

Amend

brinca muito

Commit

Commit and Push...

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

```
System.out.print("Digite a opção desejada: ");
opcao = scanner.nextInt();
scanner.nextLine(); // Consumir a quebra de linha

switch (opcao) {
    case 1:
        System.out.print("Escolha o tipo (1 para Pessoa Física, 2 para Pessoa Jurídica): ");
        int tipo = scanner.nextInt();
        scanner.nextLine(); // Consumir a quebra de linha

        if (tipo == 1) {
            System.out.println("você deve inserir o id");
            int id = scanner.nextInt();
            scanner.nextLine();
            System.out.println("você deve inserir o nome");
            String nome = scanner.nextLine();
        }
    }
}
```

Cadastro\_J > src > Main > main

214:65 LF UTF-8 4 spaces

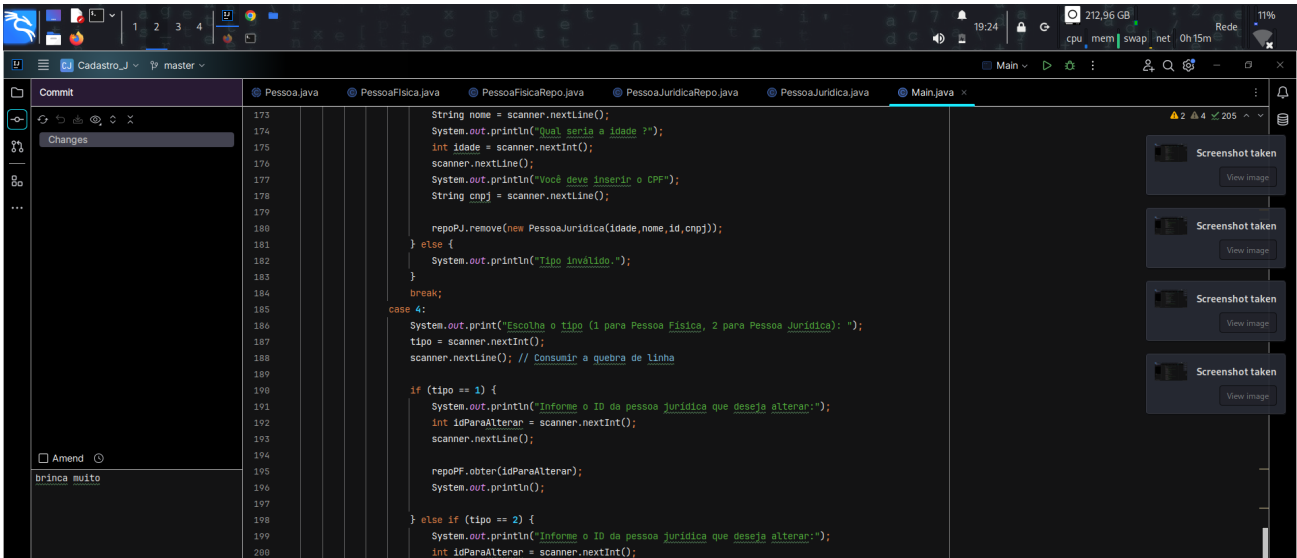
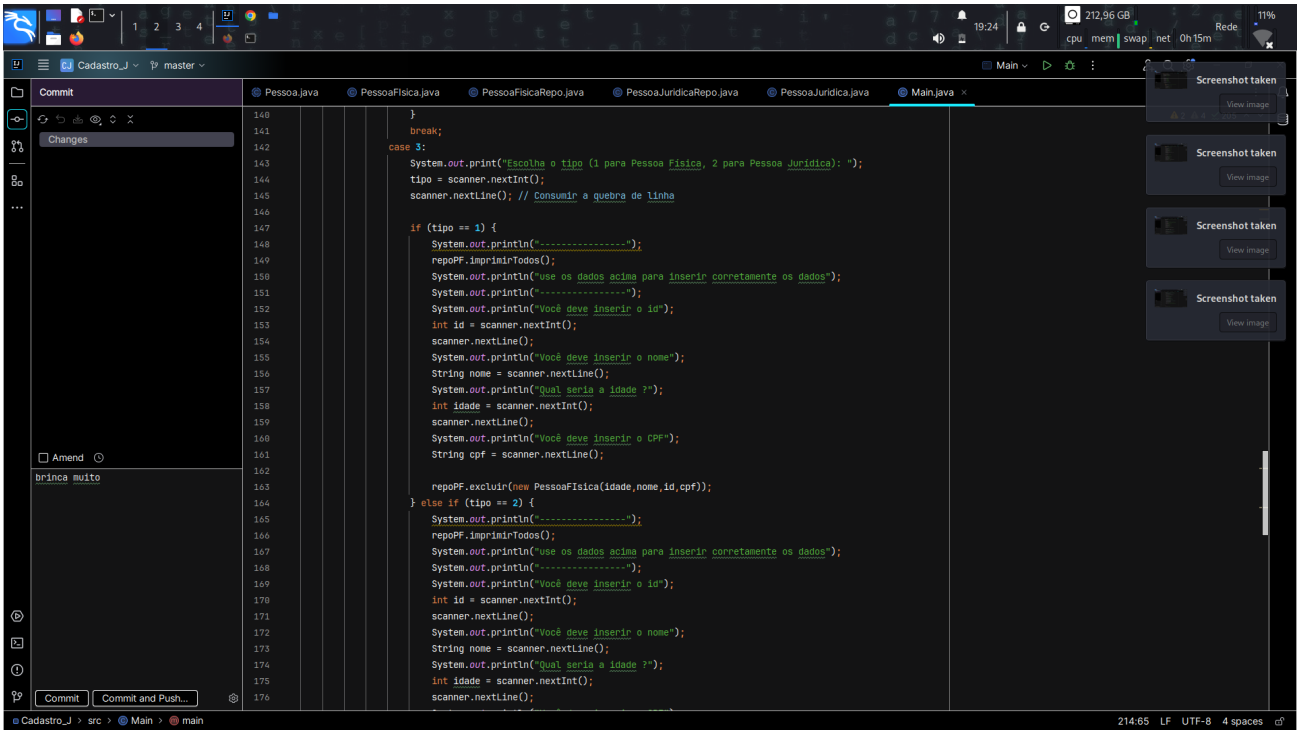
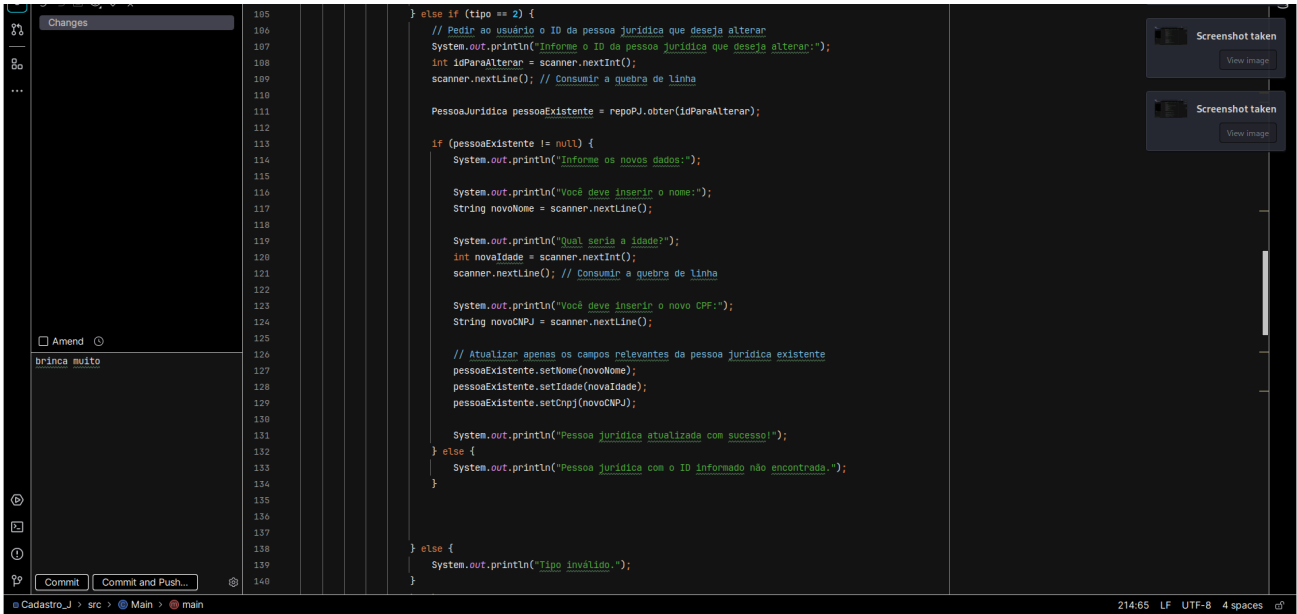
The screenshot shows an IDE window titled "Cadastro\_J" with a file explorer on the left showing a "master" branch. The main editor displays the "Main.java" file, which contains the following code:

```
1  System.out.println("Escolha o tipo (1 para Pessoa Fisica, 2 para Pessoa Juridica): ");
2  int tipo = scanner.nextInt();
3  scanner.nextLine(); // Consumir a quebra de linha
4
5  if (tipo == 1) {
6      System.out.println("você deve inserir o id");
7      int id = scanner.nextInt();
8      scanner.nextLine();
9      System.out.println("você deve inserir o nome");
10     String nome = scanner.nextLine();
11     System.out.println("Qual seria a idade ?");
12     int idade = scanner.nextInt();
13     scanner.nextLine();
14     System.out.println("você deve inserir o CPF");
15     String cpf = scanner.nextLine();
16     PessoaFisica pessoa = new PessoaFisica(idade, nome, id, cpf);
17     repoPF.inserir(pessoa);
18     repoPF.imprimirTodos();
19
20     repoPF.obterTodos();
21     System.out.println("operação finalizada");
22
23 } else if (tipo == 2) {
24     System.out.println("você deve inserir id");
25     int id = scanner.nextInt();
26     System.out.println("você deve inserir o nome");
27     String nome = scanner.next();
28     System.out.println("Qual seria a idade ?");
29     int idade = scanner.nextInt();
30     System.out.println("você deve inserir o cpf");
31     String cnpj = scanner.next();
32     PessoaJuridica pesssoa = new PessoaJuridica(idade, nome, id, cnpj);
33     repoPJ.inserir(pesssoa);
34
35     System.out.println("operação finalizada");
36 }
```

The code is written in Java and uses the Scanner class for input. It includes comments in Portuguese. The IDE interface shows a "Commit" button at the bottom left and a "Screenshot taken" notification in the top right corner.

```
177 System.out.println("Escolha o tipo (1 para Pessoa Fisica, 2 para Pessoa Juridica): ");
178 tipo = scanner.nextInt();
179 scanner.nextLine();
180
181 if (tipo == 1) {
182     System.out.println("-----");
183     repoPF.imprimirTodos();
184     System.out.println("use os dados acima para inserir corretamente os dados");
185     System.out.println("-----");
186     System.out.println("Os campos a seguir ser  para inser o do novo cadastro");
187     System.out.println("Voc  deve inserir o id");
188     int id = scanner.nextInt();
189     scanner.nextLine();
190     System.out.println("Voc  deve inserir o nome");
191     String name = scanner.nextLine();
192     System.out.println("Qual seria a idade ?");
193     int idade = scanner.nextInt();
194     scanner.nextLine();
195     System.out.println("Voc  deve inserir o CPF");
196     String cpf = scanner.nextLine();
197     PessoaFisica pessoaNova = new PessoaFisica(idade, name, id, cpf);
198     System.out.println("-----");
199
200     PessoaFisica pessoaAntiga = new PessoaFisica(idade, name, id, cpf);
201     repoPF.alterar(pessoaAntiga, pessoaNova);
202
203 } else if (tipo == 2) {
204     // Pedir ao usu rio o ID da pessoa jur dica que deseja alterar
205     System.out.println("Informe o ID da pessoa jur dica que deseja alterar:");
206     int idParaAlterar = scanner.nextInt();
207     scanner.nextLine(); // Consumir a quebra de linha
208
209     PessoaJuridica pessoaExistente = repoPJ.obter(idParaAlterar);
210
211     if (pessoaExistente != null) {
```

[illegible]



```
201         repoPJ.obter(idParaAlterar);
202     } else {
203         System.out.println("Tipo inválido.");
204     }
205     break;
206
207     case 5:
208         System.out.print("Escolha o tipo (1 para Pessoa Física, 2 para Pessoa Jurídica): ");
209         tipo = scanner.nextInt();
210         scanner.nextLine(); // Consumir a quebra de linha
```

```
203         System.out.println("Tipo inválido.");
204     }
205     break;
206
207     case 5:
208         System.out.print("Escolha o tipo (1 para Pessoa Física, 2 para Pessoa Jurídica): ");
209         tipo = scanner.nextInt();
210         scanner.nextLine(); // Consumir a quebra de linha
211
212         if (tipo == 1) {
213             System.out.println("-----");
214             repoPF.imprimirTodos();
215             System.out.println("-----");
216
217         } else if (tipo == 2) {
218             System.out.println("-----");
219             repoPJ.imprimirTodos();
220             System.out.println("-----");
221
222         } else {
223             System.out.println("Tipo inválido.");
224         }
225     break;
226
227     case 6:
228         repoPF.persistir(nomeArquivoPF);
229         repoPJ.persistir(nomeArquivoPJ);
230         System.out.println("Dados persistidos nos arquivos.");
231     break;
232
233     case 7:
234         repoPF.recuperar(nomeArquivoPF);
235         repoPJ.recuperar(nomeArquivoPJ);
236         System.out.println("Dados recuperados dos arquivos.");
237     break;
238
239     case 8:
240         System.out.println("Finalizando a execução.");
241     break;
242
243     default:
244         System.out.println("Tipo inválido.");
245     }
```

```
2  public class PessoaFisica extends Pessoa {
3
4     private String cpf;
5
6     public PessoaFisica(int idade, String nome, int id, String cpf) {
7         super(idade, nome, id);
8
9         this.cpf = cpf;
10    }
11
12    public String getCpf() { return cpf; }
13    public void setCpf(String cpf) { this.cpf = cpf; }
14
15    @Override
16    public String toString() {
17        return "\n" +
18            "    Id = " + id + "\n" +
19            "    Nome = " + nome + "\n" +
20            "    Cnpj = " + cpf + "\n" +
21            "    Idade = " + idade + "\n" +
22            "    ";
23    }
24 }
```

```
4 import java.util.ArrayList;
5 import java.io.*;
6
7 public class PessoaFisicaRepo {
8     /* Lista privada para armazenar pessoas */
9     private List<PessoaFisica> pessoas;
```

Amend

brinca muito

Commit

Commit and Push...

```
10
11 public PessoaFisicaRepo(){
12     pessoas = new ArrayList<>();
13 }
14
15
16 > 1 usage  Aquilae777
17 public void inserir(PessoaFisica pessoa){pessoas.add(pessoa);}
18
19
20 1 usage  Aquilae777
21 public void alterar(PessoaFisica pessoaAntiga, PessoaFisica pessoaNova){
22     int index = pessoas.indexOf(pessoaAntiga);
23     if (index != -1){
24         pessoas.set(index,pessoaNova);
25     }
26 }
27
28 > 1 usage  Aquilae777
29 public void excluir(PessoaFisica pessoa){pessoas.remove(pessoa);}
30
31
32 1 usage  Aquilae777
33 public PessoaFisica obter(int id){
34     for(PessoaFisica pessoa : pessoas){
35         if(pessoa.getId() == id){
36             return pessoa;
37         }
38     }
39     return null;
40 }
```

Cadastro\_J > src > model > PessoaFisicaRepo > PessoaFisicaRepo

11:6 LF UTF-8 4 spaces

Commit

Changes

Amend

brinca muito

Commit

Commit and Push...

Cadastro\_J &gt; src &gt; model &gt; Pessoa &gt; nome

7:5 LF UTF-8 4 spaces

