public class AjusteSalario {  
public static void main(String[] args) {  
// Vetor de salários brutos  
double[] SalarioBruto = {1245.45, 1455.11, 1850.00};

// Vetor para armazenar os salários líquidos

double[] SalarioLiquido = new double[SalarioBruto.length];

// Calcula os salários líquidos com desconto de 11%

for (int i = 0; i < SalarioBruto.length; i++) {

SalarioLiquido[i] = SalarioBruto[i] \* (1 - 0.11);

}

// Exibe os salários líquidos

System.out.println("Salários Líquidos:");

for (double salario : SalarioLiquido) {

System.out.printf("%.2f%n", salario);

}

}

}

Código Comentado

Claro! Vamos analisar o código linha por linha, explicando cada parte com comentários detalhados. Aqui está o código com explicações incluídas:

// Declaração da classe pública chamada AjusteSalario

public class AjusteSalario {

// Método principal, ponto de entrada do programa

public static void main(String[] args) {

// Vetor de salários brutos, inicializado com três valores

double[] SalarioBruto = {1245.45, 1455.11, 1850.00};

// Vetor para armazenar os salários líquidos; seu tamanho é igual ao do vetor SalarioBruto

double[] SalarioLiquido = new double[SalarioBruto.length];

// Loop para percorrer todos os elementos do vetor SalarioBruto

for (int i = 0; i < SalarioBruto.length; i++) {

// Calcula o salário líquido aplicando o desconto de 11%

// e armazena o resultado no vetor SalarioLiquido na mesma posição

SalarioLiquido[i] = SalarioBruto[i] \* (1 - 0.11);

}

// Exibe a mensagem "Salários Líquidos:" no console

System.out.println("Salários Líquidos:");

// Loop para percorrer todos os salários líquidos no vetor SalarioLiquido

for (double salario : SalarioLiquido) {

// Formata a saída para duas casas decimais e imprime cada salário líquido

System.out.printf("%.2f%n", salario);

}

}

}

**Explicação Detalhada:**

1. **public class AjusteSalario {**
   * Define uma nova classe chamada AjusteSalario. Em Java, tudo deve estar dentro de uma classe.
2. **public static void main(String[] args) {**
   * Este é o método main, o ponto de entrada do programa onde a execução começa. public torna o método acessível de fora da classe, e static significa que podemos chamá-lo sem criar uma instância da classe.
3. **double[] SalarioBruto = {1245.45, 1455.11, 1850.00};**
   * Declara um vetor (array) de números (double) chamado SalarioBruto, que é inicializado com três valores de salários brutos.
4. **double[] SalarioLiquido = new double[SalarioBruto.length];**
   * Cria outro vetor chamado SalarioLiquido, que terá o mesmo comprimento que SalarioBruto. Este vetor armazenará os salários após a aplicação do desconto.
5. **for (int i = 0; i < SalarioBruto.length; i++) {**
   * Inicia um loop for que irá iterar sobre cada índice do vetor SalarioBruto. A variável i começará em 0 e irá até o comprimento do vetor, permitindo acessar cada salário.
6. **SalarioLiquido[i] = SalarioBruto[i] \* (1 - 0.11);**
   * Para cada salário bruto na posição i, calcula o salário líquido ao aplicar um desconto de 11%. O resultado é armazenado no vetor SalarioLiquido na mesma posição.
7. **} // Final do loop for**
   * Esta linha indica o fim do bloco do loop for.
8. **System.out.println("Salários Líquidos:");**
   * Imprime a mensagem "Salários Líquidos:" no console, para indicar que a próxima saída será a lista dos salários ajustados.
9. **for (double salario : SalarioLiquido) {**
   * Este é um loop "for-each" que itera sobre cada elemento do vetor SalarioLiquido, permitindo que a variável salario receba o valor de cada salário líquido.
10. **System.out.printf("%.2f%n", salario);**
    * Imprime o salário líquido formatado com duas casas decimais. O %n adiciona uma nova linha após cada valor.
11. **} // Final do loop for-each**
    * Indica o fim do bloco do loop for-each.
12. **} // Final do método main**
    * Marca o final do método main.
13. **} // Final da classe AjusteSalario**
    * Indica o fim da classe AjusteSalario.

Se houver algo mais que você queira discutir ou mais informações que precise, fique à vontade para perguntar!