Um trabalhador gasta anualmente com os seguintes itens: habitação, conta de luz, conta de agua, e internet

Cada mês ele gata valores diferentes dessas contas. Por exemplo, em janeiro ele gastou R$ 100,00 de água, já em fevereiro ele gastou R$110,00 de água. Como seria um sistema para que o trabalhador pudesse controlar seus gastos anuais?

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | janeiro | fevereiro | Março | abril | maio | junho | julho | agosto | setembro | outubro | novembro | dezembro |
| Luz |  |  |  |  |  |  |  |  |  |  |  |  |
| agua |  |  |  |  |  |  |  |  |  |  |  |  |
| internet |  |  |  |  |  |  |  |  |  |  |  |  |
| aluguel |  |  |  |  |  |  |  |  |  |  |  |  |

analisar o código linha a linha para melhor compreensão. Aqui está o código novamente, com explicações detalhadas:

**Código Analisado**

import java.util.Scanner;

* **Importação da Classe Scanner**: Importa a classe Scanner, que é utilizada para ler a entrada do usuário pela linha de comando.

public class ControleGastosMatriz {

* **Declaração da Classe**: Define uma classe pública chamada ControleGastosMatriz, que contém todo o código do programa.

private static final String[] MESES = {

"Janeiro", "Fevereiro", "Março", "Abril", "Maio", "Junho",

"Julho", "Agosto", "Setembro", "Outubro", "Novembro", "Dezembro"

};

* **Constante de Meses**: Um array estático final que contém os nomes dos meses do ano. O uso de final significa que este array não pode ser alterado após a inicialização.

private static final String[] TIPOS\_GASTO = { "Luz", "Água", "Internet", "Aluguel" };

* **Constante de Tipos de Gastos**: Um array também estático final que contém os tipos de gastos que serão controlados (luz, água, internet e aluguel).

private static double[][] gastos = new double[12][4];

* **Matriz de Gastos**: Declara uma matriz chamada gastos que terá 12 linhas (uma para cada mês) e 4 colunas (uma para cada tipo de gasto). Cada célula da matriz armazenará os valores gastos.

public static void main(String[] args) {

* **Método Principal**: O ponto de entrada do programa. main é um método estático que é chamado quando o programa é iniciado.

Scanner scanner = new Scanner(System.in);

* **Instanciando Scanner**: Cria um novo objeto Scanner que será usado para ler a entrada do usuário.

int opcao;

* **Declaração da Variável opcao**: Declara uma variável inteira chamada opcao, que será usada para armazenar a escolha do usuário no menu.

do {

* **Início do Loop**: Inicia um loop do-while, que continuará executando até que a condição de saída (opção 3) seja atendida.

System.out.println("Controle de Gastos Anuais");

System.out.println("1. Inserir gastos");

System.out.println("2. Exibir gastos");

System.out.println("3. Sair");

System.out.print("Escolha uma opção: ");

* **Exibição do Menu**: Imprime no console as opções disponíveis para o usuário.

opcao = scanner.nextInt();

* **Leitura da Opção**: Lê a opção escolhida pelo usuário e a armazena na variável opcao.

switch (opcao) {

* **Início do Switch**: Inicia uma estrutura de controle switch para determinar a ação a ser tomada com base na opção escolhida.

case 1:

inserirGastos(scanner);

break;

* **Opção 1**: Se o usuário escolher 1, chama o método inserirGastos(scanner), que permitirá inserir os gastos. break finaliza o case.

case 2:

exibirGastos();

break;

* **Opção 2**: Se o usuário escolher 2, chama o método exibirGastos() para mostrar os gastos já registrados. Usa break para sair do case.

case 3:

System.out.println("Saindo...");

break;

* **Opção 3**: Se o usuário escolher 3, imprime uma mensagem de saída e utiliza break para finalizar o case.

default:

System.out.println("Opção inválida. Tente novamente.");

* **Caso Default**: Se a opção não for 1, 2 ou 3, uma mensagem de erro é exibida para o usuário.

} while (opcao != 3);

* **Fim do Loop**: O loop continua até que opcao seja igual a 3 (saída).

scanner.close();

* **Fechamento do Scanner**: Fecha o objeto Scanner para liberar os recursos associados.

private static void inserirGastos(Scanner scanner) {

* **Método inserirGastos**: Declara um método privado estático que recebe um objeto Scanner como parâmetro para inserir os gastos.

for (int mes = 0; mes < MESES.length; mes++) {

* **Loop para Meses**: Inicia um loop que itera por cada mês.

System.out.println("Mês: " + MESES[mes]);

* **Imprime o Mês**: Mostra o nome do mês atual na iteração.

for (int tipo = 0; tipo < TIPOS\_GASTO.length; tipo++) {

* **Loop para Tipos de Gastos**: Inicia um loop que itera sobre os tipos de gastos.

System.out.print("Gastos com " + TIPOS\_GASTO[tipo] + ": R$ ");

* **Solicita Gastos**: Pede ao usuário que insira o valor gasto referente ao tipo de gasto atual.

gastos[mes][tipo] = scanner.nextDouble();

* **Armazena o Gasto**: Lê o valor inserido pelo usuário e o armazena na matriz gastos na posição correspondente ao mês e tipo de gasto.

}

System.out.println();

* **Nova Linha**: Após inserir todos os gastos de um mês, imprime uma linha em branco para melhor formatação.

private static void exibirGastos() {

* **Método exibirGastos**: Declara um método privado estático que exibe os gastos registrados.

System.out.println("\nGastos Anuais:");

System.out.printf("%-10s %-10s %-10s %-10s %-10s%n", "Mês", TIPOS\_GASTO[0], TIPOS\_GASTO[1], TIPOS\_GASTO[2], TIPOS\_GASTO[3]);

* **Cabeçalho da Tabela**: Imprime o título "Gastos Anuais" e formata a linha de cabeçalho para exibir os meses e os tipos de gasto.

for (int mes = 0; mes < MESES.length; mes++) {

* **Loop para Exibir Meses**: Inicia um loop que itera por cada mês.

System.out.printf("%-10s ", MESES[mes]);

* **Imprime o Mês**: Formata e imprime o nome do mês atual na tabela.

for (int tipo = 0; tipo < TIPOS\_GASTO.length; tipo++) {

* **Loop para Exibir Tipos de Gastos**: Inicia um loop para percorrer os tipos de gastos.

System.out.printf("%-10.2f ", gastos[mes][tipo]);

* **Imprime os Gastos**: Formata e imprime o valor do gasto correspondente ao mês e tipo, utilizando duas casas decimais.

}

System.out.println();

* **Nova Linha**: Após imprimir os dados de cada mês, imprime uma linha em branco para separar os meses.

System.out.println();

* **Nova Linha**: Imprime uma linha em branco após a exibição de todos os dados para melhor formatação.

**Resumo**

Esse programa oferece um sistema simples de controle de gastos anuais, permitindo que um usuário insira e visualize despesas mensais em quatro categorias. Utiliza estruturas de controle fundamentais como arrays, loops e métodos separados para organização e modularização do código. Se você tiver mais perguntas ou precisar de esclarecimentos sobre alguma parte específica, fique à vontade para perguntar!