

PROGRAMAÇÃO ORIENTADA A OBJETOS

AULA 6 – OPERADOR CONDICIONAL SWITCH



WALTER TRAVASSOS SARINHO
@PROF.WALTERTRAVASSOS
WALTER.TRAVASSOS@SECTRAS.EDU.BR

Dúvidas sobre a aula passada



i forgot

Mapa de Estudos

AULA 01

Paradigmas de Programação e história do Java.
IDEs e Hello Universe



AULA 02

Manipulação de datas.
Sintaxe da Linguagem Java e variáveis primitivas e de referência.

AULA 03

Visibilidade de variáveis, classe String e casting.



AULA 04

Classes Wrapper.
Operadores e métodos de conversão.



AULA 05

Formatar a saída de dados numérica.
Operador condicional if.
Métodos para comparar Strings



Avaliação 1

Mapa de Estudos

AULA 06

Manipulação de Strings.
Operador condicional
switch.



AULA 08

Vetores (for each) e
matrizes. Classe
ArrayList



AULA 10

Construindo
métodos e
Métodos
construtores



AULA 09

O paradigma
Orientado a
Objetos



AULA 07

Operadores de
atribuição
compostos.
Estruturas de
repetição: while,
do while e for.

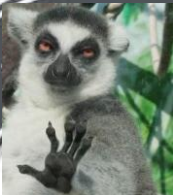


Avaliação 2

Mapa de Estudos



AULA 11
Pilares da POO:
Abstração,
Encapsulamento, Herança
e Polimorfismo



AULA 12
Classes Abstratas e
Interfaces



Avaliação 3

Exercícios de Revisão

10 minutos pra fazer agora!

1. Faça um programa que receba três notas de um aluno (receba como String e transforme para double), calcule sua média final e diga se o mesmo está aprovado (maior que 7), reprovado (menor que 5) ou vai fazer prova final (entre 5 e 7).
2. Faça um programa que leia um número e informe se o mesmo é maior (>) ou menor e igual (<=) a zero.
3. Faça um programa que receba dois números e escreva o maior deles. Caso sejam iguais, informe que são iguais.

1

```
Problems @ Javadoc Declaration  
<terminated> Principal [Java Application]  
Digite a 1ª nota:  
4  
Digite a 2ª nota:  
4  
Digite a 3ª nota:  
4  
Reprovado. Média final = 4.0
```

2

```
Problems @ Javadoc Declaration  
<terminated> Principal [Java Application] C:\Program  
Digite o 1º número:  
0  
O número é menor ou igual a zero: 0.0
```

3

```
Problems @ Javadoc  
<terminated> Principal [Java Application]  
Digite o 1º número:  
5  
Digite o 2º número:  
5  
Os números são iguais
```

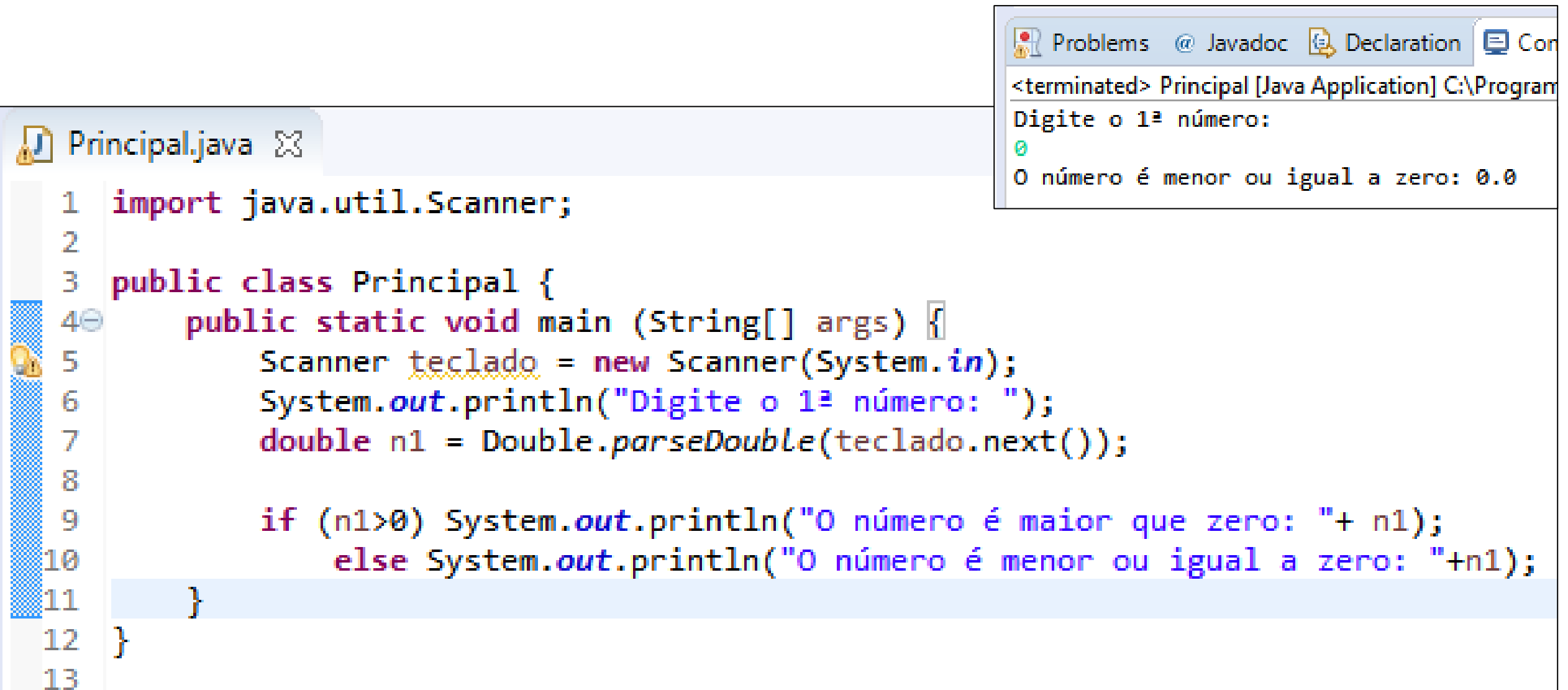
Resposta - Exercício 1

```
Principal.java ✕
1 import java.util.Scanner;
2
3 public class Principal {
4     public static void main (String[] args) {
5         Scanner teclado = new Scanner(System.in);
6         System.out.println("Digite a 1ª nota: ");
7         double n1 = Double.parseDouble(teclado.next());
8         System.out.println("Digite a 2ª nota: ");
9         double n2 = Double.parseDouble(teclado.next());
10        System.out.println("Digite a 3ª nota: ");
11        double n3 = Double.parseDouble(teclado.next());
12        double m = (n1+n2+n3)/3;
13
14        if (m >= 7) System.out.println("Aprovado. Média = "+m);
15        else if (m >= 5) System.out.println("Prova final. Média parcial = "+m);
16        else System.out.println("Reprovado. Média final = "+m);
17    }
18 }
19
```

Problems @ Javadoc Declaration

<terminated> Principal [Java Application] C:\P
Digite a 1ª nota:
4
Digite a 2ª nota:
4
Digite a 3ª nota:
4
Reprovado. Média final = 4.0

Resposta - Exercício 2



The image shows a screenshot of an IDE with a Java file named 'Principal.java' open. The code is a simple Java application that prompts the user to enter a number and then checks if it is greater than zero or less than or equal to zero. The output window on the right shows the program's execution, displaying the prompt 'Digite o 1º número:' and the result 'O número é menor ou igual a zero: 0.0'.

```
1 import java.util.Scanner;
2
3 public class Principal {
4     public static void main (String[] args) {
5         Scanner teclado = new Scanner(System.in);
6         System.out.println("Digite o 1º número: ");
7         double n1 = Double.parseDouble(teclado.next());
8
9         if (n1>0) System.out.println("O número é maior que zero: "+ n1);
10        else System.out.println("O número é menor ou igual a zero: "+n1);
11    }
12 }
13
```

Problems @ Javadoc Declaration Console

<terminated> Principal [Java Application] C:\Program

Digite o 1º número:

0

O número é menor ou igual a zero: 0.0

Resposta - Exercício 3

```
Principal.java ✕
1 import java.util.Scanner;
2
3 public class Principal {
4     public static void main (String[] args) {
5         Scanner teclado = new Scanner(System.in);
6         System.out.println("Digite o 1º número: ");
7         double n1 = Double.parseDouble(teclado.next());
8         System.out.println("Digite o 2º número: ");
9         double n2 = Double.parseDouble(teclado.next());
10
11         if (n1>n2) System.out.println("O primeiro é maior que o segundo: "+ n1);
12         else if (n2>n1) System.out.println("O segundo é maior que o primeiro: "+n2);
13         else System.out.println("Os números são iguais");
14     }
15 }
16
```

Problems @ Javadoc

<terminated> Principal [Java Applet]

Digite o 1º número:
5
Digite o 2º número:
5
Os números são iguais

Métodos para Manipulação de Strings

substring(), indexOf(), lastIndexOf(), length()

Posição dos caracteres de uma String

String str = "HENSHIN";

0	1	2	3	4	5	6
H	E	N	S	H	I	N



Método substring(arg)

Seleciona um trecho de uma String para formar uma nova String. Este método vai capturar trechos de uma String **a partir** da posição de um ou dois caracteres.

Exemplo

0	1	2	3	4	5	6
H	E	N	S	H	I	N

String **str** = "henshin";

String str2 = **str.substring**(2);

System.out.println("o conteúdo da nova String é: " + str2);



Método substring(arg 1, arg2)

ATENÇÃO QUE O PRIMEIRO ARGUMENTO É EXATAMENTE O CARACTERE DOIS. O SEGUNDO ARGUMENTO É ATÉ ONDE ELE VAI, SEM SELECIONAR NÚMERO 5.

Ao utilizar dois argumentos no método substring(), a nova String será formada pelo intervalo dos índices especificados.

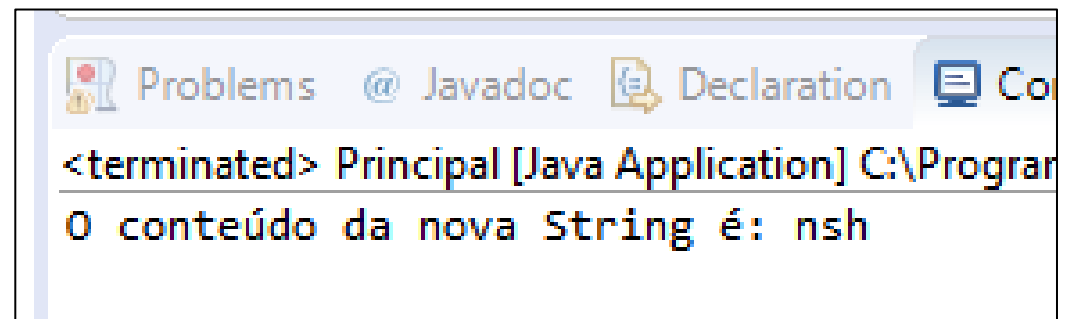
Exemplo:

0	1	2	3	4	5	6
H	E	N	S	H	I	N

String **str** = "henshin";

String str2 = **str.substring**(2, 5);

System.out.println("o conteúdo da nova String é: " + str2);



```
Problems  @ Javadoc  Declaration  Co
<terminated> Principal [Java Application] C:\Progran
O conteúdo da nova String é: nsh
```

Quais os valores de str, str2, str3 e str4?

1. String **sobrenome** = "Albuquerque";
2. String **str** = **sobrenome**.substring(2);
3. String str2 = **str**.substring (2, 5);
4. String **nome** = "Walter";
5. String str3 = **nome**.substring(4);
6. String str4 = **nome**.substring(0, 1);

Quais os valores de str, str2, str3 e str4?

1. String **sobrenome** = "Albuquerque";
2. String **str** = **sobrenome**.substring(2); //buquerque
3. String str2 = **str**.substring (2, 5); //que
4. String **nome** = "Walter";
5. String str3 = **nome**.substring(4); //er
6. String str4 = **nome**.substring(0, 1); //W

Pergunta!

Resposta: B

Qual resposta irá aparecer no console a partir do trecho de código a seguir:

```
String str = "Pizza";  
System.out.println(str.substring(0));
```

- a) lzza
- b) Pizza
- c) Zza
- d) Za
- e) P

Método indexOf e lastIndexOf

Capturam a primeira e última ocorrência de um dado caractere na String.

Exemplo:

```
String nome = "Walter Travassos Sarinho";
```

```
int x = nome.indexOf('T');
```

```
int y = nome.lastIndexOf('a');
```

```
System.out.println(x);    //7
```

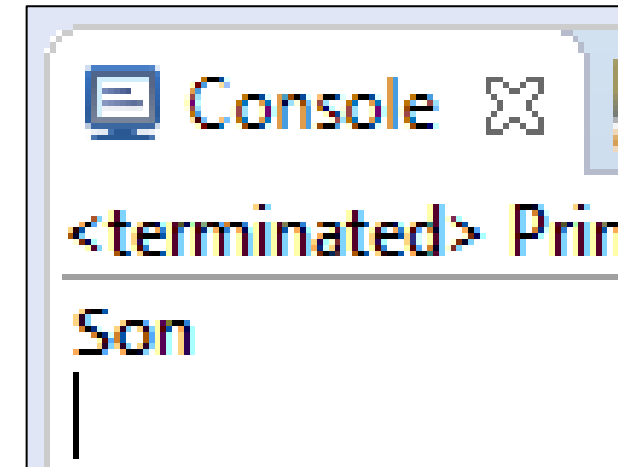
```
System.out.println(y);    //18
```

Qual o resultado apresentado no console?



Podemos usar os métodos em conjunto para criar uma nova String a partir de outra:

```
String str = "Son Goku";  
String str2 = str.substring(0, str.indexOf(" "));  
System.out.println(str2);
```



Método length()

Captura o tamanho da String.

Exemplo:

```
String nome = "walter";
```

```
int i = nome.length(); // i = 6
```

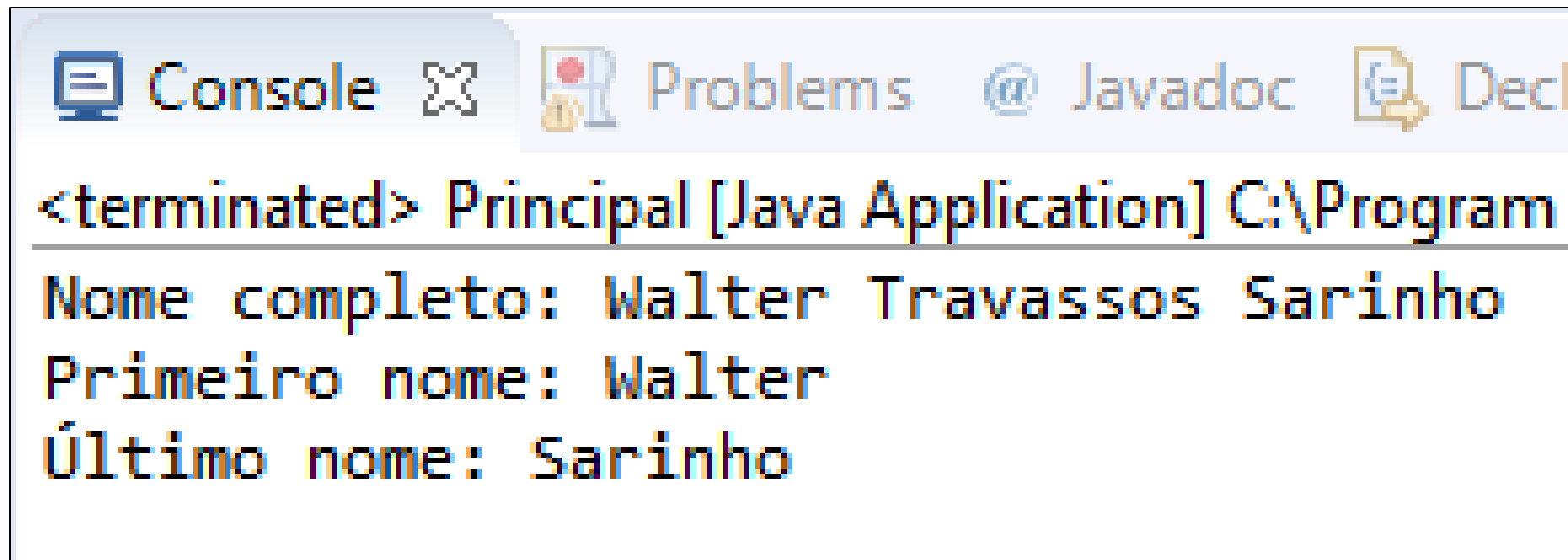
1	2	3	4	5	6
w	a	l	t	e	r

Atente que, apesar de acessarmos cada caractere de uma String a partir do número zero, o tamanho é contado a partir de 1.

Exercício 4

5 minutos

Faça um programa que receba seu nome completo na variável nome, em seguida extraia dessa String seu primeiro e seu último nome. O resultado final deve ser:



```
<terminated> Principal [Java Application] C:\Program  
Nome completo: Walter Travassos Sarinho  
Primeiro nome: Walter  
Último nome: Sarinho
```

Exercício 4 - Resposta

```
*Principal.java ✕  
1 public class Principal {  
2     public static void main (String[] args) {  
3         String primeiro, último, nome;  
4         nome = "Walter Travassos Sarinho";  
5         primeiro = nome.substring(0,nome.indexOf(" "));  
6         último = nome.substring(nome.lastIndexOf(" ")+1);  
7         System.out.println("Nome completo: "+ nome);  
8         System.out.println("Primeiro nome: "+ primeiro);  
9         System.out.println("último nome: "+último);  
10  
11     }  
12 }
```

Console ✕ Problems @ Javadoc Decl

<terminated> Principal [Java Application] C:\Program
Nome completo: Walter Travassos Sarinho
Primeiro nome: Walter
Último nome: Sarinho

Exercício Desafio

Faça um programa que receba seu nome completo na variável **nome** e gere um **login** e **senha** de acordo com os requisitos abaixo. Ao final apresente o login e senha gerado.

- **Login:** primeira letra do primeiro nome + último nome + @ estudante.edu.br
- **Senha:** primeira letra de cada um dos nomes + ano atual (use a classe `LocalDate`) + 4 números aleatórios utilizando o método `Math.random()`.

Por exemplo

- Nome: Walter Travassos Sarinho
- Login: wsarinho@estudante.edu.br
- Senha: wts20236891

Dica para pegar as iniciais de cada letra do nome

```
String nome = "Walter Travassos Sarinho";  
String iniciais = nome.substring(0,1);  
for (int i=0; i<nome.length(); i++) {  
    if( nome.substring(i,i+1).equals(" ") ) {  
        iniciais = iniciais + nome.substring(i+1, i+2);  
    }  
}  
System.out.println(iniciais);
```

Intervalo: 15 minutos

Estrutura Condicional switch

O que já sabemos...

Utilizar a estrutura de decisão `if` e aninhar “blocos ifs” quando o processo de decisão exige mais que duas opções (verdadeiro e falso).

Agora iremos aprender a utilizar o “switch case” em Java, que é outra estrutura para manipular o controle do fluxo do seu programa.

Java – Estruturas Condicionais

switch

- Trata-se de uma estrutura condicional de **seleção múltipla**.
- Funcionamento similar a estrutura **escolha...caso** em algoritmos.
- Trata **várias condições ao mesmo tempo**.
- Trata condições genéricas (equivalentes ao **outrocaso** em algoritmos).

Sintaxe

```
switch (variável) {  
    case valor_1:  
        // Bloco de comandos 1  
        break;  
    case valor_2:  
        // Bloco de comandos 2  
        break;  
    case valor_3:  
        // Bloco de comandos 3  
        break;  
    default:  
        // Lista de comandos caso a opção escolhida seja “inválida”.  
}  

```

switch

switch (**var**)

Onde, *var* é uma variável do tipo **int**, **String** ou **char**.

```
switch (var) {  
    case valor1: comando1; break;  
    case valor2: comando2; break;  
    ...  
    default: comandoN; break;  
}
```

```
int dia = teclado.nextInt();  
switch (dia) {  
    case 1: nome = "segunda"; break;  
    case 2: nome = "terça"; break;  
    ...  
    case 7: nome = "domingo"; break;  
    default: nome = "inexistente";  
}
```

**switch
com variável do tipo char**

switch – capturando o caractere digitado pelo usuário

Você pode capturar a decisão do usuário utilizando os métodos da classe Scanner (nextInt(), por exemplo) para em seguida usá-la no switch;

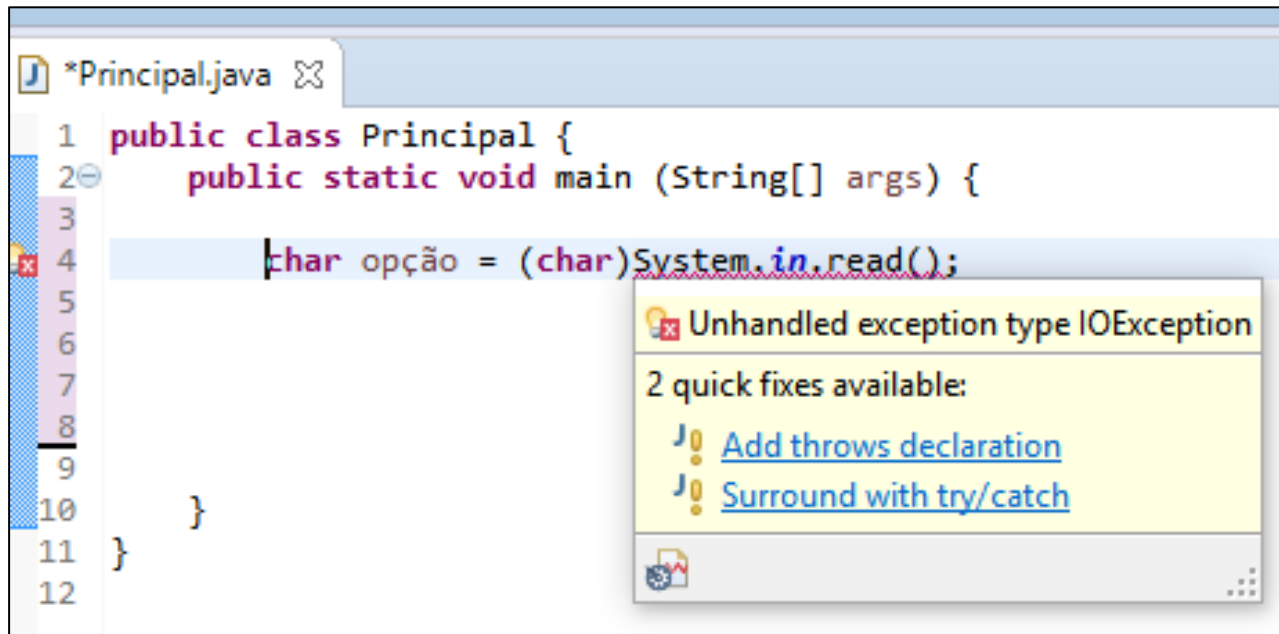
Contudo, para **capturar uma variável do tipo char**, é necessário utilizar o seguinte método:

```
char opção = (char)System.in.read();
```

char – cercando com try catch

Observe que será necessário cercar o comando com um **try.. catch**.
Opção *Surround with try/catch*.

Veja a opção de correção que o eclipse sugere

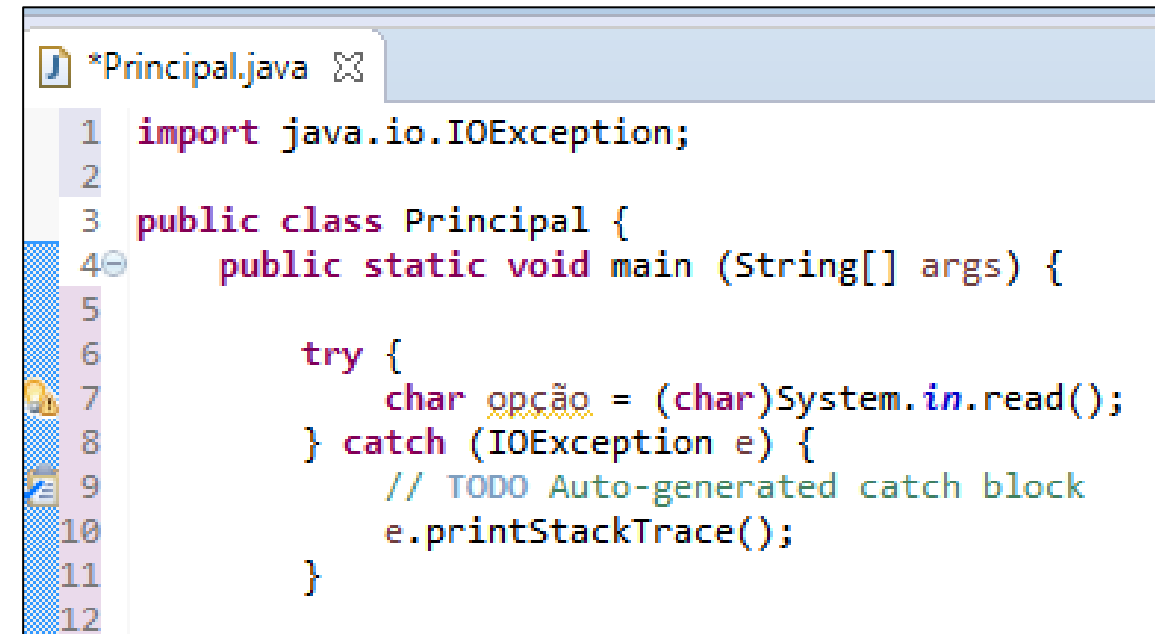


The screenshot shows the Eclipse IDE with a file named `*Principal.java`. The code is as follows:

```
1 public class Principal {
2     public static void main (String[] args) {
3
4         char opção = (char)System.in.read();
5
6
7
8
9
10    }
11 }
12
```

A red squiggly line under the `System.in.read()` call on line 4 indicates an error. A tooltip is displayed over this line, showing the message "Unhandled exception type IOException" and two quick fixes:

- [Add throws declaration](#)
- [Surround with try/catch](#)



The screenshot shows the Eclipse IDE with the same file `*Principal.java` after applying the "Surround with try/catch" quick fix. The code is now wrapped in a try-catch block:

```
1 import java.io.IOException;
2
3 public class Principal {
4     public static void main (String[] args) {
5
6         try {
7             char opção = (char)System.in.read();
8         } catch (IOException e) {
9             // TODO Auto-generated catch block
10            e.printStackTrace();
11        }
12    }
13 }
```

switch – exemplo com variável char

```
try {  
    char opção = (char)System.in.read();  
} catch (IOException e) {  
    e.printStackTrace();  
}  
  
switch(opção) {  
    case 1: System.out.println("1"); break;  
    case 2: System.out.println("2"); break;  
    default: System.out.println("outro número");  
}
```


switch
com variável do tipo int

switch – exemplo com variável int

```
Scanner teclado = new Scanner(System.in);  
int opção = teclado.nextInt();
```

```
switch(opção) {  
    case 1: System.out.println("1"); break;  
    case 2: System.out.println("2"); break;  
    default: System.out.println("outro número");  
}
```

**switch
com variável do tipo String**

switch – exemplo com variável String

```
Scanner teclado = new Scanner(System.in);
```

```
String opção = teclado.next();
```

```
switch(opção) {
```

```
    case "1": System.out.println("1"); break;
```

```
    case "2": System.out.println("2"); break;
```

```
    default: System.out.println("outro número");
```

```
}
```

Mais Exercícios

Exercício 5

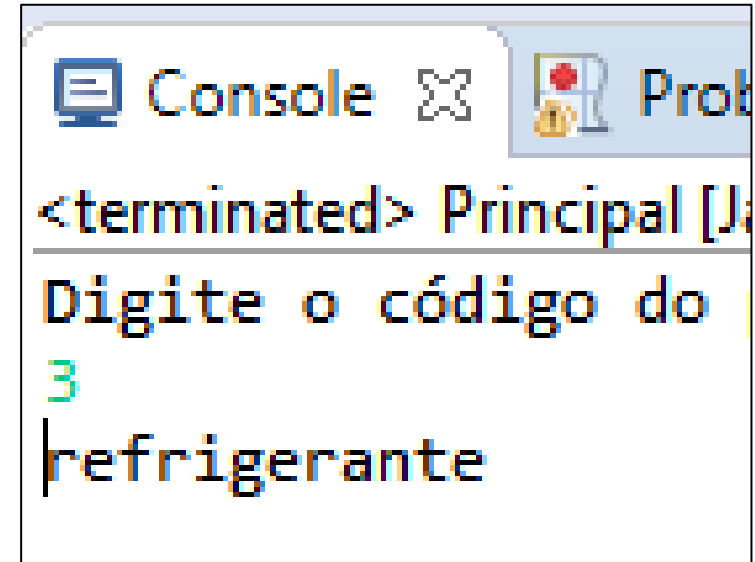
Faça um programa que escreve o nome de um produto através do código digitado pelo usuário.

Se o usuário digitar 1, o programa deverá escrever 'Pizza'.

Se o usuário digitar 2, o programa deverá escrever 'Hamburger'.

Se o usuário digitar 3, o programa deverá escrever 'Refrigerante'.

Se o usuário digitar 4, o programa deverá escrever 'Batata Frita'.



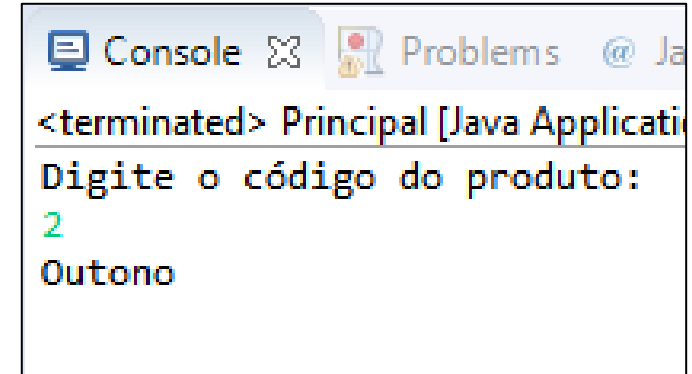
```
<terminated> Principal [J
Digite o código do
3
refrigerante
```

Resposta – Exercício 5

```
Principal.java ✕
1 import java.util.Scanner;
2
3 public class Principal {
4     public static void main (String[] args) {
5
6         Scanner entrada = new Scanner(System.in);
7         int codigo;
8         System.out.println("Digite o código do produto:");
9         codigo = entrada.nextInt();
10        switch(codigo) {
11            case 1:      System.out.println("pizza");
12                        break;
13            case 2:      System.out.println("hamburger");
14                        break;
15            case 3:      System.out.println("refrigerante");
16                        break;
17            case 4:      System.out.println("batata frita");
18                        break;
19            default:     System.out.println("Produto não cadastrado");
20        }
21    }
22 }
23
```

```
Console ✕ Problem
<terminated> Principal [Ja
Digite o código do
3
refrigerante
```

Exercício 6



The screenshot shows a Java IDE window with a console pane. The console output is as follows:

```
<terminated> Principal [Java Applicati  
Digite o código do produto:  
2  
Outono
```

Faça um programa que escreve o nome das estações do ano de acordo com o valor numérico digitado pelo usuário.

Caso o usuário:

Digite 1, o programa deverá escrever 'Verão';

Digite 2, o programa deverá escrever 'Outono';

Digite 3, o programa deverá escrever 'Inverno';

Digite 4, o programa deverá escrever 'Primavera';

Resposta – Exercício 6

```
*Principal.java ✕  
1 import java.util.Scanner;  
2  
3 public class Principal {  
4     public static void main (String[] args) {  
5         Scanner entrada = new Scanner(System.in);  
6         int codigo;  
7         System.out.println("Digite o código do produto:");  
8         codigo = entrada.nextInt();  
9         switch(codigo) {  
10            case 1:      System.out.println("Verão");  
11                        break;  
12            case 2:      System.out.println("Outono");  
13                        break;  
14            case 3:      System.out.println("Inverno");  
15                        break;  
16            case 4:      System.out.println("Primavera");  
17                        break;  
18            default:     System.out.println("Opção inválida");  
19        }  
20    }  
21 }  
22
```

```
Console ✕ Problems @ Ja  
<terminated> Principal [Java Applicati  
Digite o código do produto:  
2  
Outono
```

Exercício 7

5 minutos

Faça um programa que receba um **código** e a **quantidade** de um **produto** no momento de sua compra.

O programa deverá escrever o nome do produto e calcular o valor total da compra.

Segue a tabela de código, produto e preço:

Código	Nome	Preço Unitário
0	Cachorro Quente	1.50
1	Refrigerante	2.00

```
Console Problems @ Javadoc
<terminated> Principal [Java Application] C:\Progra
Digite o código do produto e a quant.:
0
4
Cachorro Quente
Total: R$ 6.0
```

Resposta – Exercício 7

```
*Principal.java
1 import java.util.Scanner;
2
3 public class Principal {
4     public static void main (String[] args) {
5
6         Scanner entrada = new Scanner(System.in);
7         int codigo, quant;
8         System.out.println("Digite o código do produto e a quant.:");
9         codigo = entrada.nextInt();
10        quant = entrada.nextInt();
11        switch(codigo) {
12            case 0:      System.out.println("Cachorro Quente");
13                        System.out.println("Total: R$ " + (1.50 * quant));
14                        break;
15            case 1:      System.out.println("Refrigerante");
16                        System.out.println("Total: R$ " + (2.00 * quant));
17                        break;
18            default:     System.out.println("Produto Inexistente");
19        }
20    }
21 }
```

Console Problems Javadoc

<terminated> Principal [Java Application] C:\Progra

Digite o código do produto e a quant.:

0

4

Cachorro Quente

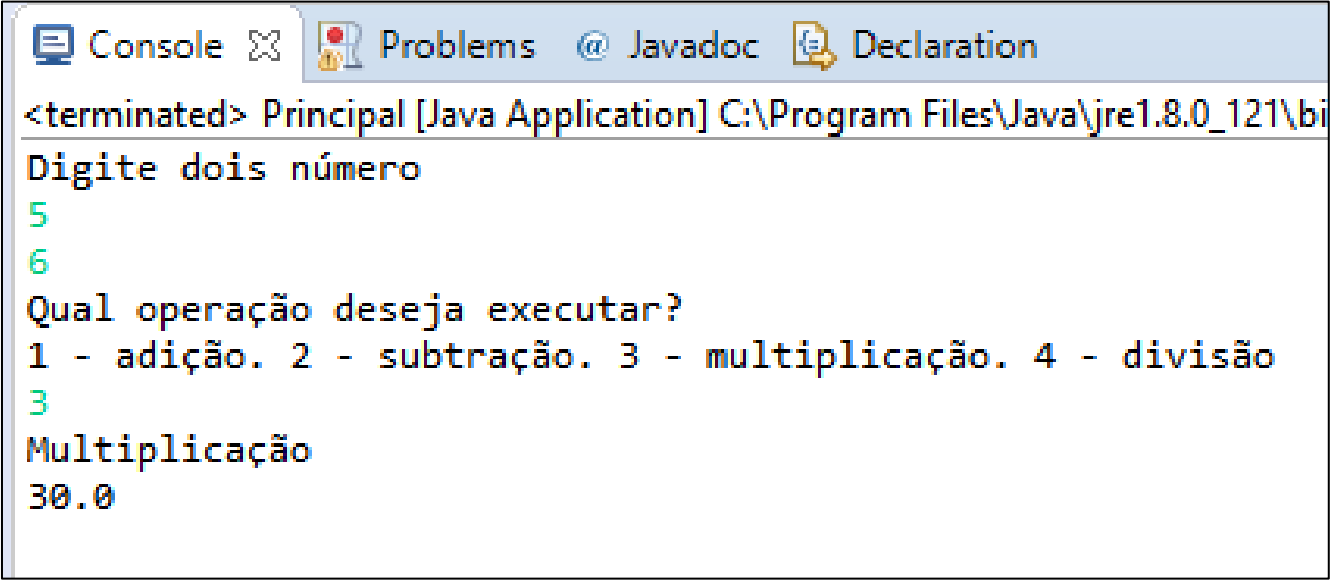
Total: R\$ 6.0

Exercício 8

Faça um programa que efetue as operações matemáticas básicas com dois números reais (double) digitados pelo usuário. O usuário deve também selecionar qual operação será executada.

Códigos de cada operação:

1. Adição;
2. Subtração;
3. Multiplicação;
4. Divisão;
5. Sair;



```
<terminated> Principal [Java Application] C:\Program Files\Java\jre1.8.0_121\bin
Digite dois número
5
6
Qual operação deseja executar?
1 - adição. 2 - subtração. 3 - multiplicação. 4 - divisão
3
Multiplicação
30.0
```

Tente criar um método **menu()** que será chamado ao final de cada operação. Assim, o programa só vai parar quando for escolhida a opção 5.

Resposta – Exercício 8

Principal.java

```
1 import java.util.Scanner;
2
3 public class Principal {
4     public static void main (String[] args) {
5         Scanner entrada = new Scanner(System.in);
6         System.out.println("Digite dois número");
7         double a = entrada.nextDouble();
8         double b = entrada.nextDouble();
9         System.out.println("Qual operação deseja executar?");
10        System.out.println("1 - adição. 2 - subtração. 3 - multiplicação. 4 - divisão");
11        int codigo = entrada.nextInt();
12        switch(codigo) {
13            case 1: System.out.println("Adição");
14                   System.out.println(a+b);
15                   break;
16            case 2: System.out.println("Subtração");
17                   System.out.println(a-b);
18                   break;
19            case 3: System.out.println("Multiplicação");
20                   System.out.println(a*b);
21                   break;
22            case 4: System.out.println("Divisão");
23                   System.out.println(a/b);
24                   break;
25            default: System.out.println("Opção inválida");
26        }
27    }
28 }
29
```

Console Problems @ Javadoc Declaration

```
<terminated> Principal [Java Application] C:\Program Files\Java\jre1.8.0_121\bin
Digite dois número
5
6
Qual operação deseja executar?
1 - adição. 2 - subtração. 3 - multiplicação. 4 - divisão
3
Multiplicação
30.0
```

Resposta com o método menu() – parte 1

```
1 import java.util.Scanner;
2
3 public class Principal {
4
5     public static void main(String[] args) {
6         menu();
7     }
8
9     public static void menu() {
10         Scanner teclado = new Scanner(System.in);
11         System.out.println("Digite o primeiro número: ");
12         double n1 = teclado.nextDouble();
13         System.out.println("Digite o segundo número: ");
14         double n2 = teclado.nextDouble();
15         System.out.println("Escolha uma operação (1 a 4)");
16         System.out.println("5 pra sair.");
17
18         String opcao = teclado.next();
```

Resposta com o método menu() – parte 2

```
20      switch (opcao) {
21      case "1":
22          System.out.println("Adição: " + (n1+n2));
23          menu();
24      case "2":
25          System.out.println("Subtração: " + (n1-n2));
26          menu();
27      case "3":
28          System.out.println("Multiplicação: " + (n1*n2));
29          menu();
30      case "4":
31          System.out.println("Divisão: " + (n1/n2));
32          menu();
33      case "5":
34          System.out.println("Fim do programa");
35          System.exit(0);
36      default:
37          System.out.println("algum erro");
38      }
39  }
40 }
```

PROGRAMAÇÃO ORIENTADA A OBJETOS

AULA 6 – OPERADOR CONDICIONAL SWITCH

WALTER TRAVASSOS SARINHO

@PROF.WALTERTRAVASSOS

WALTER.TRAVASSOS@SECTRAS.EDU.BR