



Projeto do bloco I



Conheça a Equipe



Guiherme Sousa

Classe IMC



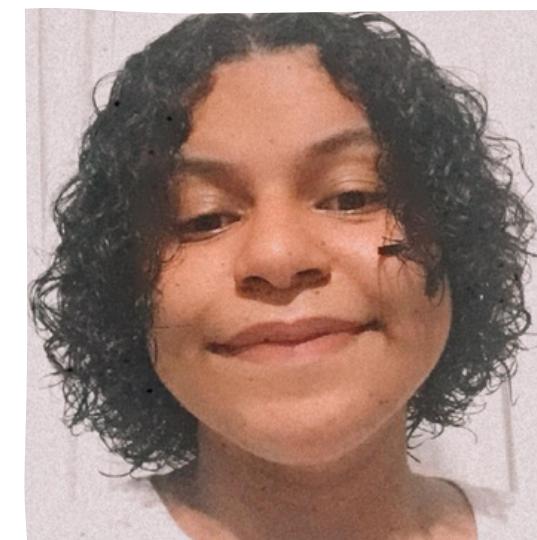
Malcolm Silva

Classe Pessoa



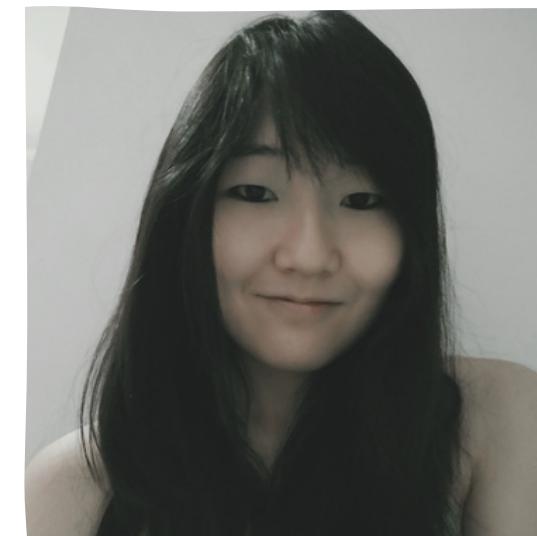
Bruno Pajtak

Classe clinicas



Caroline Calixto

Classe dieta



Helen Lissa Sonoda

Classe main



Problema

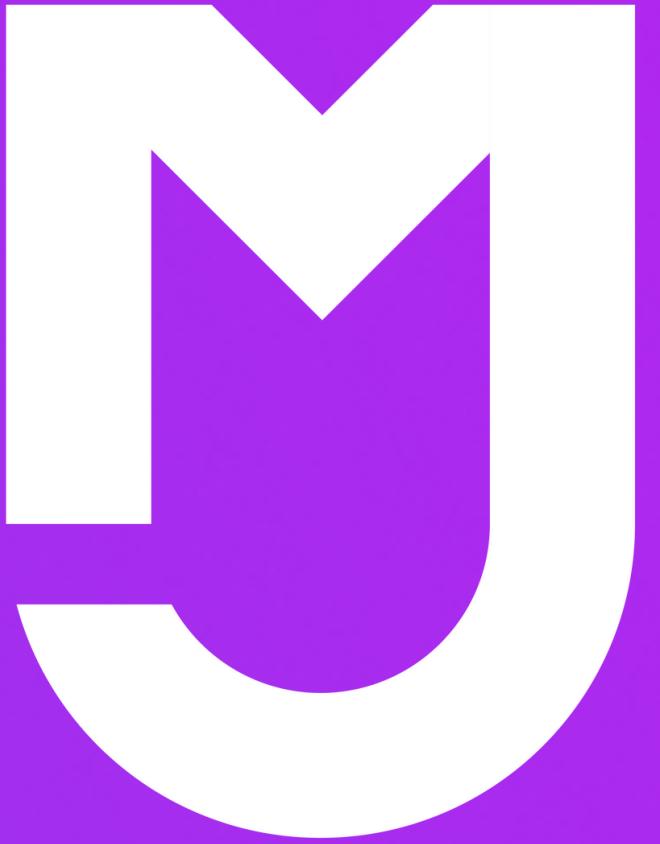
Má alimentação da população

Objetivos da aplicação

Garantir o acesso à saúde de qualidade e promover o bem-estar para todos, em todas as idades



Manjarva é a junção das palavras MANGER, que significa comer em Francês + JAVA que é a linguagem de programação utilizada.



manjarva

Proposta

Nossa principal proposta é conscientizar as pessoas sobre uma alimentação saudável e o cuidado necessário com a saúde de maneira responsável e acolhedora. Não teremos aqui um aplicativo que gira em torno de um ideal de beleza, mas sim, um ideal de saúde.

Funcionalidades para o usuário

- MENU
- CADASTRO DO USUARIO
- COMO CALCULAR IMC
- COMO ESTÁ SUA ALIMENTAÇÃO
- CLINICAS DE NUTRIÇÃO PRÓXIMAS



manjarva

MAIN

Classe principal da aplicação



manjarva

```
1 package br.com.generation.projeto;
2
3 import javax.swing.JOptionPane;
4
5 public class Main {
6
7     public static void main(String[] args) {
8
9         PessoaCaracteristica pc = new PessoaCaracteristica();
10
11        pc.setNome(JOptionPane.showInputDialog("Digite seu nome: "));
12        pc.setIdade(Integer.parseInt(JOptionPane.showInputDialog("Digite sua idade: ")));
13        pc.setSexo(JOptionPane.showInputDialog("Digite seu sexo:\n [M] ou [F]: "));
14        pc.setAltura(Double.parseDouble(JOptionPane.showInputDialog("Digite sua altura: ")));
15        pc.setPeso(Double.parseDouble(JOptionPane.showInputDialog("Digite seu peso: ")));
16
17        JOptionPane.showMessageDialog(null, "***** Dados de " + pc.getNome() +
18            "\nIdade: " + pc.getIdade() + " anos" + "
```

Main (Principal)

A Classe Main é a classe responsável por executar o programa desenvolvido.

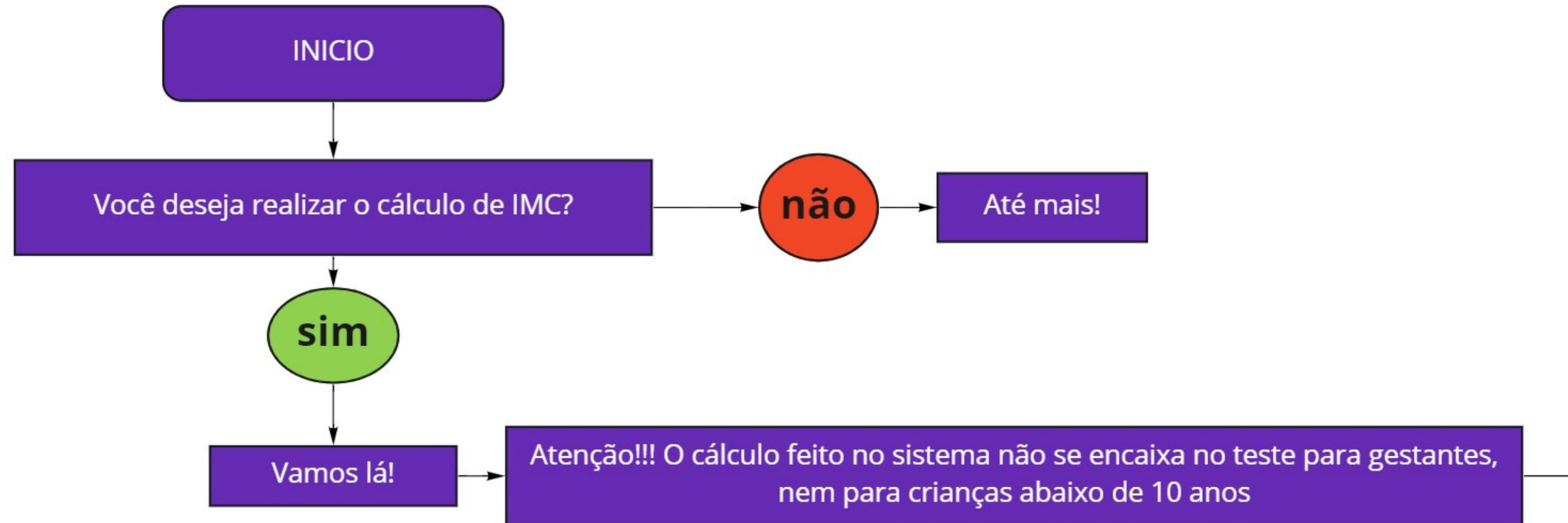
Por conta do manjarva ter outras 4 classes com funcionalidades diferentes, ela foi criada separadamente.



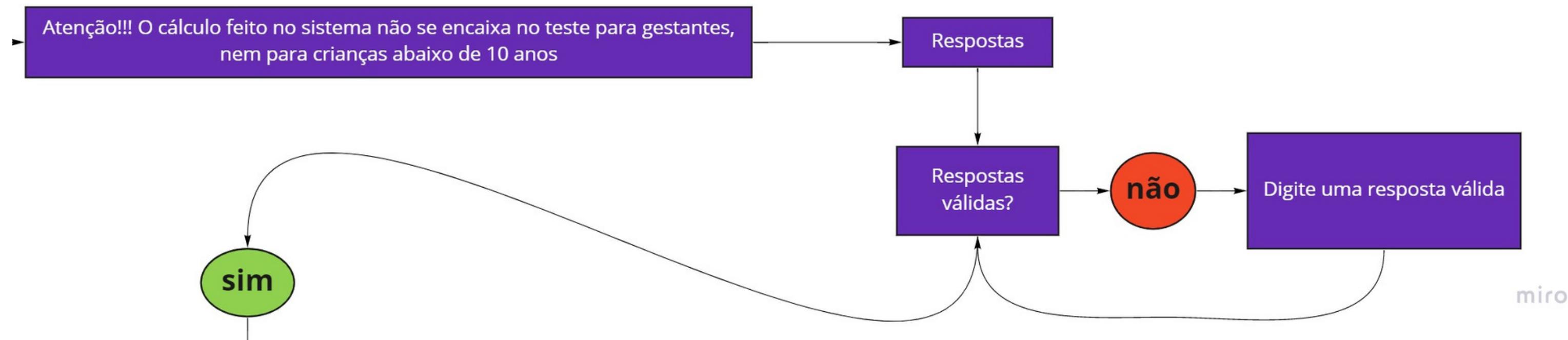
Fluxo de condicionais

Classe MAIN

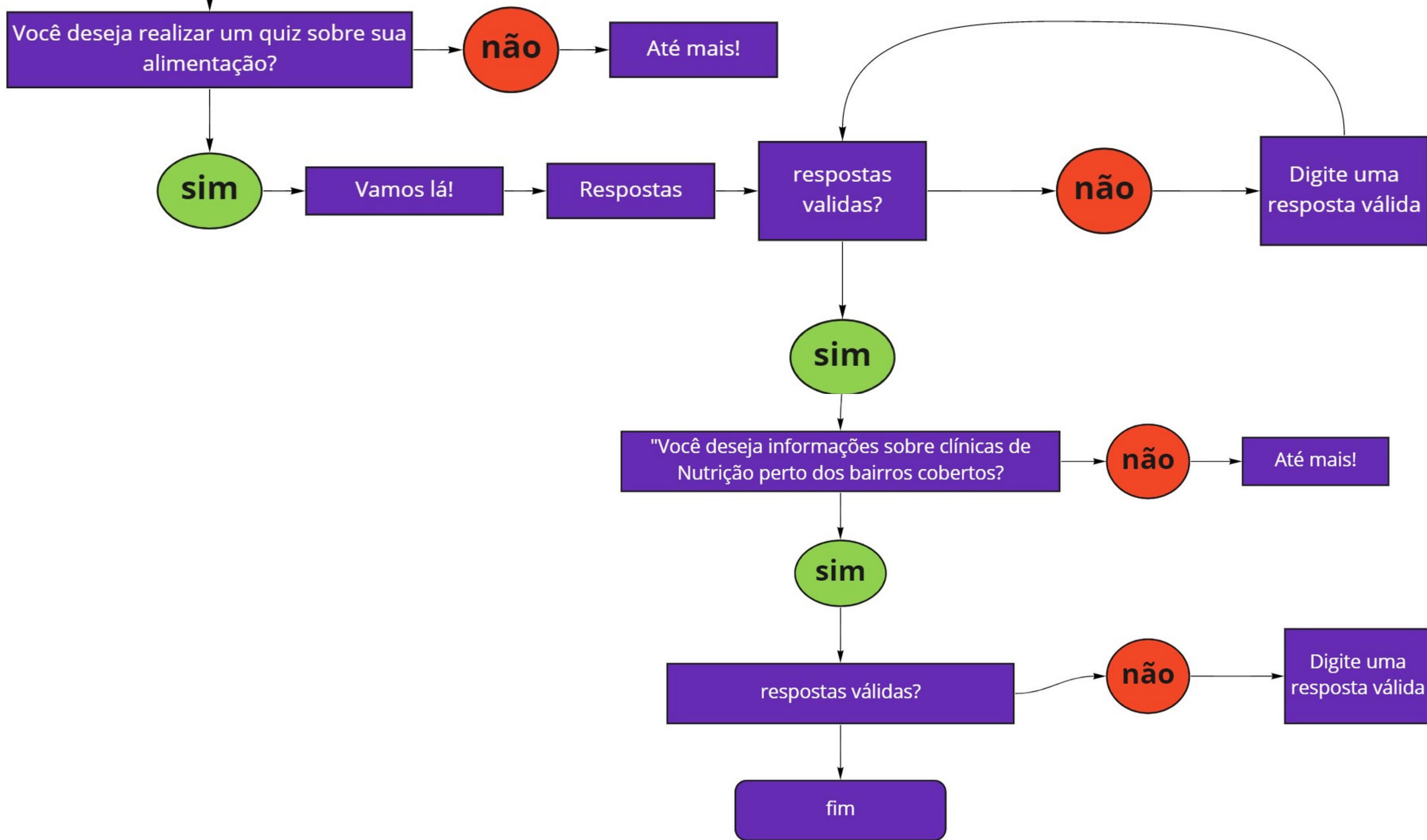
Generation Brasil | Grupo 4



miro



miro



miro

miro

Prints do código



```
1 package br.com.generation.projeto;
2
3 import javax.swing.JOptionPane;
4
5 public class Main {
6
7     public static void main(String[] args) {
8
9         PessoaCaracteristica pc = new PessoaCaracteristica();
10
11        pc.setNome(JOptionPane.showInputDialog("Digite seu nome: "));
12        pc.setIdade(Integer.parseInt(JOptionPane.showInputDialog("Digite sua idade: ")));
13        pc.setSexo(JOptionPane.showInputDialog("Digite seu sexo:\n [M] ou [F]"));
14        pc.setAltura(Double.parseDouble(JOptionPane.showInputDialog("Digite sua altura: ")));
15        pc.setPeso(Double.parseDouble(JOptionPane.showInputDialog("Digite o seu peso: ")));
16
17        JOptionPane.showMessageDialog(null, "***** Dados de " + pc.getNome() + " *****\n\n" +
18            "Idade: " + pc.getIdade() + " anos" +
19            "\nSexo: " + pc.getSexo() +
20            "\nAltura: " + pc.getAltura() + " m" +
21            "\nPeso: " + pc.getPeso() + " kg");
22
23        //Pergunta sobre o cálculo de IMC
24        int imc = JOptionPane.showConfirmDialog(null, "Você deseja realizar o cálculo do IMC?",
25            "Pergunta", JOptionPane.YES_NO_OPTION);
```

Prints do código



```
27     switch(imc) {
28         case 0:
29             JOptionPane.showMessageDialog(null, "Vamos lá!"); //Yes option
30             JOptionPane.showMessageDialog(null, "Atenção!!!\n\n" +
31                 "O cálculo feito no sistema não se\n" +
32                 "encaixa no teste para gestantes, nem\n" +
33                 "para crianças abaixo de 10 anos.\n\n", "Aviso", JOptionPane.INFORMATION_MESSAGE);
34             Imc.validacaoIdade(pc.getIdade(), pc.getPeso(), pc.getAltura(), pc.getSexo());
35             break;
36         case 1:
37             JOptionPane.showMessageDialog(null, "Até mais!"); //No option
38             break;
39         case -1:
40             JOptionPane.showMessageDialog(null, "Até mais!"); //X option
41             break;
42     }
43
44     //Pergunta sobre o questionário de dieta
45     int diet = JOptionPane.showConfirmDialog(null, "Você deseja realizar um quiz sobre sua alimentação?", "Pergunta", JOptionPane.YES_NO_OPTION);
46
47     switch(diet) {
48         case 0:
49             JOptionPane.showMessageDialog(null, "Vamos lá!"); //Yes option
50             Dieta.resultado(Dieta.somaPontos(Dieta.perguntas())));
51             break;
52     }
```

Prints do código



```
53     case 1:  
54         JOptionPane.showMessageDialog(null, "Até mais!"); //No option  
55         break;  
56     case -1:  
57         JOptionPane.showMessageDialog(null, "Até mais!"); //X option  
58         break;  
59     }  
60  
61     //Informações sobre clínicas de Nutrição perto dos bairros cobertos  
62     int clin = JOptionPane.showConfirmDialog(null, "Você deseja informações sobre clínicas"  
63             + "| de Nutrição perto dos bairros cobertos?",  
64             "Pergunta", JOptionPane.YES_NO_OPTION);  
65  
66     switch(clin) {  
67         case 0:  
68             JOptionPane.showMessageDialog(null, "Vamos lá!"); //Yes option  
69             Bairro.GetBairro(args, args);  
70             break;  
71         case 1:  
72             JOptionPane.showMessageDialog(null, "Até mais!"); //No option  
73             break;  
74         case -1:  
75             JOptionPane.showMessageDialog(null, "Até mais!"); //X option  
76             break;  
77     }  
78 }  
79 }
```

INPUT DO USUÁRIO

Classe para input do usuário



manjarva

PessoaCaracterística

- A Classe PessoaCaracterística é uma classe base criada para as outras classes herdarem Atributos/Características em forma de herança.

Prints do código



```
1 package br.com.generation.projeto;
2
3 public class PessoaCaracteristica {
4
5     private String nome;
6     private String sexo;
7     private int idade;
8     private double altura;
9     private double peso;
10
11    public String getNome() {
12        return nome;
13    }
14
15    public void setNome(String nome) {
16        this.nome = nome;
17    }
18
19    public String getSexo() {
20        return sexo;
21    }
22
23    public void setSexo(String sexo) {
24        this.sexo = sexo;
25    }
26
27    public int getIdade() {
28        return idade;
29    }
30
31    public void setIdade(int idade) {
32        this.idade = idade;
33    }
34
35    public double getAltura() {
36        return altura;
37    }
38
39    public void setAltura(double altura) {
40        this.altura = altura;
41    }
42
43    public double getPeso() {
44        return peso;
45    }
46
47    public void setPeso(double peso) {
48        this.peso = peso;
49    }
50
51 }
```

CÁLCULO IMC

Classe para análise do peso do usuário



Para que serve?

O significado de IMC é Índice de Massa Corporal, ele é uma parâmetro utilizado para verificar se uma pessoa está com uma dieta balanceada e se ela está se exercitando o suficiente. Para fornecer um cálculo mais preciso a OMS fornece uma tabela com os valores de imc para adultos

IMC - CLASSIFICAÇÃO:

- Menor que 18.5 - Abaixo do peso ;
- Entre 18.5 e 24.9 - Peso normal ;
- Entre 25.0 e 29.9 - Pré-obesidade ;
- Entre 30.0 e 34.9 - Obesidade Grau 1 ;
- Entre 35.0 e 39.9 - Obesidade Grau 2 ;
- Acima de 40 - Obesidade Grau 3

O cálculo mais comum do imc é $\text{IMC} = \text{peso} / (\text{altura}^*\text{altura})$

Você tem feito atividades físicas recentemente?

Usando o imc e sua tabela como base o aplicativo recomenda algumas ações a serem tomadas, como por exemplo exercícios leves.

Exercícios leves ajudam a emagrecer, mas o mais importante é que ele auxiliam na expansão de sua estamina, além da facilidade de executa-los em qualquer ambiente. Exemplos dessas atividades são:

- Caminhadas
- Corridas
- Polichinelo
- Pular Corda
- BURPEE



manjarva

Como funciona ?

O programa realiza todos os cálculos, quando o usuário entra com as informações relevantes como: sexo, idade, peso e altura. O imc é calculado, o código pega também o sexo e idade para dar uma a resposta de acordo com o perfil do cliente. Por exemplo:

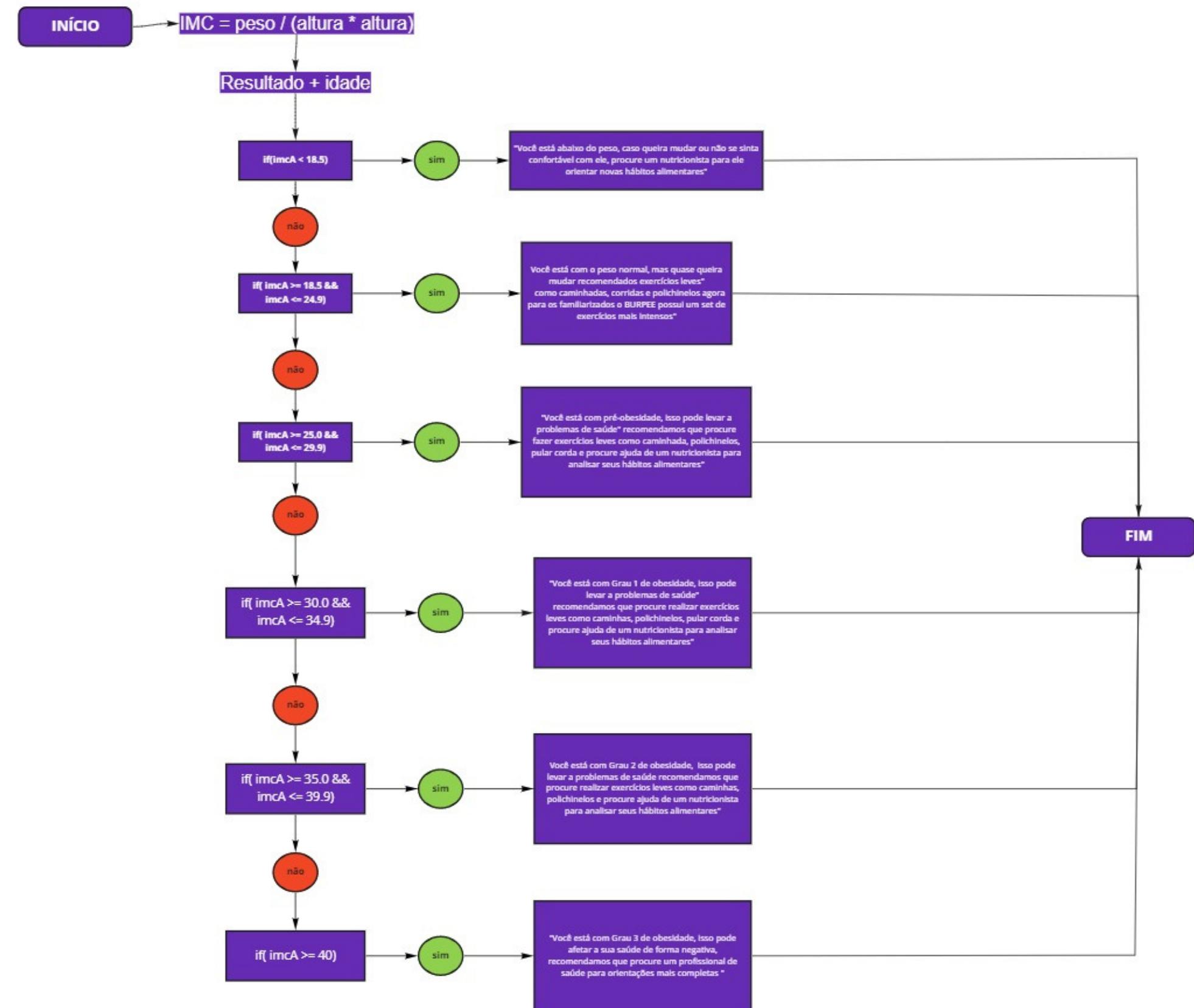
- Você está abaixo do peso, caso queira mudar ou não se sinta confortável com ele, procure um nutricionista para ele orientar novas hábitos alimentares
- Você está com o peso normal, mas quase queira mudar recomendados exercícios leves como caminhadas, corridas e polichinelos agora para os familiarizados o BURPEE possui um set de exercícios mais intensos"
- Você está com pré-obesidade, isso pode levar a problemas de saúde" recomendamos que procure fazer exercícios leves como caminhada, polichinelos, pular corda e procure ajuda de um nutricionista para analisar seus hábitos alimentares



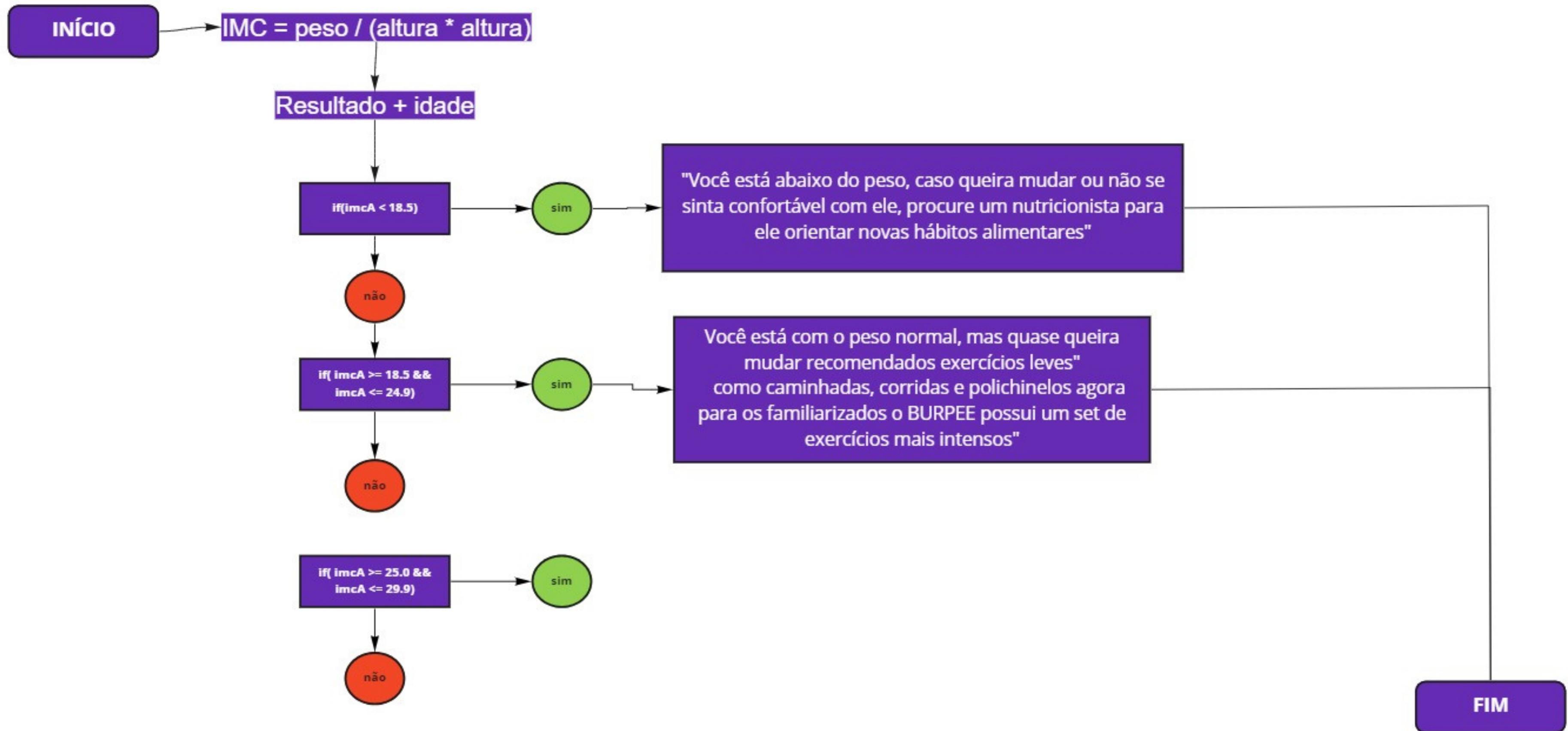
manjarva

Fluxo de condicionais

Exibição dos resultados



miro





Prints do código

```
1/* @authors Guilherme de Sousa Virdiano e Helen Lissa Sonoda*/
4
5 package br.com.generation.projeto;
6
7 import javax.swing.JOptionPane;
8
9 public class Imc {
10
11    static void validacaoIdade(int idade, double peso, double altura, String sexo) {
12
13        if(idade < 10){
14            JOptionPane.showMessageDialog(null, "A idade não é válida para o cálculo de IMC",
15                + "\n\nO cálculo do IMC será encerrado.\n\n",
16                "Aviso", JOptionPane.INFORMATION_MESSAGE);
17        }
18        else {
19            calculoPorIdade(idade, peso, altura, sexo);
20        }
21    }
22}
```

Prints do código

```
22
23•     static void calculoPorIdade(int idade, double peso, double altura, String sexo) {
24
25         if(idade >= 10 && idade <= 19) {
26
27             switch(sexo) {
28                 case "F", "f":
29                     calcularImcAdolecF(idade, peso, altura);
30                     break;
31                 case "M", "m":
32                     calcularImcAdolecM(idade, peso, altura);
33                     break;
34             }
35         }
36     }
37     else if(idade >= 20 && idade < 60){
38         calcularImcAdulto(peso, altura);
39     }
40 }
41 else if(idade >= 60){
42     calcularImcIdoso(peso, altura);
```

Prints do código

```
45
46•static void calcularImcAdolecF(int idade, double p, double a) {
47    // calculo para crianças adolescentes feminino de 10 a 19 anos
48
49    double imcF = p/(a*a);
50
51    switch(idade) {
52        case 10:
53
54        if (imcF < 14.23) {
55            JOptionPane.showMessageDialog(null, " "+ "\nO seu IMC é: " + imcF + "\n" +
56            "\nVocê está abaixo do peso. \n\nCaso queira mudar ou não se sinte confortável com ele, "
57            + "\nprocure um nutricionista, para que ele possa orientar novos hábitos alimentares." +
58            "Informação", JOptionPane.INFORMATION_MESSAGE);
59
60        } else if (imcF >= 14.23 && imcF <= 20.19) {
61            JOptionPane.showMessageDialog(null, " "+ "\nO seu IMC é: " + imcF + "\n" +
62            "\nVocê está com o peso normal. \n\nCaso queira mudar ou não está confortável com ele, "
63            + "\nvocê pode realizar exercícios leves, como caminhadas, polichinelos ou pular corda."
64            "Informação", JOptionPane.INFORMATION_MESSAGE);
65
```

Prints do código

```
65
66     } else {
67         JOptionPane.showMessageDialog(null, "
68         \"\nVocê está com sobrepeso. \n\nIsso pode ser prejudicial à sua saú
69         +"\nÉ recomendável que entre em contato com um profissional de saú
70         + "\npara orientá-lo sobre quais medidas você deve tomar.\n\nSe vo
71         + "\ne você deseja mudar, recomendamos que faça exercícios regulares
72         + "\ne polichinelos para melhorar o seu condicionamento físico.\n\n
73         "Informação", JOptionPane.INFORMATION_MESSAGE);
74
75     }
76     break;
```

ALIMENTAÇÃO

Exibição da classe dieta

Como está sua alimentação

Alimentação saudável é uma dieta bem equilibrada, que fornece os nutrientes, vitaminas e minerais nas proporções adequadas para o bom funcionamento do organismo.

Para que uma alimentação seja saudável, é importante ter variedade, equilíbrio, controle de quantidade (comer o que lhe é adequado) e qualidade dos alimentos.

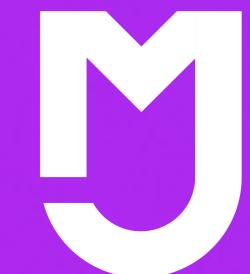




Como está sua alimentação

Para elaboração da classe dieta, foram escritas 11 perguntas para identificação do padrão da alimentação do usuário referenciada ao Guia Alimentar para a População Brasileira ed 2014 e com base na soma de seus pontos, indicado uma recomendação:

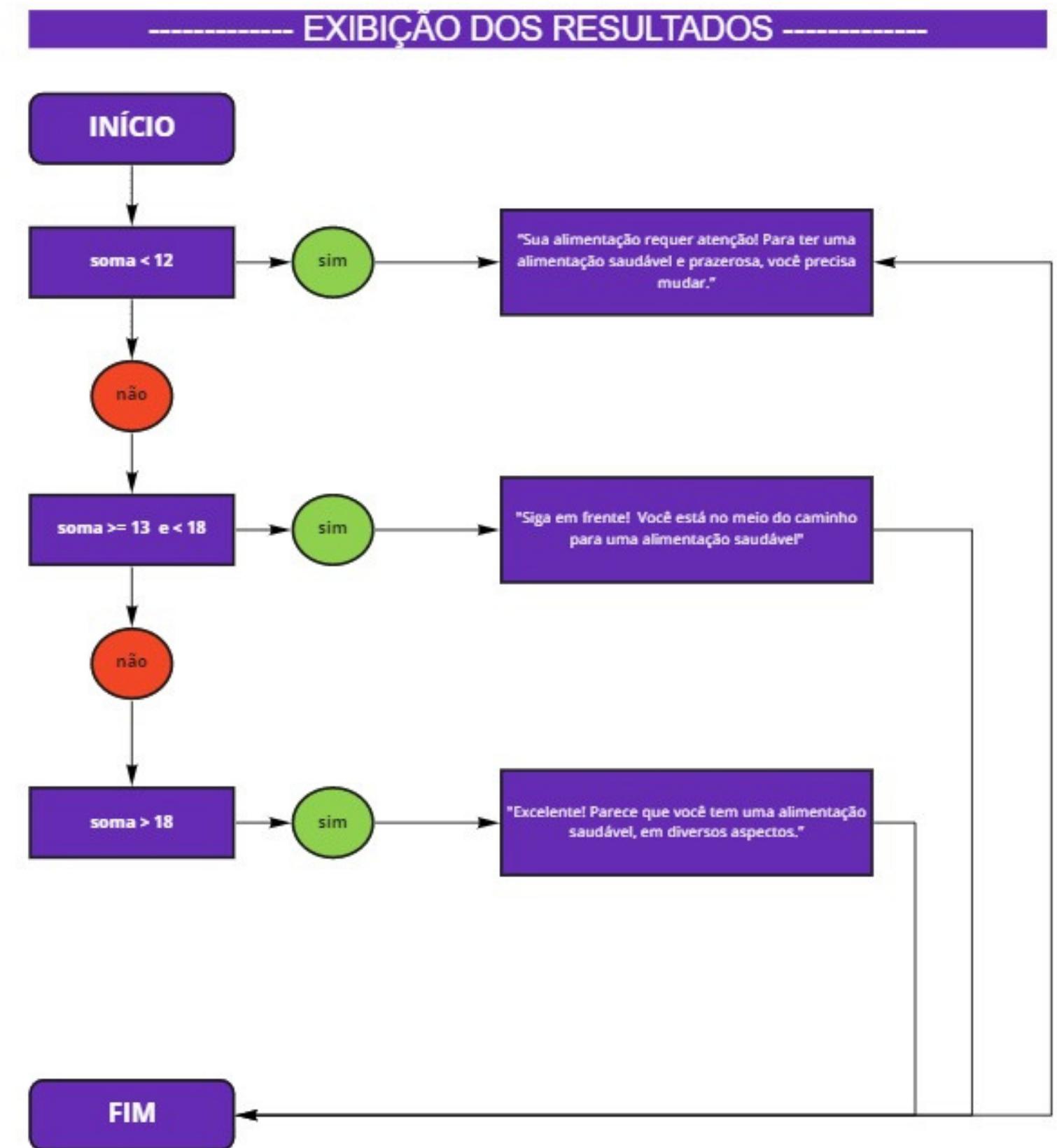
- "Sua alimentação requer atenção! Para ter uma alimentação saudável e prazerosa, você precisa mudar."
- "Siga em frente! Você está no meio do caminho para uma alimentação saudável"
- "Excelente! Parece que você tem uma alimentação saudável, em diversos aspectos."



manjarva

Fluxo de condicionais

Exibição dos resultados

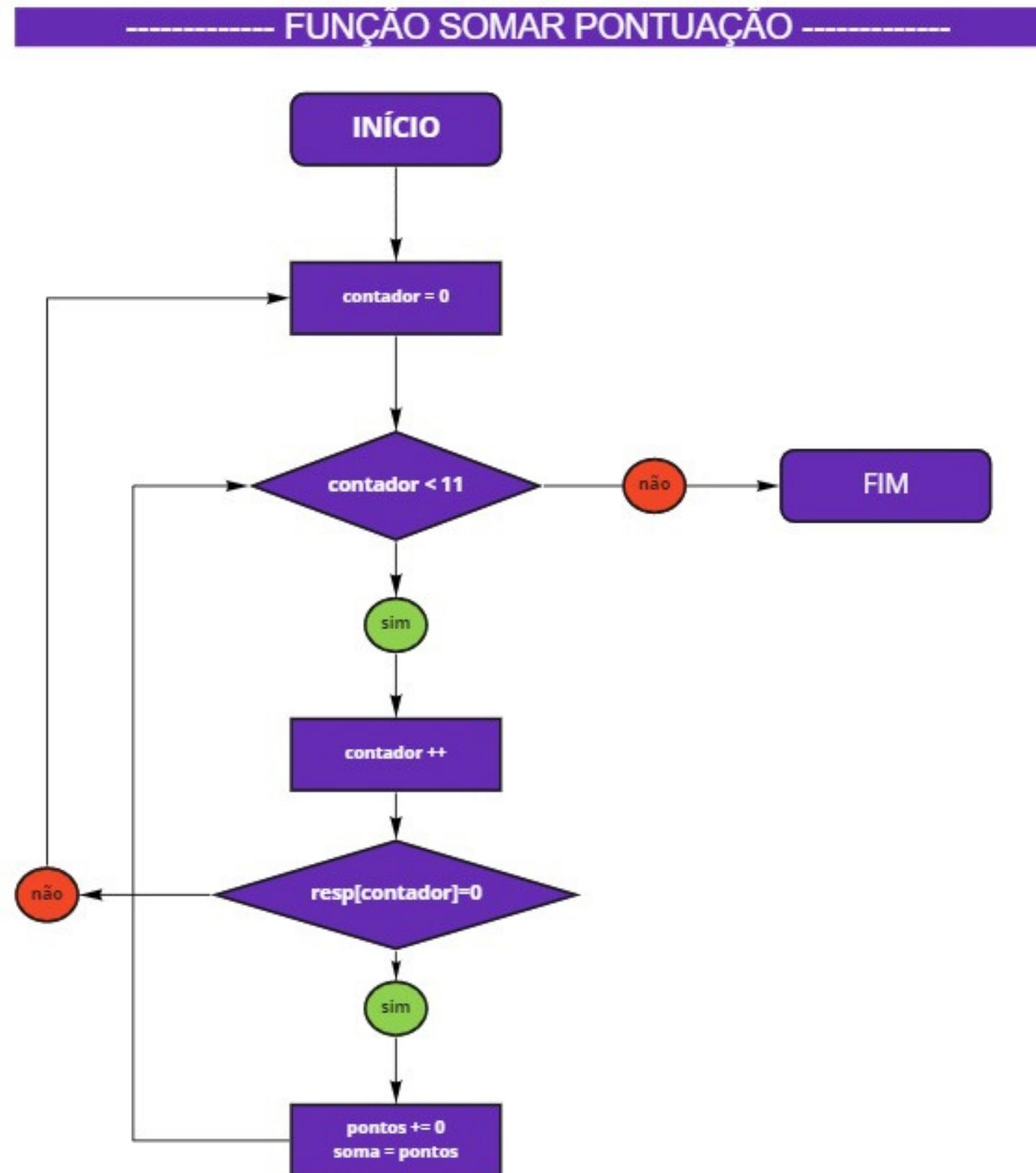




manjarva

Fluxo de condicionais

Somar pontuação



Prints do código

```
9  */
10
11 package br.com.generation.ProjetoBloco1;
12
13 import javax.swing.JOptionPane;
14
15 public class Dieta {
16
17     public static int[] perguntas() {
18
19         int resp[] = new int[11];
20
21         // ----- BANCO DE QUESTÕES -----
22
23
24         resp[0] = Integer.parseInt(
25             JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
26                                         + "\n\n1 - Quando faço pequenos lanches ao longo do dia, costumo comer frutas ou castanhas"
27                                         + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
28
29
30         resp[1] = Integer.parseInt(JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
31                                         + "\n\n2 - Quando escolho frutas, verduras e legumes, dou preferência para aqueles que são de produção local"
32                                         + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
33
34         resp[2] = Integer.parseInt(JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
35                                         + "\n\n3 - Quando escolho frutas, legumes e verduras, dou preferência para aqueles que são orgânicos"
36                                         + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
37
38         resp[3] = Integer.parseInt(
39             JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
40                                         + "\n\n4 - Costumo levar algum alimento comigo em caso de sentir fome ao longo do dia"
41                                         + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
42
43         resp[4] = Integer.parseInt(
```

Prints do código



```
40         + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
41
42     resp[4] = Integer.parseInt(
43         JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
44             + "\n\n5 - Costumo planejar as refeições que farei no dia"
45             + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
46
47     resp[5] = Integer.parseInt(
48         JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
49             + "\n\n6 - Costumo variar o consumo de feijão por ervilha, lentilha ou grão de bico"
50             + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
51
52     resp[6] = Integer.parseInt(
53         JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
54             + "\n\n7 - Na minha casa é comum usarmos farinha de trigo integral"
55             + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
56
57
58     resp[7] = Integer.parseInt(
59         JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
60             + "\n\n8 - Costumo comer fruta no café da manhã"
61             + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
62
63
64     resp[8] = Integer.parseInt(
65         JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
66             + "\n\n9 - Costumo fazer minhas refeições sentado(a) à mesa."
67             + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre"));
68
69
70     resp[9] = Integer.parseInt(
71         JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
72             + "\n\n10 - Procuro realizar as refeições com calma. ")
```

Prints do código



```
74
75
76     resp[10] = Integer.parseInt(
77             JOptionPane.showInputDialog("Responda as perguntas a seguir de acordo com seu dia a dia."
78                     + "\n\n11 - Costumo participar do preparo dos alimentos na minha casa"
79                     + "\n[0] Nunca [1] Raramente [2] Muitas vezes [3] Sempre")));
80
81
82     return resp;
83
84
85     // ----- FIM BANCO DE QUESTÕES -----
86 }
87
88 public static int somaPontos(int[] resp) {
89
90     // ----- FUNÇÃO SOMAR PONTUAÇÃO -----
91
92     int c, pontos = 0, soma = 0;
93
94     for (c = 0; c < 11; c++) {
95
96         if (resp[c] == 0) {
97             pontos += 0;
98             soma = pontos;
99
100        }
101
102        else if (resp[c] == 1) {
103            pontos += 1;
104            soma = pontos;
105        }
106    }
```

Prints do código



```
105
106
107     else if (resp[c] == 2) {
108         pontos += 2;
109         soma = pontos;
110     }
111
112     else if (resp[c] == 3) {
113         pontos += 3;
114         soma = pontos;
115     }
116
117 }
118 return soma; // ----- FIM SOMA DA PONTUAÇÃO -----
119 }
120
121 public static void resultado(int soma) {
122
123     // ----- EXIBIÇÃO DOS RESULTADOS -----
124
125     JOptionPane.showMessageDialog(null, "O resultado do seu teste retornou o total de " + soma + " pontos.");
126
127     if (soma < 12) {
128
129         JOptionPane.showMessageDialog(null, "Sua alimentação requer atenção! \n"
130             + "Para ter uma alimentação saudável e prazerosa, você precisa mudar.");
131
132     }
133
134     else if (soma >= 13 && soma < 18) {
135
136         JOptionPane.showMessageDialog(null,
```

Prints do código



manjarva

```
117     }
118     return soma; // ----- FIM SOMA DA PONTUAÇÃO -----
119 }
120
121 public static void resultado(int soma) {
122     // ----- EXIBIÇÃO DOS RESULTADOS -----
123
124     JOptionPane.showMessageDialog(null, "O resultado do seu teste retornou o total de " + soma + " pontos.");
125
126     if (soma < 12) {
127
128         JOptionPane.showMessageDialog(null, "Sua alimentação requer atenção! \n"
129             + "Para ter uma alimentação saudável e prazerosa, você precisa mudar.");
130
131     }
132
133     else if (soma >= 13 && soma < 18) {
134
135         JOptionPane.showMessageDialog(null,
136             "Siga em frente! \n" + "Você está no meio do caminho para uma alimentação saudável");
137
138     }
139
140     else if (soma > 18) {
141
142         JOptionPane.showMessageDialog(null,
143             "Excelente! \n" + "Parece que você tem uma alimentação saudável, em diversos aspectos.");
144
145     }
146     // ----- FIM EXIBIÇÃO DOS RESULTADOS -----
147 }
148 }
149 }
```

CLÍNICAS

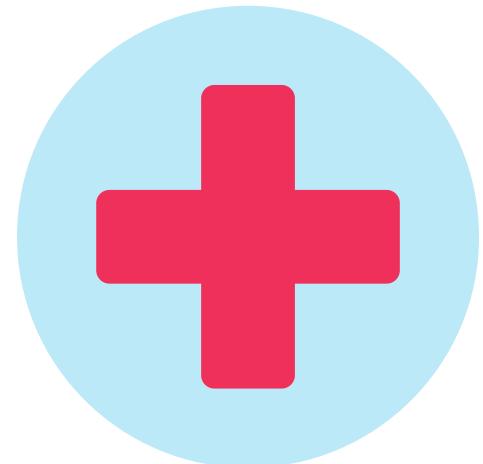
Exibição da classe de Clínicas



Caso o usuário deseje ajuda Profissional

Caso a dieta do usuário e o seu IMC estejam críticos, o usuário será recomendado à procurar ajuda profissional para melhorar a sua saúde e dieta.

Se o usuário não desejar procurar nenhuma ajuda profissional o aplicativo encerrará, no entanto, caso ele deseje uma nova aba vai surgir perguntando qual o bairro (coberto pelo MANJARVA) que o cliente mora.





Cobertura do Aplicativo

Atualmente o aplicativo cobre 4(Quatro) bairros da cidade de São Paulo. São eles: Tatuapé, Mooca, Belém e Vila Carrão.

Para encontrar as clínicas presentes no seu bairro, o usuário escolherá um dos bairros na barra de opções e o aplicativo vai retornar a lista de Endereços de Clínicas existentes no seu bairro!



Prints do código



manjarva

```
lic class Bairro {  
  
private Map<String, List<String>> end = new HashMap<String, List<String>>();  
  
public Bairro(){  
List<String> tatuapeList = new ArrayList<String>();  
tatuapeList.add("Nutriclin|| Endereço: Rua Apucarana, 326. Telefone: (11) 3798-2829");  
tatuapeList.add("Clínica Equilíbrio Nutricional || Endereço: Praça Silvio Romero, 55 - sala 67. Telefone: (11) 2672-7300");  
tatuapeList.add("SF Nutrição || Endereço: R. Itapura, 300 - Conjunto 1302. Telefone: (11) 3713-2023");  
tatuapeList.add("CliNutri || Endereço: Rua Doutor Corinto Baldoino Costa, 74. Telefone: (11) 98224-6063");  
tatuapeList.add("NutreClínica || Endereço: R. Restinga, 113 - conjunto 305. Telefone: (11) 4114-9305");  
end.put("Tatuape", tatuapeList);  
List<String> vilaCarraoList = new ArrayList<String>();  
vilaCarraoList.add("Nutricionista Dra. Brenda Berlla || Endereço: Av. Conselheiro Carrão, 2620 - sala 1203. Telefone: (11) 98420-1266");  
vilaCarraoList.add("Clinica Bio Ser || Endereço: Av. Conselheiro Carrão, 2939. Telefone: (11)2098-2971");  
vilaCarraoList.add("Nutricionista Dra. Renata Giudice || Endereço: Av. Guilherme Giorgi, 374. Telefone: (11) 99246-3632");  
vilaCarraoList.add("Evs - Espaço Vida Saudável || Endereço: Av. Conselheiro Carrão, 1857 - Sala 2. Telefone: (11) 97404-2367");  
end.put("Vila Carrão", vilaCarraoList);  
List<String> belemList = new ArrayList<String>();  
belemList.add("Phillipe Rios - Nutricionista Esporte e Estética || Endereço: Rua Serra da Bocaina, 570 - Cj 41. Telefone: (11) 95892-8819");  
belemList.add("Herba Saúde Nutricional || Endereço: Av. Celso Garcia, 1907. Telefone: (11) 97952-8499");  
end.put("Belém", belemList);  
List<String> moocaList = new ArrayList<String>();  
moocaList.add("Espaço Saúde À Vida || Endereço: R. Canuto Saraiva, 59. Telefone: (11) 98707-8323");  
moocaList.add("NutriMente Psicologia, Neuropsicologia e Nutrição || Endereço: R. da Mooca, 2343 - Sala 1. Telefone: (11) 99515-3854");  
moocaList.add("Andréia Rodrigues - Nutrição Esportiva || Endereço: Rua Camé, 532. Telefone: (11) 98876-3513");  
moocaList.add("Thaís Pereira - Nutricionista Esportiva || Endereço: Rua Celso de Azevedo Marques, 395. Telefone: (11) 98488-0651");  
moocaList.add("Bella Nutri - Nutrição & Estética || Endereço: Edifício Atrio Giorno - R. Borges de Figueiredo, 303 - Sala 609. Telefone: (11) 97457-4444");  
end.put("Mooca", moocaList);  
}
```

Prints do código



```
public Map<String, List<String>> getEnd() {
    return end;
}
public static void GetBairro(String[] args, Object[] Categories) {
    boolean terminar = false;
    while (!terminar) {
        Object endereco;
        Scanner scanner = new Scanner(System.in);
        Bairro bairros = new Bairro();
        endereco = JOptionPane.showInputDialog(null, "Digite aqui um dos Bairros cobertos pelo MANJARVA: \n[Tatuapé]\n[Belém]\n[Mooca]\n[Vila Carrão]", "AVISO", JOptionPane.INFORMATION_MESSAGE);
        List<String> resultado = bairros.getEnd().get(endereco);

        if (resultado == null) {

            JOptionPane.showMessageDialog(null, "O resultado não foi encontrado, por favor digite um dos bairros cobertos pelo aplicativo: ", "AVISO", JOptionPane.INFORMATION_MESSAGE);
        }
        else {
            for (String obj: resultado) {
                JOptionPane.showMessageDialog(null, obj, "RESULTADO", JOptionPane.INFORMATION_MESSAGE);
            }
            terminar = true;
        }

        scanner.close();
    }
}

/*Map<String, List<String>> hm = new HashMap<String, List<String>>();
List<String> values = new ArrayList<String>();
values.add("Value 1");
values.add("Value 2");
hm.put("Key1", values);*/
```



manjarva

Melhorias / Propostas Futuras

- Alterar as opções de Yes / No para português Sim / Não
 - Arredondar valores do IMC
- Implementar validação do campo de input para apenas números para o quiz de alimentação
- Implementar validação do campo de input para apenas números para os dados de idade, peso, altura
- Implementar validação do campo de input para apenas letras para o nome e sexo
 - Expandir o sistema para atender mais bairros e regiões de São Paulo



OBRIGADO À
TODES

Generation Brasil | Grupo 4