

FUTAPP

É um programa simples que gerencia informações sobre clubes e jogadores de futebol usando a biblioteca JOptionPane para interação com o usuário. Cada parte do código:

Pacote e Importações:

```
java Copy code

package application;
import java.util.ArrayList;
import java.util.List;
import java.util.Locale;
import java.util.Scanner;
import javax.swing.*; // JOptionPane
import java.io.*;
import entities.Jogadores;
import entities.Emprestimo;
import entities.Clube;
import entities.Definitivo;
```

- O programa está no pacote application.
- Importa várias classes e pacotes, incluindo ArrayList, List, Locale, Scanner, JOptionPane, File, e algumas classes definidas pelo usuário (Jogadores, Emprestimo, Clube, Definitivo).

Classe Program:

```
java Copy code

public class Program {
```

- Inicia a classe principal chamada Program.

Método main:

```
java Copy code

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

-  O ponto de entrada do programa.

❓ Configurações

Iniciais:

```
java Copy code

Locale.setDefault(Locale.US);
Scanner sc = new Scanner(System.in);
List<Jogadores> list = new ArrayList<>();
List<Clube> list1 = new ArrayList<>();
```

- Define o local padrão como EUA.
- Cria um objeto Scanner para entrada de dados do console.
- Cria listas para armazenar objetos das classes Jogadores e Clube.

❓ Menu

Principal:

```
java Copy code

do {
    int menu = Integer.parseInt(JOptionPane.showInputDialog("Escolha uma opção:\n"
        + "\n 1 - Cadastrar dados do clube\n"
        + " 2 - Cadastrar jogadores\n"
        + " 3 - Atualizar jogadores\n"
        + " 4 - Atualizar dados do clube\n"
        + " 5 - Remover jogadores\n"
        + " 6 - Listar jogadores\n"
        + " 7 - Listar dados do clube\n"
        + " 8 - Sair\n"
        + "Digite uma opção:"));
```

- O programa entra em um loop do-while exibindo um menu para o usuário usando a classe JOptionPane.

? Switch-Case para Opções do Menu:

```
java Copy code

switch (menu) {
    case 1: // Cadastrar dados do clube
        // ...
        break;

    case 2: // Cadastrar jogadores
        // ...
        break;

    case 3: // Atualizar jogadores
        // ...
        break;

    case 4: // Atualizar dados do clube
        // ...
        break;

    case 5: // Remover jogadores
        // ...
        break;

    case 6: // Listar jogadores
        // ...
        break;

    case 7: // Listar dados do clube
        // ...
        break;

    case 8: // Sair
        JOptionPane.showMessageDialog(null, "Programa encerrado");
        break;

    default:
        JOptionPane.showMessageDialog(null, "Inválido");
        break;
}
```

- Um switch-case é usado para lidar com diferentes opções do menu.

? Loop do-while:

```
java Copy code

} while (menu != 8);
```

- O programa continua executando até que a opção 8 (Sair) seja selecionada.

? Tratamento de Exceções e Fechamento:

```
java Copy code

} catch (java.lang.NumberFormatException e) {
    JOptionPane.showMessageDialog(null, "Inválido");
}

sc.close();
```

- ? Captura exceções de formato numérico.
- ? Fecha o objeto Scanner

Este código faz uso extensivo da classe `JOptionPane` para interação com o usuário, o que é uma abordagem alternativa à entrada de console tradicional. As funcionalidades específicas de cada opção do menu estão implementadas nas respectivas seções do `switch-case`.