

```
package CadastroPoo;
```

```
import java.io.IOException;
```

```
import Model.PessoaFisica;
```

```
import Model.PessoaFisicaRepo;
```

```
import Model.PessoaJuridica;
```

```
import Model.PessoaJuridicaRepo;
```

```
import java.util.InputMismatchException;
```

```
import java.util.Scanner;
```

```
public class CadastroPOO {
```

```
    private static final Scanner scan = new Scanner(System.in);
```

```
    private static final PessoaFisicaRepo rPessoaFisica = new PessoaFisicaRepo();
```

```
    private static final PessoaJuridicaRepo rPessoaJuridica = new PessoaJuridicaRepo();
```

```
    public static void main(String[] args) {
```

```
        try (scan) {
```

```
            int opcao;
```

```
            do {
```

```
                exibirMenu();
```

```
                opcao = receberOpcao();
```

```
                switch (opcao) {
```

```
                    case 1 -> incluirPessoa();
```

```
                    case 2 -> alterarPessoa();
```

```

        case 3 -> excluirPessoa();

        case 4 -> exibirPessoaId();

        case 5 -> exibirTodasPessoas();

        case 6 -> salvarDados();

        case 7 -> recuperarDados();

        case 0 -> System.out.println("Estamos Finalizando . . .");

        default -> System.out.println("Opção inválida.");

    }

    } while (opcao != 0);

}
}

```

```

private static void exibirMenu() {

    System.out.println("Selecione uma opção: ");

    System.out.println("1 - Adicionar");

    System.out.println("2 - Alterar");

    System.out.println("3 - Excluir");

    System.out.println("4 - Apresentar pelo ID");

    System.out.println("5 - Apresentar todos");

    System.out.println("6 - Salvar todos os dados");

    System.out.println("7 - Recuperar os dados");

    System.out.println("0 - Terminar");

}

```

```

private static int receberOpcao() {

```

```
System.out.print("opção: ");  
  
return scan.nextInt();  
  
}
```

```
private static void incluirPessoa() {  
  
    System.out.println("Escolha a Opção (1) - Pessoa Física, (2) - Pessoa Juridica:");  
  
    int cadastroTipo = scan.nextInt();  
  
    scan.nextLine();  
  
    try {  
  
        switch (cadastroTipo) {  
  
            case 1 -> incluirPessoaFisica();  
  
            case 2 -> incluirPessoaJuridica();  
  
            default -> System.out.println("Opção inválida.");  
  
        }  
  
    } catch (InputMismatchException e) {  
  
        System.out.println("Entrada invalida, Cetifique-se de colocar um número valido.");  
  
    }  
  
}
```

```
private static void incluirPessoaFisica() {  
  
    System.out.println("Entre com o ID, Nome, CPF e Idade: ");  
  
    int id = scan.nextInt();  
  
    String nome = scan.next();  
  
    String cpf = scan.next();  
  
    int idade = scan.nextInt();  
  
}
```

```
        rPessoaFisica.inserir(new PessoaFisica(id, nome, idade, cpf));  
  
        System.out.println("Pessoa fisica cadastrada com Exito!");  
    }  
}
```

```
private static void incluirPessoaJuridica() {  
  
    System.out.println("Entre com o ID, Nome e CNPJ: ");  
  
    int id = scan.nextInt();  
  
    String nome = scan.next();  
  
    String cnpj = scan.next();  
  
    rPessoaJuridica.inserir(new PessoaJuridica(id, nome, cnpj));  
  
    System.out.println("Pessoa Juridica cadastrada com Exito! ");  
}
```

```
private static void alterarPessoa() {  
  
    System.out.println("Escolha a opcao (1) - Pessoa Física, (2) - Pessoa Juridica:");  
  
    int cadastroTipo = scan.nextInt();  
  
    scan.nextInt();  
  
    try {  
  
        switch (cadastroTipo) {  
  
            case 1 -> alterarPessoaFisica();  
  
            case 2 -> alterarPessoaJuridica();  
  
            default -> System.out.println("Opção Inválida.");  
  
        }  
  
    } catch (InputMismatchException e) {  
  
        System.out.println("Entrada invalida, Cetifique-se de colocar um número valido.");  
    }  
}
```

```
}  
}
```

```
private static void alterarPessoaFisica() {  
  
    System.out.println("Digite o ID que precisa ser alterado:");  
  
    int id = scan.nextInt();  
  
    PessoaFisica pessoa = rPessoaFisica.obter(id);  
  
    if (pessoa != null) {  
  
        System.out.println("Dados atuais do Usuario:");  
  
        pessoa.exibir();  
  
        System.out.println("Entre com os novos dados: (CPF, Nome e Idade)");  
  
        String cpf = scan.next();  
  
        String nome = scan.next();  
  
        int idade = scan.nextInt();  
  
        pessoa.setCpf(cpf);  
  
        pessoa.setNome(nome);  
  
        pessoa.setIdade(idade);  
  
        rPessoaFisica.alterar(pessoa);  
  
        System.out.println("Pessoa física alterada com sucesso!");  
  
    } else {  
  
        System.out.println("Pessoa física não encontrada");  
  
    }  
  
}
```

```
private static void alterarPessoaJuridica() {
```

```
System.out.println("Digite o ID que precisa ser alterado:");

int id = scan.nextInt();

PessoaJuridica pessoa = rPessoaJuridica.obter(id);

if (pessoa != null) {

    System.out.println("Dados atuais do Usuario:");

    pessoa.exibir();

    System.out.println("Entre com os novos dados: (CNPJ e Nome)");

    String cnpj = scan.next();

    String nome = scan.next();

    pessoa.setNome(nome);

    pessoa.setCnpj(cnpj);

    rPessoaJuridica.alterar(pessoa);

    System.out.println("Pessoa Juridica alterada com sucesso!");

} else {

    System.out.println("Pessoa Juridica não encontrada");

}

}
```

```
private static void excluirPessoa() {

    System.out.println("Escolha a opção (1) - Pessoa Física, (2) - Pessoa Jurídica:");

    int cadastroTipo = scan.nextInt();

    scan.nextInt();

    try {

        switch (cadastroTipo) {

            case 1 -> excluirPessoaFisica();

        }

    }

}
```

```
        case 2 -> excluirPessoaJuridica();

        default -> System.out.println("Opção Invalida");

    }

} catch (InputMismatchException e) {

    System.out.println("InputMismatchException e");

}

}
```

```
private static void excluirPessoaFisica() {

    System.out.println("Por favor entre com o ID da pessoa que vai ser Excluida: ");

    int id = scan.nextInt();

    PessoaFisica pessoa = rPessoaFisica.obter(id);

    if (pessoa != null) {

        rPessoaFisica.excluir(id);

        System.out.println("Pessoa fisica Excluida com Exito!");

    } else {

        System.out.println("Pessoa fisica não encontrada");

    }

}
```

```
private static void excluirPessoaJuridica() {

    System.out.println("Por favor entre com o ID da Pessoa Jurica que vai ser Excluida: ");

    int id = scan.nextInt();

    PessoaJuridica pessoa = rPessoaJuridica.obter(id);

}
```

```

if (pessoa != null) {

    rPessoaJuridica.excluir(id);

    System.out.println("Pessoa Juridica excluida com Exito!");

} else {

    System.out.println("Pessoa Juridica não encontrada");

}

}

```

```

private static void exibirPessoaId() {

    System.out.println("Escolha a opcao (1) - Pessoa Física, (2) - Pessoa Juridica:");

    int cadastroTipo = scan.nextInt();

    scan.nextInt();

    try {

        switch (cadastroTipo) {

            case 1 -> exibirPessoaFisicald();

            case 2 -> exibirPessoaJuridicald();

            default -> System.out.println("Opção invalida");

        }

    } catch (InputMismatchException e) {

        System.out.println("Entrada invalida, Cetifique-se de colocar um número valido.");

    }

}

```

```

private static void exibirPessoaFisicald() {

    System.out.println("Entre com o ID da pessoa fisica que vai ser exebida: ");

}

```



```

int id = scan.nextInt();

PessoaFisica pessoa = rPessoaFisica.obter(id);

if (pessoa != null) {

    pessoa.exibir();

} else {

    System.out.println("Pessoa fisica não encontrada");

}

}

private static void exibirPessoaJuridicaId() {

    System.out.println("Entre com o ID da pessoa juridica que vai ser exibida: ");

    int id = scan.nextInt();

    PessoaJuridica pessoa = rPessoaJuridica.obter(id);

    if (pessoa != null) {

        pessoa.exibir();

    } else {

        System.out.println("Pessoa Juridica não encontrada");

    }

}

private static void exibirTodasPessoas() {

    System.out.println("Escolha a opção (1) - Pessoa Física, (2) - Pessoa Jurídica:");

    int cadastroTipo = scan.nextInt();

    scan.nextInt();

    try {

```

```
switch (cadastroTipo) {  
    case 1 -> exibirTodasPessoasFisicas();  
    case 2 -> exibirTodasPessoasJuridicas();  
    default -> System.out.println("Opção Invalida");  
}  
} catch (InputMismatchException e) {  
    System.out.println("Entrada invalida, Cetifique-se de colocar um número valido.");  
}  
}
```

```
private static void exibirTodasPessoasFisicas() {  
    System.out.println("Pessoas Fisicas");  
    for (PessoaFisica pessoa : rPessoaFisica.obterTodos()) {  
        pessoa.exibir();  
        System.out.println();  
    }  
}
```

```
private static void exibirTodasPessoasJuridicas() {  
    System.out.println("Pessoas Juridicas");  
    for (PessoaJuridica pessoa : rPessoaJuridica.obterTodos()) {  
        pessoa.exibir();  
        System.out.println();  
    }  
}
```

```
private static void salvarDados() {  
  
    System.out.println("Entre com o prefixo para salvar os arquivos: ");  
  
    String salvarComo = scan.next();  
  
    try {  
  
        rPessoaFisica.persistir(salvarComo + ".fisica.bin");  
  
        rPessoaJuridica.persistir(salvarComo + ".juridica.bin");  
  
        System.out.println("Dados salvos com Exito!");  
  
    } catch (IOException e) {  
  
        System.out.println("Erro, não consigo salvar seus dados!" + e.getMessage());  
  
    }  
}
```

```
private static void recuperarDados() {  
  
    System.out.println("Entre com o prefixo para recuperar os arquivos: ");  
  
    String recuperarComo = scan.next();  
  
    try {  
  
        rPessoaFisica.recuperar(recuperarComo + ".fisica.bin");  
  
        rPessoaJuridica.recuperar(recuperarComo + ".juridica.bin");  
  
        System.out.println("Dados recuperados com Exito!");  
  
    } catch (IOException | ClassNotFoundException e) {  
  
        System.out.println("Erro na tentativa de recuperar seus dados." + e.getMessage());  
  
    }  
}  
  
}
```