

TEMA PROJETO: Cadeia de Restaurantes(Bob's ou Subway)

Discente: João Batista Araújo de Lima Filho.

Classe RepositorioSanduiche

```
import java.util.ArrayList;
```

```
public class RepositorioSanduiche {
```

```
// I - Complementando o que você fez até agora no projeto. Crie um repositório de uma das
//classes do seu projeto utilizando ArrayList. Você deve utilizar uma classe criada em
//atividades anteriores que possua no mínimo dois atributos, um destes atributos deve ser
do
//tipo String. Você deve criar a classe RepositorioTipo, em que "Tipo" é o nome da classe do
//seu projeto (vou utilizar o nome Tipo porque cada projeto tem um nome diferente). Sua
//classe repositório deve ter os métodos descritos abaixo, cada um deles deve ser testado
//dentro do método main() da classe mostrando a saída da chamada destes métodos na
//tela.
```

“Aqui foi criado um repositorio de sanduiches usando ArrayList”

```
RepositorioSanduiche(){
    sandwiches.add(new Sanduiche("Double Cheeseburger","Pequeno"));
    sandwiches.add(new Sanduiche("Frango","Medio"));
    sandwiches.add(new Sanduiche("Big Bob", "Grande"));
    sandwiches.add(new Sanduiche("Crispy Bacon","Pequeno"));
    sandwiches.add(new Sanduiche("Frango Tomato Artesanal"));
    sandwiches.add(new Sanduiche("Cheddar Australiano","Grande"));
    sandwiches.add(new Sanduiche("Big Bob Frango","Grande"));
    sandwiches.add(new Sanduiche("Tentador Frango","Pequeno"));
    sandwiches.add(new Sanduiche("Tentador Carne","Medio"));
    sandwiches.add(new Sanduiche("Tentador Zero Beef","Pequeno"));
    sandwiches.add(new Sanduiche("Bobs Artesanal","Pequeno"));
    sandwiches.add(new Sanduiche("Costela Artesanal","Grande"));
    sandwiches.add(new Sanduiche("Cheeseburger","Pequeno"));
}
```

“Esse é o ArrayList de Sanduiches, para armazenar o repositorio de sanduiches”

```
ArrayList<Sanduiche> sandwiches = new ArrayList<Sanduiche>();
```

```
public ArrayList<Sanduiche> getSanduiches() {
    return sandwiches;
}
```

“Esse metodo getSanduiches retorna os sanduiches”

```
// I - a) procurar(String id), este método deve retornar o objeto que possui o valor
//do seu atributo String igual a id. Faça testes exibindo no terminal uma descrição do
//objeto encontrado com todos os seus atributos. Caso o objeto não seja encontrado, você
//deve mostrar uma mensagem na tela comunicando este fato.
```

“Esse metodo procura a String nomeSanduiche no repositorio sanduiches e retorna se encontrou ou não e se encontrou ele mostra as informações do sanduiche”

```
boolean procurar(String nomeSanduiche) {  
    boolean encontrou = false;  
    for (Sanduiche i: sanduiches) {  
        if (i.nomeSanduiche.equals(nomeSanduiche)) {  
            System.out.println(i.toString());  
            encontrou = true;  
        }  
    }  
    if (!encontrou) {  
        System.out.println("Sanduiche "+nomeSanduiche+" não encontrado!");  
    }  
    return encontrou;  
}
```

// I - b) inserir(Tipo tipo), este método deve armazenar uma nova instância da classe com a seguinte restrição: nova instância só pode ser inserida se não houver no repositório uma outra instância com mesmo id. Após inserir mostra uma mensagem na tela confirmando a inserção. Caso não seja possível inserir mostra uma mensagem na tela informando que não foi possível pois esta instância já existe no repositório.

“Esse metodo recebe um sanduiche e verifica se ele esta no repositorio e se não tiver ele insere o sanduiche no repositorio, caso contrario ele informa que não pode inserir pois o sanduiche já esta no repositorio”

```
void inserir(Sanduiche sanduiche) {  
    boolean encontrou = false;  
    for (Sanduiche i: sanduiches) {  
        if (i.nomeSanduiche.equals(sanduiche.nomeSanduiche)) {  
            System.out.println("Não foi possivel inserir o Sanduiche  
"+sanduiche.nomeSanduiche+  
            " pois ele ja existe no repositorio!");  
            encontrou = true;  
        }  
    }  
    if (!encontrou) {  
        sanduiches.add(sanduiche);  
        System.out.println("O Sanduiche "+sanduiche.nomeSanduiche+" foi inserido no  
repositorio");  
    }  
}
```

// I - c) remover(String id), este método busca no repositório o objeto com o id passado como parâmetro e: (i) se encontra o objeto o remove da base mostra uma mensagem informando que o objeto foi removido; (ii) se não encontra informa que o objeto não existe na base. Após a mensagem, em ambos os casos, mostra na tela todos os objetos do repositório.

“Esse metodo recebe o nome do sanduiche e verifica se ele esta no repositorio, caso esteja ele remove o sanduiche do repositorio e mostra os que sobraram, e caso não seja encontrado retorna que não deu pra remover pois não se encontra no repositorio e mostra os objetos todos os objetos do repositorio”

```
boolean remover(String nomeSanduiche) {  
  
    for (Sanduiche i : sanduiches) {
```

```

        if (i.nomeSanduiche.equals(nomeSanduiche)) {
            sandwiches.remove(i);
            System.out.println("\n-----");
            System.out.println("O Sanduiche "+nomeSanduiche+" foi removido do repositorio\n");
            System.out.println(" SANDUICHES NO REPOSITORIO APOS REMOÇÃO:\n");
            mostrar(sandwiches);
            return true;
        }
    }
    System.out.println("\n-----");
    System.out.println("\nO Sanduiche "+nomeSanduiche+" não foi removido pois não se encontra no repositorio\n");
    System.out.println(" SANDUICHES NO REPOSITORIO QUE PODEM SER REMOVIDOS :\n");
    mostrar(sandwiches);
    return false;
}

```

“Esse metodo mostra todos os sanduiches do repositorio”

```

public static void mostrar (ArrayList<Sanduiche> lista) {
    for (Sanduiche i: lista) {
        System.out.println(i.nomeSanduiche);
    }
}

```

```

public static void main(String[] args) {

```

```

    System.out.println("SISTEMA DE REDE DE RESTAURANTES BOB'S\n");

```

```

    System.out.println("----- QUESTÃO 1 ----- \n");

```

“Aqui foi criado uma instancia da classe RepositorioSanduiche”

```

    RepositorioSanduiche r = new RepositorioSanduiche();

```

```

    // inserir(Sanduiche sanduiche)

```

“Aqui o metodo insere os sanduiches no repositorio”

```

    System.out.println("==*==*==*==*==*==*==*==*==*==*==*==*==* SAIDA METODO INSERIR
==*==*==*==*==*==*==*==*==*==*==*==*==*\n    ");

```

```

    r.inserir(new Sanduiche("Frango Tomato Artesanal","Medio"));

```

```

    r.inserir(new Sanduiche("Chicken Crispy","Grande"));

```

```

    r.inserir(new Sanduiche("Parmesao Artesanal","Pequeno"));

```

```

    r.inserir(new Sanduiche("Crispy Bacon"));

```

```

    r.inserir(new Sanduiche("Tentador Zero Frango","Grande"));

```

```

    r.inserir(new Sanduiche("Big Cheddar Bacon","Grande"));

```

```

    // procurar(String nomeSanduiche)

```

“Aqui o metodo procura os sanduiches no repositorio”

```

    System.out.println("\n==*==*==*==*==*==*==*==*==*==*==*==*==* SAIDA METODO
PROCURAR ==*==*==*==*==*==*==*==*==*==*==*==*==*\n");

```

```

    r.procurar("Frango Tomato Artesanal");

```

```

    r.procurar("Parmegiana Artesanal");

```

```

    r.procurar("Tentador Carne");

```

```

    // remover(String nomeSanduiche)

```

“Aqui o metodo remove os sanduiches no repositorio”

```

    System.out.println("\n==*==*==*==*==*==*==*==*==*==*==*==*==* SAIDA METODO
REMOVER ==*==*==*==*==*==*==*==*==*==*==*==*==*");

```

```

r.remove("Bobs Artesanal");
r.remove("Big");
r.remove("Frango");

// II - Crie dois vetores multidimensionais (com, pelo menos, três linhas e três
//colunas) de uma classe "Tipo" (vou utilizar o nome Tipo porque cada projeto tem um
nome
//diferente). Você deve utilizar uma classe criada em atividades anteriores que possua no
//mínimo dois atributos, um destes atributos deve ser do tipo String. Utilize dois laços " for
"
//aninhados para responder esta questão. Mostre na tela o vetor resultante da
concatenação
//das Strings nas posições correspondentes em cada vetor multidimensional. A saída na
tela
//deve ser exibida em linhas e colunas, conforme a estrutura de dados subjacente.

System.out.println("\n***** SAIDA QUESTAO 2
*****");

"Aqui é criado dois vetores 3x3 do tipo Sanduiche"

Sanduiche[][] s = { {new Sanduiche("Frango","Grande"),new Sanduiche("Bacon"),
    new Sanduiche("Cheddar Bacon","Pequeno")},{new Sanduiche("Creme Cheese",
    "Medio"), new Sanduiche("Caipira","Grande"), new Sanduiche("" +
    "Vegano","Pequeno")},{new Sanduiche("Big Bob","Grande"),new
Sanduiche("Crispy"),
    new Sanduiche("Big Beef","Pequeno")}};

Sanduiche[][] s2 = { {new Sanduiche("Frango Acebolado","Grande"),new
Sanduiche("Bacon Artesanal"),
    new Sanduiche("Big Frango","Pequeno")},{new Sanduiche("Bobs Salada",
    "Medio"), new Sanduiche("Cheddar Australiano","Grande"), new Sanduiche("" +
    "Costela Artesanal","Pequeno")},{new Sanduiche("Frango Caipira","Grande"),new
Sanduiche("Barbecue"),
    new Sanduiche("Parmesao Artesanal","Pequeno")}};

"Aqui foi criado um vetor 3x3 para receber a concatenação dos dois vetores anteriores"

String[][] conc = new String[3][3];

"Aqui foi criado dois for aninhado para responder a questão"

System.out.println("Sanduiches do Dia 1(por pedido):\n");

for (int i = 0; i < s.length; i++) {
    for (int j = 0; j < s[i].length ; j++) {

        if ((i == 0 && j==0) || (i==1 && j==0) || (i==2 && j==0)) {
            System.out.print("P" + (i + 1) + " = ");
        }

        System.out.print(s[i][j].nomeSanduiche+" ", "");
        if (j == 2) {
            System.out.println();
        }
    }
}
}

```

```
System.out.println("\nSandwiches do Dia 2(por pedido):\n");
```

```
for (int i = 0; i < s.length; i++) {  
    for (int j = 0; j < s[i].length ; j++) {  
  
        if ((i == 0 && j==0) || (i==1 && j==0) || (i==2 && j==0)) {  
            System.out.print("P" + (i + 1) + " = ");  
        }  
  
        System.out.print(s2[i][j].nomeSanduiche+" , ");  
        if (j == 2) {  
            System.out.println();  
        }  
    }  
}
```

```
System.out.println("\nSandwiches do Dia 3(por pedido): (CONCATENADOS DIA 1 E 2)\n");
```

```
for (int i = 0; i < s.length; i++) {  
    for (int j = 0; j < s[i].length ; j++) {  
  
        conc[i][j] = s[i][j].nomeSanduiche +", "+ s2[i][j].nomeSanduiche ;  
  
        if ((i == 0 && j==0) || (i==1 && j==0) || (i==2 && j==0)) {  
            System.out.print("P" + (i + 1) + " = ");  
        }  
  
        System.out.print(conc[i][j]+" , ");  
        if (j == 2) {  
            System.out.println();  
        }  
    }  
}  
  
}  
  
}
```

IMAGENS DAS SAÍDAS NO CONSOLE

```
Run: RepositorioSanduiche x
/usr/lib/jvm/java-11-openjdk-amd64/bin/java -javaagent:/snap/intellij-idea-community/352/lib/idea_r
SISTEMA DE REDE DE RESTAURANTES BOB'S

----- QUESTÃO 1 -----

***** SAIDA METODO INSERIR *****

Não foi possivel inserir o Sanduiche Frango Tomato Artesanal pois ele ja existe no repositorio!
O Sanduiche Chicken Crispy foi inserido no repositorio
O Sanduiche Parmesao Artesanal foi inserido no repositorio
Não foi possivel inserir o Sanduiche Crispy Bacon pois ele ja existe no repositorio!
O Sanduiche Tentador Zero Frango foi inserido no repositorio
O Sanduiche Big Cheddar Bacon foi inserido no repositorio

***** SAIDA METODO PROCURAR *****

Sanduiche encontrado: Frango Tomato Artesanal, tamanho: Medio, preco: R$14.99
Sanduiche Parmegiana Artesanal não encontrado!
Sanduiche encontrado: Tentador Carne, tamanho: Medio, preco: R$9.99
```

```
***** SAIDA METODO REMOVER *****

-----

O Sanduiche Bobs Artesanal foi removido do repositorio

SANDUICHES NO REPOSITORIO APOS REMOÇÃO:

Double Cheeseburger
Frango
Big Bob
Crispy Bacon
Frango Tomato Artesanal
Cheddar Australiano
Big Bob Frango
Tentador Frango
Tentador Carne
Tentador Zero Beef
Costela Artesanal
Cheeseburger
Chicken Crispy
Parmesao Artesanal
Tentador Zero Frango
Big Cheddar Bacon

-----
```

O Sanduiche Big não foi removido pois não se encontra no repositório

SANDUICHES NO REPOSITÓRIO QUE PODEM SER REMOVIDOS :

Double Cheeseburger
Frango
Big Bob
Crispy Bacon
Frango Tomato Artesanal
Cheddar Australiano
Big Bob Frango
Tentador Frango
Tentador Carne
Tentador Zero Beef
Costela Artesanal
Cheeseburger
Chicken Crispy
Parmesao Artesanal
Tentador Zero Frango
Big Cheddar Bacon

O Sanduiche Frango foi removido do repositório

SANDUICHES NO REPOSITÓRIO APOS REMOÇÃO:

Double Cheeseburger
Big Bob
Crispy Bacon
Frango Tomato Artesanal
Cheddar Australiano
Big Bob Frango
Tentador Frango
Tentador Carne
Tentador Zero Beef
Costela Artesanal
Cheeseburger
Chicken Crispy
Parmesao Artesanal
Tentador Zero Frango
Big Cheddar Bacon


```
***** SAIDA QUESTAO 2 *****
```

```
Sandwiches do Dia 1(por pedido):
```

```
P1 = Frango, Bacon, Cheddar Bacon,  
P2 = Creme Cheese, Caipira, Vegano,  
P3 = Big Bob, Crispy, Big Beef,
```

```
Sandwiches do Dia 2(por pedido):
```

```
P1 = Frango Acebolado, Bacon Artesanal, Big Frango,  
P2 = Bobs Salada, Cheddar Australiano, Costela Artesanal,  
P3 = Frango Caipira, Barbecue, Parmesao Artesanal,
```

```
Sandwiches do Dia 3(por pedido): (CONCATENADOS DIA 1 E 2)
```

```
P1 = Frango, Frango Acebolado, Bacon, Bacon Artesanal, Cheddar Bacon, Big Frango,  
P2 = Creme Cheese, Bobs Salada, Caipira, Cheddar Australiano, Vegano, Costela Artesanal,  
P3 = Big Bob, Frango Caipira, Crispy, Barbecue, Big Beef, Parmesao Artesanal,
```

```
Process finished with exit code 0
```

Bookmarks Structure

Cit Run TODO Problems Build Terminal

Build completed successfully in 1 sec, 721 ms (7 minutes ago)