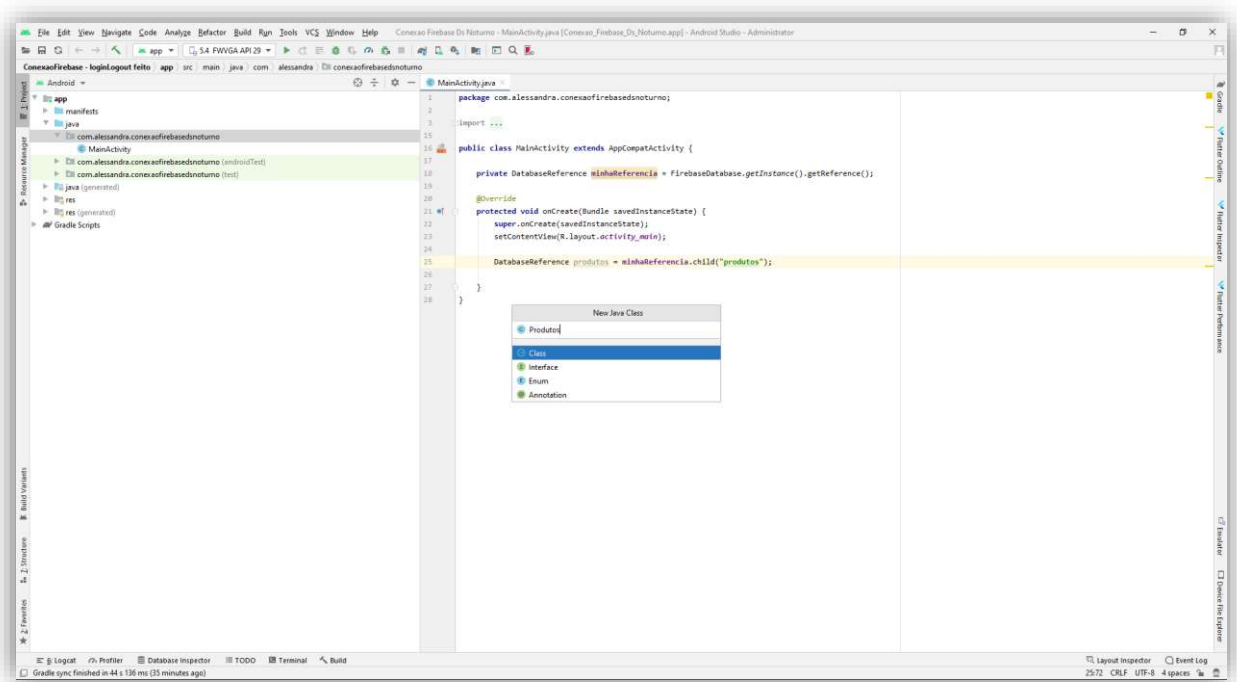
## Salvando Dados de Maneira Orientada a Objetos

Há uma maneira de fazer tudo isso de forma orientada a objetos. Isso porque o `setValue` também aceita objetos como parâmetros. Então, vamos criar uma classe.

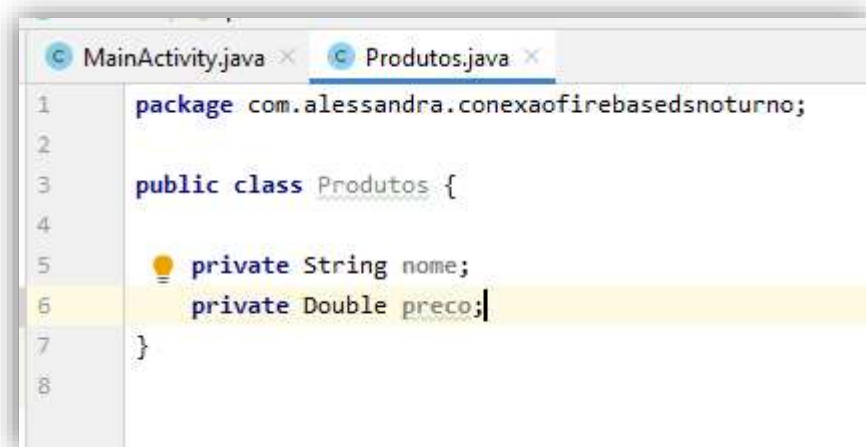
Clique com o botão direito do mouse no nome do seu package, depois em **new** e em seguida em **Java Class**:



Crie uma classe com o nome Produtos:

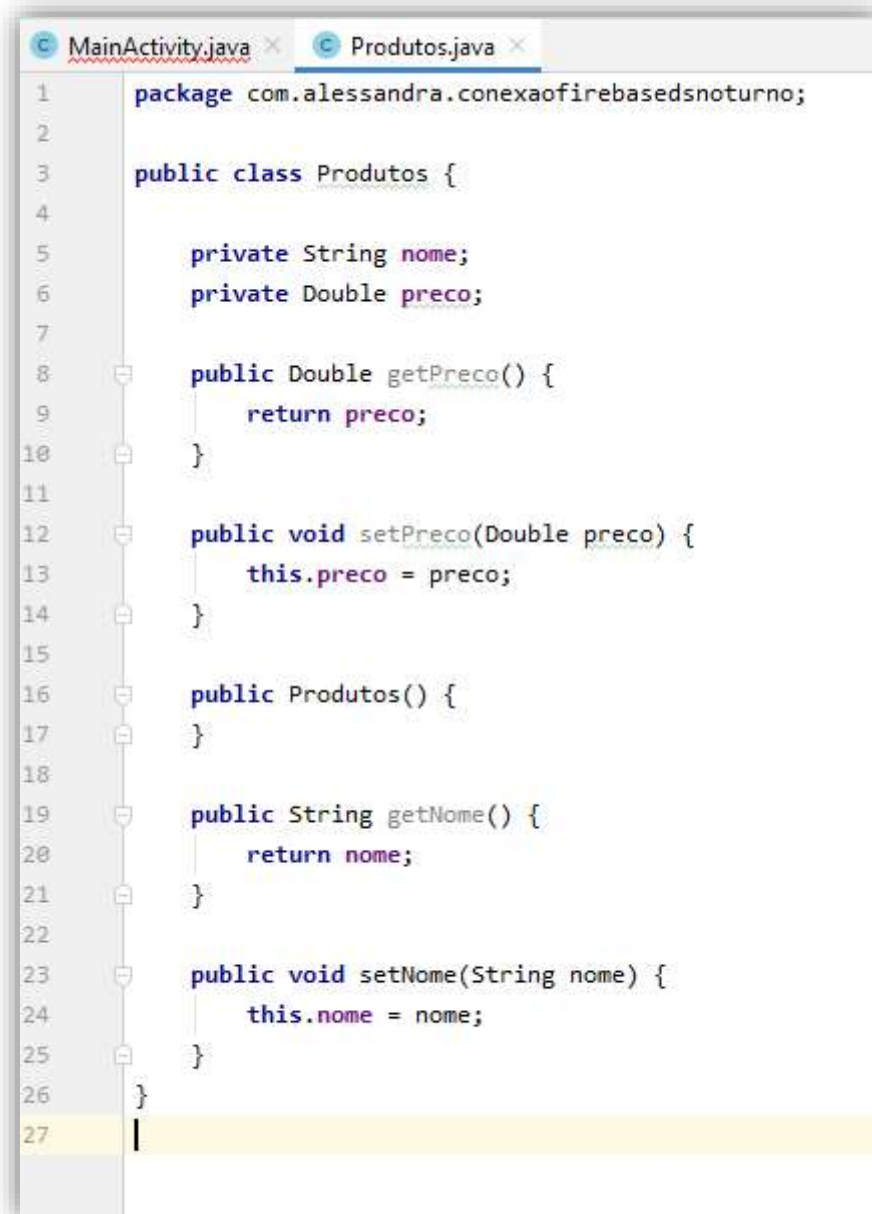


Dentro desta nova classe, crie os atributos que desejar. No meu caso, criei nome e preço:



```
1 package com.alessandra.conexaofirebasedsnoturno;
2
3 public class Produtos {
4
5     private String nome;
6     private Double preco;
7 }
8
```

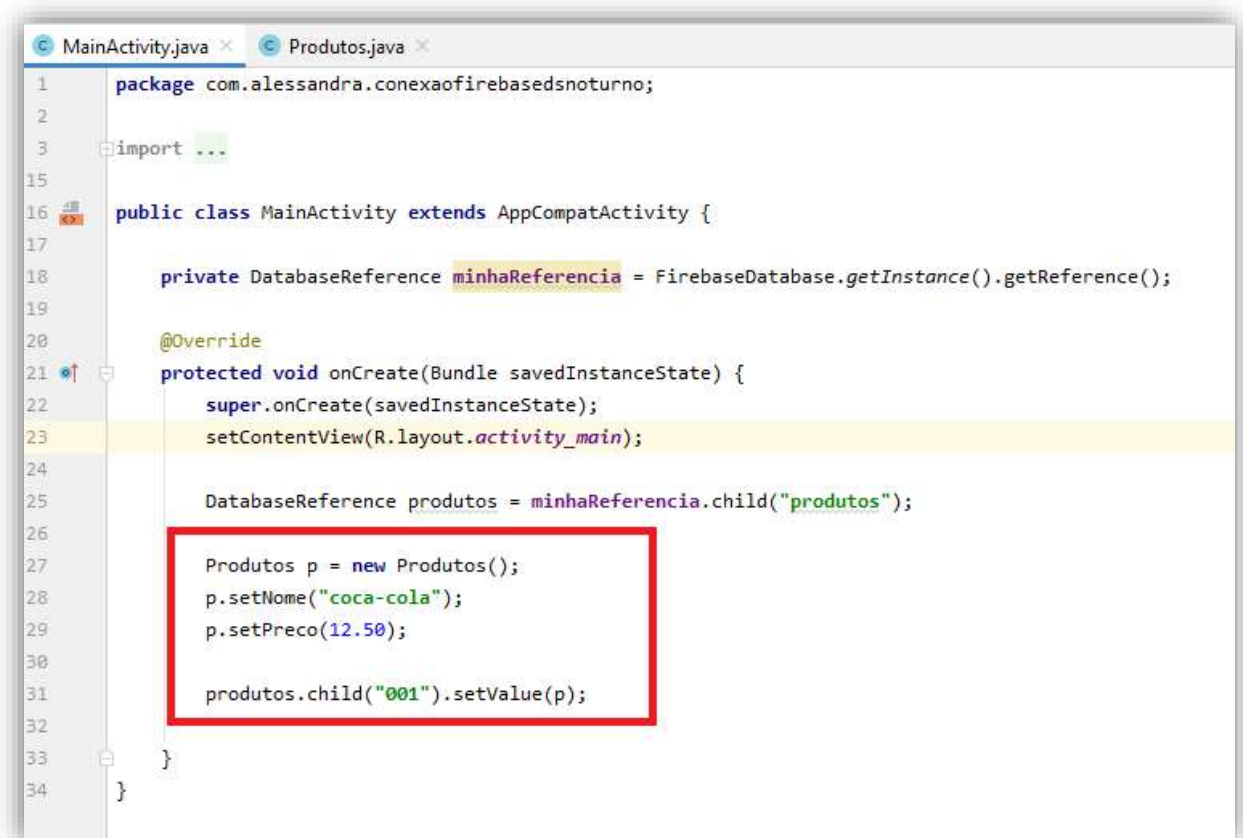
Em seguida, clique com o botão direito do mouse e clique em **Generate** e gere, primeiro o **construtor** vazio e depois os métodos **gets e sets**. Ficará assim:

A screenshot of an IDE window showing two tabs: MainActivity.java and Produtos.java. The Produtos.java tab is active, displaying the following Java code:

```
1 package com.alessandra.conexaofirebasedsnoturno;
2
3 public class Produtos {
4
5     private String nome;
6     private Double preco;
7
8     public Double getPreco() {
9         return preco;
10    }
11
12    public void setPreco(Double preco) {
13        this.preco = preco;
14    }
15
16    public Produtos() {
17    }
18
19    public String getNome() {
20        return nome;
21    }
22
23    public void setNome(String nome) {
24        this.nome = nome;
25    }
26 }
27
```

The code defines a class named 'Produtos' with two private attributes: 'nome' (String) and 'preco' (Double). It includes four methods: 'getPreco()' returning the price, 'setPreco()' setting the price, 'getNome()' returning the name, and 'setNome()' setting the name. There is also a default constructor. The line numbers 1 through 27 are visible on the left side of the editor.

Agora, na classe principal, eu posso setar os valores e passar como parâmetro do setValue apenas o objeto da classe Produtos:



```
1 package com.alessandra.conexaofirebasednoturno;
2
3 import ...
4
15
16 public class MainActivity extends AppCompatActivity {
17
18     private DatabaseReference minhaReferencia = FirebaseDatabase.getInstance().getReference();
19
20     @Override
21     protected void onCreate(Bundle savedInstanceState) {
22         super.onCreate(savedInstanceState);
23         setContentView(R.layout.activity_main);
24
25         DatabaseReference produtos = minhaReferencia.child("produtos");
26
27         Produtos p = new Produtos();
28         p.setNome("coca-cola");
29         p.setPreco(12.50);
30
31         produtos.child("001").setValue(p);
32     }
33 }
34 }
```

Execute e você verá que os dados serão escritos na base de dados, da mesma maneira.