

Explicação da lógica para criar um algoritmo de uma calculadora em Java.

Calculadora:

-Exercícios necessários;

Exercício 4 - Calculadora de Média de notas;

Exercício 5 - Tabuada Numérica;

Exercício 6 - Dados de usuário.

Uso do Scanner:

```
import java.util.Scanner;
```

```
import java.util.Scanner;
```

Ao usar o comando import, estou instruindo o programa trazer uma nova biblioteca java. Nesse caso, estarei pedindo para trazer a classe Scanner que está localizada no pacote java.util.

```
try (Scanner scanner = new Scanner(System.in)) {
```

Nesse código, utilizei a classe try para que fosse possível executar um código que tem chances de apresentar erros.

Em Scanner scanner = new Scanner(System.in), defini que a variável Scanner terá o nome “scanner” e será uma nova variável do tipo Scanner.

```
String resposta = "s"; //

do {
    System.out.print("Digite um numero inteiro: ");

    if (!scanner.hasNextInt()) {
        System.out.println("Entrada invalida. Por favor, digite um numero inteiro.");
        scanner.nextLine(); // limpa entrada inválida
        continue;
    }
}
```

Criei uma variável chamada **resposta** do tipo **String** e defini o valor inicial como "s". Essa variável controla se o programa continua executando.

O código começa um laço **do-while**, que executa o bloco pelo menos uma vez e repete enquanto a condição no final for verdadeira (true).

Dentro do laço, o programa pede para o usuário digitar um número inteiro.

Em seguida, verifica se a entrada não é um número inteiro usando `scanner.hasNextInt()`. Se for inválida, exibe uma mensagem avisando, limpa a entrada errada com `scanner.nextLine()` e volta para o início do laço com `continue` como se fosse um loop.

Se a entrada for válida, lê o número digitado com `scanner.nextInt()` e armazena na variável **numero**. Depois, usa `scanner.nextLine()` para limpar o ENTER deixado na entrada.

```
int numero = scanner.nextInt();
scanner.nextLine(); // limpa o ENTER após nextInt

if (numero <= 1) {
    System.out.println(numero + " nao e primo.");
} else {
    boolean primo = true;
    for (int i = 2; i <= Math.sqrt(numero); i++) {
        if (numero % i == 0) {
            primo = false;
            break;
        }
    }
}
```

O programa verifica se o número é menor ou igual a 1. Se for, imprime que o número não é primo. Caso contrário, inicia uma verificação para saber se o número é primo.

Então, ele define uma variável booleana **primo** como verdadeira e usa um laço (while) **for** para testar se o número é divisível por algum valor entre 2 e a raiz quadrada do número. Se encontrar um divisor, muda **primo** para falso e interrompe o laço.

Após o teste, o Netbeans exibe se o número é primo ou não, conforme o valor da variável **primo**.

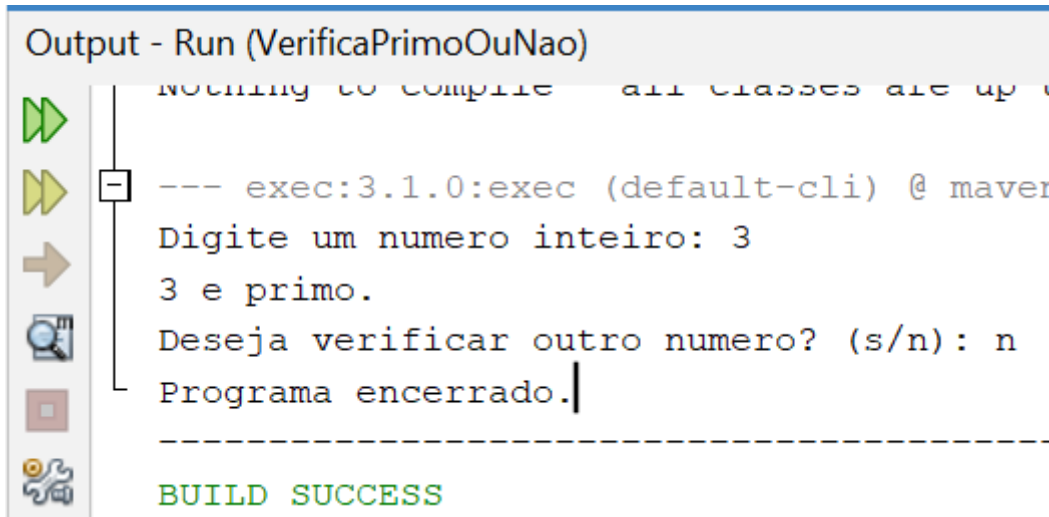
Depois, pergunta ao usuário se deseja verificar outro número (s para sim e n para não; s/n) e armazena a resposta na variável resposta.

O laço se repete enquanto a resposta for "s" (ignorando maiúsculas e minúsculas).

Quando o usuário digita qualquer coisa diferente de "s", o programa imprime "Programa encerrado." e termina.

Fiz a tarefa pelo Netbeans, mas também pode ser feita no VSCode ou outros editores de código.

Na prática:



```
Output - Run (VerificaPrimoOuNao)
--- exec:3.1.0:exec (default-cli) @ maver
Digite um numero inteiro: 3
3 e primo.
Deseja verificar outro numero? (s/n): n
Programa encerrado.
-----
BUILD SUCCESS
```

Tabuada Numérica:

```
import java.util.Scanner;

public class TabuadaNumerica {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Descubra a tabuada de um numero");

        System.out.println("Digite um numero");
        double numero = scanner.nextDouble();

        System.out.println("Tabuada de" + numero + ":");

        for (double i = 1; i <=10; i++) {
            double resultado = numero * i;
            System.out.println(numero + " x " + i + " = " + resultado);
        }
        scanner.close();
    }
}
```

Seguindo a mesma estrutura do import e Scanner, utilizei o laço double para que seja possível

Calcular a tabuada de números decimais.

```
for (double i = 1; i <=10; i++) {
```

Em (double i = 1; i <=10; i++) temos a seguinte sintaxe: “for” para executar esse código mais de uma vez, onde double indica que o valor inicial será igual a 1, e mantém o código executando até que seja menor ou igual a 10. Em i++, indica que essa variável i, deverá soma mais 1 em cada repetição de laço.

Ou seja, o número que for inserido deverá ser multiplicado do 1 até 10.

Na prática:

```
Output - Run (TabuadaNumerica)

>> Descubra a tabuada de um numero
>> Digite um numero
> 2
> Tabuada de 2.0:
2.0 x 1.0 = 2.0
2.0 x 2.0 = 4.0
2.0 x 3.0 = 6.0
2.0 x 4.0 = 8.0
2.0 x 5.0 = 10.0
2.0 x 6.0 = 12.0
2.0 x 7.0 = 14.0
2.0 x 8.0 = 16.0
2.0 x 9.0 = 18.0
2.0 x 10.0 = 20.0

-----
BUILD SUCCESS
```

Média das Notas:

```
try (Scanner scanner = new Scanner(System.in)) {
while (true) {

    System.out.println("Voce precisara digitar 4 notas"); //Evitei usar caracteres

    System.out.println("Primeira Nota");
    double nota1 = scanner.nextInt();

    System.out.println("Segunda Nota");
    double nota2 = scanner.nextDouble();

    System.out.println("Terceira Nota");
    double nota3 = scanner.nextDouble();

    System.out.println("Quarta Nota");
    double nota4 = scanner.nextDouble();

    double media = (nota1+nota2+nota3+nota4)/4;

    //Resultado da MÉDIA das notas:
    System.out.println("A media sera:" + media);

    // Pergunta se quer refazer a média:

        System.out.println("Deseja refazer a media? s/n");
        String resposta = scanner.next();
    }
}
```

Dentro de um laço infinito while true o programa avisa que o usuário precisara digitar 4 notas.

Pede para digitar a primeira nota, lê com scanner nextInt e armazena em nota1.

Pede para digitar a segunda, terceira e quarta nota, lendo cada uma com scanner nextDouble e armazenando em nota2, nota3 e nota4.

Calcula a média somando as quatro notas e dividindo por 4, armazenando em média.

Após inserir os valores, mostra o resultado da média para o usuário.

Então, o código irá perguntar se deseja refazer o cálculo da média lendo a resposta com scanner next.

Se a resposta for diferente de s o programa exhibe encerrando e sai do laço com break encerrando a execução, assim como os códigos anteriores.

Na prática:

```
- Voce precisara digitar 4 notas
Primeira Nota
8
Segunda Nota
9
Terceira Nota
10
Quarta Nota
7
A media sera:8.5
Deseja refazer a media? s/n
n
- Encerrando....
-----
BUILD SUCCESS
-----
```

Dados de Usuário:

```
import java.util.Scanner;

public class DadosDeUsuario {

    public static void main(String[] args) { //Altura deve ser com vírgula não com ponto, exemplo: 1,74
        Scanner scanner = new Scanner(System.in);

        System.out.println("Digite seu nome:");
        String nome = scanner.nextLine();

        System.out.println("Digite sua data de nascimento (dd/mm/aaaa):");
        String dataNascimento = scanner.nextLine();

        System.out.println("Digite seu curso:");
        String curso = scanner.nextLine();

        System.out.println("Digite sua série:");
        String serie = scanner.nextLine();

        System.out.println("Digite sua altura (em metros):");
        double altura = scanner.nextDouble();

        System.out.println("Digite seu peso (em kg):");
        double peso = scanner.nextDouble();

        System.out.println("\n--- Dados cadastrados ---");
        System.out.println("Nome: " + nome);
        System.out.println("Data de nascimento: " + dataNascimento);
    }
}
```

O programa exibe a mensagem "Digite seu nome" e lê a entrada do usuário com scanner `nextLine` armazenando na variável `nome`.

Em seguida exibe "Digite sua data de nascimento dd barra mm barra aaaa" e lê a resposta com scanner `nextLine` armazenando em `dataNascimento`.

Depois pede para o usuário digitar o curso e a série lendo ambos com scanner `nextLine` e armazenando nas variáveis `curso` e `serie`.

Pede para o usuário digitar a altura lê o valor do tipo `double` com scanner `nextDouble` e armazena em `altura`.

```
System.out.println("Digite sua altura (em metros):");
double altura = scanner.nextDouble();
```

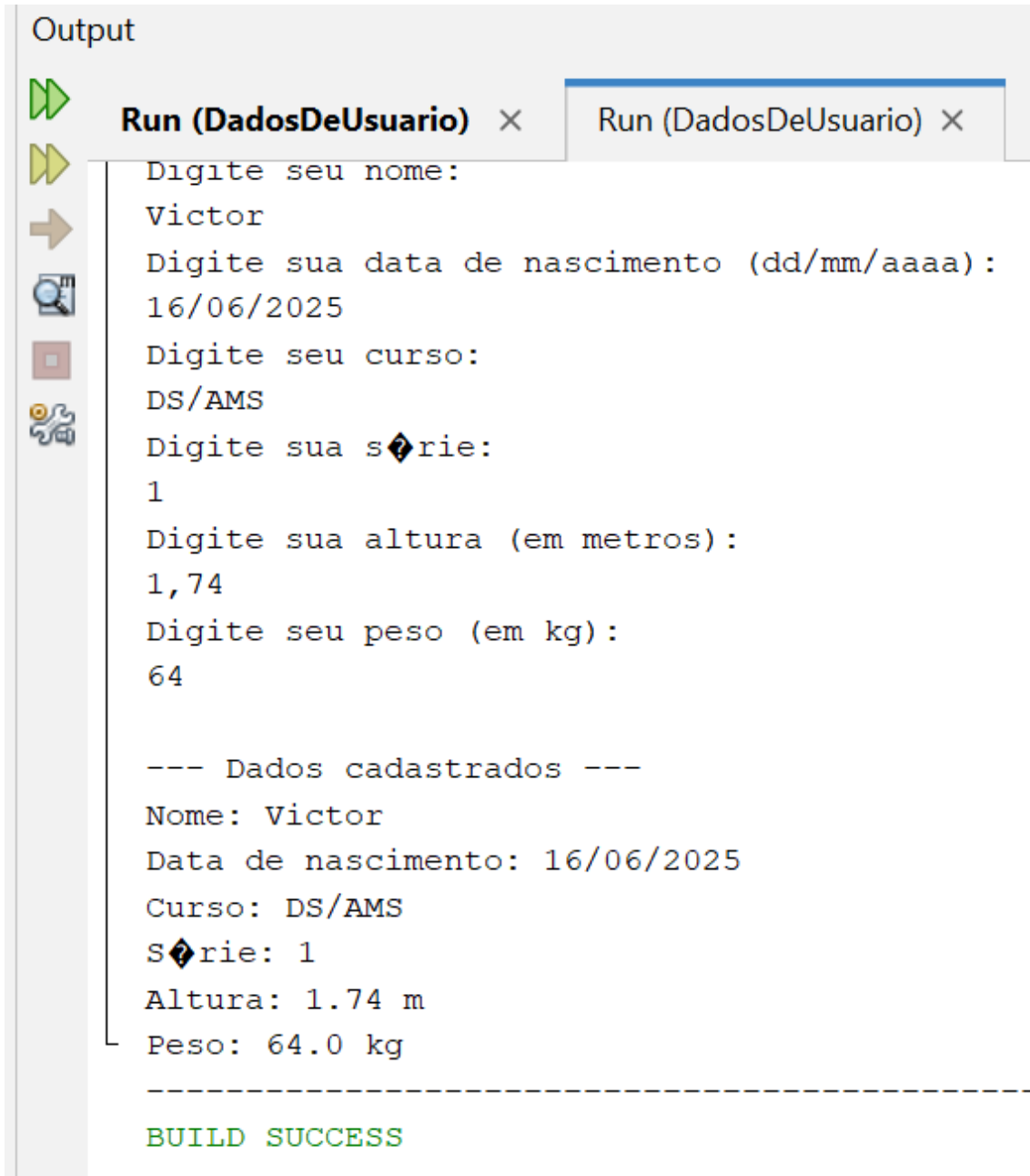
Em seguida solicita o peso lê também com scanner `nextDouble` e armazena em `peso`.

Por fim o programa imprime todas as informações cadastradas na tela para o usuário conferir.

No final fecha o objeto scanner para liberar os recursos usados na leitura dos dados.


```
scanner.close();
```

Na prática:



```
Output
Run (DadosDeUsuario) × Run (DadosDeUsuario) ×
Digite seu nome:
Victor
Digite sua data de nascimento (dd/mm/aaaa):
16/06/2025
Digite seu curso:
DS/AMS
Digite sua série:
1
Digite sua altura (em metros):
1,74
Digite seu peso (em kg):
64

--- Dados cadastrados ---
Nome: Victor
Data de nascimento: 16/06/2025
Curso: DS/AMS
Série: 1
Altura: 1.74 m
Peso: 64.0 kg
-----
BUILD SUCCESS
```

