

Aula 08 – Condicionais e Laços

Norton Trevisan Roman

10 de abril de 2018

Switch

- Considere o código que calcula o valor da construção:

```
static double valor(  
    double area) {  
    if (area >= 0)  
        return(valorM2*area);  
    return(-1);  
}
```

Switch

- Considere o código que calcula o valor da construção:

```
static double valor(  
    double area) {  
    if (area >= 0)  
        return(valorM2*area);  
    return(-1);  
}
```

- Suponha agora que haja 4 materiais, com preços diferentes, para a piscina

Switch

- Considere o código que calcula o valor da construção:

```
static double valor(  
    double area) {  
    if (area >= 0)  
        return(valorM2*area);  
    return(-1);  
}
```

- Suponha agora que haja 4 materiais, com preços diferentes, para a piscina
- Teremos que desmembrar o código, criando método apenas para a piscina. Como fazer?

Switch

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;

    if (material == ALVENARIA) valor = 1500;
    else
        if (material == VINIL) valor = 1100;
        else
            if (material == FIBRA) valor = 750;
            else
                if (material == PLASTICO)
                    valor = 500;
                else valor = -1;
    return(area*valor);
}
```

Switch

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;

    if (material == ALVENARIA) valor = 1500;
    else
        if (material == VINIL) valor = 1100;
        else
            if (material == FIBRA) valor = 750;
            else
                if (material == PLASTICO)
                    valor = 500;
                else valor = -1;
    return(area*valor);
}
```

- **Constantes** também podem ser usadas para definir categorias

Switch

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;

    if (material == ALVENARIA) valor = 1500;
    else
        if (material == VINIL) valor = 1100;
        else
            if (material == FIBRA) valor = 750;
            else
                if (material == PLASTICO)
                    valor = 500;
                else valor = -1;
    return(area*valor);
}
```

- **Constantes** também podem ser usadas para definir categorias
- Deixam o código bem mais legível

Switch

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;

    if (material == ALVENARIA) valor = 1500;
    else
        if (material == VINIL) valor = 1100;
        else
            if (material == FIBRA) valor = 750;
            else
                if (material == PLASTICO)
                    valor = 500;
                else valor = -1;
    return(area*valor);
}
```

- **Constantes** também podem ser usadas para definir categorias
- Deixam o código bem mais legível
- Haveria como evitar esse **aninhamento**?

Switch

```
switch (variavel) {  
    case valor1: código1  
        break;  
    case valor2: código2  
        break;  
    ...  
    case valorn: códigon  
        break;  
    default: códigopadrão;  
}
```

Correspondendo a:

```
if (variavel == valor1)  
    código1;  
else  
    if (variavel == valor2)  
        código2;  
    else  
        ...  
        else  
            if (variavel == valorn)  
                códigon;  
            else códigopadrão;
```

Switch

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;

    if (material == ALVENARIA)
        valor = 1500;
    else
        if (material == VINIL) valor = 1100;
        else
            if (material == FIBRA) valor = 750;
            else
                if (material == PLASTICO)
                    valor = 500;
                else valor = -1;
    return(area*valor);
}
```

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                        break;
        default: valor = -1;
    }
    return(area*valor);
}
```

Switch

- Switch testa a variável contra todos os valores listados

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                       break;
        default: valor = -1;
    }
    return(area*valor);
}
```

Switch

- Switch testa a variável contra todos os valores listados
- Se o valor dela não estiver listado, ativa o código em **default**

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                       break;
        default: valor = -1;
    }
    return(area*valor);
}
```

Switch

- Switch testa a variável contra todos os valores listados
- Se o valor dela não estiver listado, ativa o código em **default**
- default é opcional – se não houver, simplesmente sai do switch e continua o programa

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                        break;
        default: valor = -1;
    }
    return(area*valor);
}
```

Switch

- E o **break**?

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                        break;
        default: valor = -1;
    }
    return(area*valor);
}
```

Switch

- E o **break**?
- Faz com que o switch pare nesse ponto

```
static final int ALVENARIA = 0;
static final int VINIL = 1;
static final int FIBRA = 2;
static final int PLASTICO = 3;
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                   break;
        case FIBRA: valor = 750;
                   break;
        case PLASTICO: valor = 500;
                       break;
        default: valor = -1;
    }
    return(area*valor);
}
```

Switch

- O que acontece se tirarmos o break?

```
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
        case VINIL: valor = 1100;
        case FIBRA: valor = 750;
        case PLASTICO: valor = 500;
        default: valor = -1;
    }
    return(area*valor);
}
```

```
...
public static void main(String[] args) {
    System.out.println(
        valorPiscina(100,ALVENARIA));
}
```


Switch

- O que acontece se tirarmos o break?

```
$ java AreaCasa  
-100.0
```

```
...  
static double valorPiscina(double area,  
                           int material) {  
    double valor;  
    switch (material) {  
        case ALVENARIA: valor = 1500;  
        case VINIL: valor = 1100;  
        case FIBRA: valor = 750;  
        case PLASTICO: valor = 500;  
        default: valor = -1;  
    }  
    return(area*valor);  
}
```

```
...  
public static void main(String[] args) {  
    System.out.println(  
        valorPiscina(100,ALVENARIA));  
}
```

Switch

- O que acontece se tirarmos o break?

```
$ java AreaCasa  
-100.0
```

- Quando o switch começou, **ALVENARIA** fez valor=1500

```
...  
static double valorPiscina(double area,  
                           int material) {  
    double valor;  
    switch (material) {  
        case ALVENARIA: valor = 1500;  
        case VINIL: valor = 1100;  
        case FIBRA: valor = 750;  
        case PLASTICO: valor = 500;  
        default: valor = -1;  
    }  
    return(area*valor);  
}  
  
...  
public static void main(String[] args) {  
    System.out.println(  
        valorPiscina(100,ALVENARIA));  
}
```

Switch

- Como não houve o break, ele continuou, e fez valor=1100, 750, 500 e, finalmente, -1

```
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
        case VINIL: valor = 1100;
        case FIBRA: valor = 750;
        case PLASTICO: valor = 500;
        default: valor = -1;
    }
    return(area*valor);
}

...
public static void main(String[] args) {
    System.out.println(
        valorPiscina(100,ALVENARIA));
}
```

Switch

- Como não houve o `break`, ele continuou, e fez `valor=1100, 750, 500` e, finalmente, `-1`
- Então pegou esse `valor -1` e multiplicou por `area`, retornando

```
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
        case VINIL: valor = 1100;
        case FIBRA: valor = 750;
        case PLASTICO: valor = 500;
        default: valor = -1;
    }
    return(area*valor);
}

...
public static void main(String[] args) {
    System.out.println(
        valorPiscina(100,ALVENARIA));
}
```

Switch

- Como não houve o `break`, ele continuou, e fez `valor=1100, 750, 500` e, finalmente, `-1`
- Então pegou esse `valor -1` e multiplicou por `area`, retornando
- `case` define o ponto de entrada. Só isso.

```
...
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
        case VINIL: valor = 1100;
        case FIBRA: valor = 750;
        case PLASTICO: valor = 500;
        default: valor = -1;
    }
    return(area*valor);
}

...
public static void main(String[] args) {
    System.out.println(
        valorPiscina(100,ALVENARIA));
}
```

Switch

- E se agora removemos o default?
- Note que também removemos o último break, por ser desnecessário
- Irá compilar?

```
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                    break;
    }
    return(area*valor);
}
...
public static void main(String[] args) {
    System.out.println(
        valorPiscina(100,5));
}
```

Switch

```
$ javac AreaCasa.java
AreaCasa.java:85: variable
valor might not have been
initialized
    return(area*valor);
                ^
1 error
```

```
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                    break;
    }
    return(area*valor);
}
...
public static void main(String[] args) {
    System.out.println(
        valorPiscina(100,5));
}
```

Switch

```
$ javac AreaCasa.java
AreaCasa.java:85: variable
valor might not have been
initialized
```

```
    return(area*valor);
                ^
```

1 error

- O compilador não usa algo que pode não ter sido inicializado

```
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                    break;
    }
    return(area*valor);
}

...
public static void main(String[] args) {
    System.out.println(
        valorPiscina(100,5));
}
```


Switch

- Então inicializemos...

```
static double valorPiscina(double area,
                           int material) {
    double valor=0;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                  break;
        case FIBRA: valor = 750;
                  break;
        case PLASTICO: valor = 500;
    }
    return(area*valor);
}
...
public static void main(String[] args) {
    System.out.println(
        valorPiscina(100,5));
}
```

Switch

- Então inicializemos...
- E agora?

```
$ java AreaCasa  
0.0
```

```
static double valorPiscina(double area,  
                           int material) {  
    double valor=0;  
    switch (material) {  
        case ALVENARIA: valor = 1500;  
                        break;  
        case VINIL: valor = 1100;  
                   break;  
        case FIBRA: valor = 750;  
                   break;  
        case PLASTICO: valor = 500;  
    }  
    return(area*valor);  
}  
...  
public static void main(String[] args) {  
    System.out.println(  
        valorPiscina(100,5));  
}
```

Switch

- Então inicializemos...

- E agora?

```
$ java AreaCasa  
0.0
```

- Um valor inválido deixou de ser tratado

```
static double valorPiscina(double area,  
                           int material) {  
    double valor=0;  
    switch (material) {  
        case ALVENARIA: valor = 1500;  
                        break;  
        case VINIL: valor = 1100;  
                   break;  
        case FIBRA: valor = 750;  
                   break;  
        case PLASTICO: valor = 500;  
    }  
    return(area*valor);  
}  
...  
public static void main(String[] args) {  
    System.out.println(  
        valorPiscina(100,5));  
}
```

Switch

```
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                       break;
        default: valor = -1;
    }
    return(area*valor);
}
```

- Daria para enxugar o código, dado que há uma escolha e um retorno?

Switch

```
static double valorPiscina(double area,
                           int material) {
    double valor;
    switch (material) {
        case ALVENARIA: valor = 1500;
                        break;
        case VINIL: valor = 1100;
                    break;
        case FIBRA: valor = 750;
                    break;
        case PLASTICO: valor = 500;
                       break;
        default: valor = -1;
    }
    return(area*valor);
}
```

- Daria para enxugar o código, dado que há uma escolha e um retorno?

```
static double valorPiscina(double area,
                           int material) {
    switch (material) {
        case ALVENARIA: return(area*1500);
        case VINIL: return(area*1100);
        case FIBRA: return(area*750);
        case PLASTICO: return(area*500);
        default: return(-1);
    }
}
```

- Queremos agora comparar o valor de uma piscina de 100m^2 , para cada material
- Como faríamos?

- Queremos agora comparar o valor de uma piscina de 100m^2 , para cada material
- Como faríamos?
- Primeira tentativa:

```
public static void main(String[]  
                        args) {  
    double area = 100;  
    System.out.println("Material\t  
                        Valor");  
    System.out.println(ALVENARIA+  
                        "\t\t"+  
                        valorPiscina(area,ALVENARIA));  
    System.out.println(VINIL+"\t\t"+  
                        valorPiscina(area,VINIL));  
    System.out.println(FIBRA+"\t\t"+  
                        valorPiscina(area,FIBRA));  
    System.out.println(PLASTICO+"\t\t"+  
                        +valorPiscina(area,PLASTICO));  
}
```

- \t – **tabulação**
- E qual a saída?

Material	Valor
0	150000.0
1	110000.0
2	75000.0
3	50000.0

```
public static void main(String[]  
                                args) {  
    double area = 100;  
    System.out.println("Material\t  
                                Valor");  
    System.out.println(ALVENARIA+  
                                "\t\t"+  
                                valorPiscina(area,ALVENARIA));  
    System.out.println(VINIL+"\t\t"+  
                                valorPiscina(area,VINIL));  
    System.out.println(FIBRA+"\t\t"+  
                                valorPiscina(area,FIBRA));  
    System.out.println(PLASTICO+"\t\t"+  
                                +valorPiscina(area,PLASTICO));  
}
```


- \t – **tabulação**
- E qual a saída?

Material	Valor
0	150000.0
1	110000.0
2	75000.0
3	50000.0

- Com 4 tipos foi fácil...

```
public static void main(String[]  
                                args) {  
    double area = 100;  
    System.out.println("Material\t  
                        Valor");  
    System.out.println(ALVENARIA+  
                        "\t\t"+  
                        valorPiscina(area,ALVENARIA));  
    System.out.println(VINIL+"\t\t"+  
                        valorPiscina(area,VINIL));  
    System.out.println(FIBRA+"\t\t"+  
                        valorPiscina(area,FIBRA));  
    System.out.println(PLASTICO+"\t\t"+  
                        +valorPiscina(area,PLASTICO));  
}
```

- E se tivéssemos 20 tipos diferentes de materiais?

- E se tivéssemos 20 tipos diferentes de materiais?
- Repare nos valores dos tipos de material para a piscina:

```
static final int ALVENARIA = 0;  
static final int VINIL = 1;  
static final int FIBRA = 2;  
static final int PLASTICO = 3;
```

- E se tivéssemos 20 tipos diferentes de materiais?
- Repare nos valores dos tipos de material para a piscina:

```
static final int ALVENARIA = 0;  
static final int VINIL = 1;  
static final int FIBRA = 2;  
static final int PLASTICO = 3;
```

- São inteiros sequenciais...

- E se tivéssemos 20 tipos diferentes de materiais?
- Repare nos valores dos tipos de material para a piscina:

```
static final int ALVENARIA = 0;  
static final int VINIL = 1;  
static final int FIBRA = 2;  
static final int PLASTICO = 3;
```

- São inteiros sequenciais...
- Como podemos usar isso?

Laços

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= PLASTICO) {  
        System.out.println(tipo+  
                            "\t\t"+  
                            valorPiscina(area,tipo));  
        tipo = tipo+1;  
    }  
}
```

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = 0;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= 3) {  
        System.out.println(tipo+  
                            "\t\t"+  
                            valorPiscina(area,tipo));  
        tipo = tipo+1;  
    }  
}
```

Laços

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= PLASTICO) {  
        System.out.println(tipo+  
                            "\t\t"+  
                            valorPiscina(area,tipo));  
        tipo = tipo+1;  
    }  
}
```

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = 0;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= 3) {  
        System.out.println(tipo+  
                            "\t\t"+  
                            valorPiscina(area,tipo));  
        tipo = tipo+1;  
    }  
}
```

Ambos são equivalentes

Laços

```
while (CONDIÇÃO) {  
    //condição verdadeira  
    COMANDO1;  
    COMANDO2;  
    ...  
    COMANDOn;  
}
```

```
enquanto (CONDIÇÃO) {  
    //condição verdadeira  
    COMANDO1;  
    COMANDO2;  
    ...  
    COMANDOn;  
}
```

While diz que enquanto a condição for verdadeira, os comandos em seu corpo serão executados

- Inicialmente, testa a **condição**
- Se verdadeira, executa o corpo
- Se falsa, sai do *while*

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= PLASTICO) {  
        System.out.println(tipo+"\t\t"+  
                            valorPiscina(area,tipo));  
        tipo = tipo+1;  
    }  
}
```

Laços

- Ao **final do corpo**, testa novamente a condição (voltando ao início)
- Cada vez que o corpo é rodado chama-se **iteração**

```
public static void main(String[]  
                        args) {  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= PLASTICO) {  
        System.out.println(tipo+"\t\t"+  
                            valorPiscina(area,tipo));  
        tipo = tipo+1;  
    }  
}
```

- O que acontece se removermos a linha `tipo = tipo+1;`?

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= PLASTICO) {  
        System.out.println(tipo+"\t\t"+  
                            valorPiscina(area,tipo));  
  
    }  
}
```

Laços

- O que acontece se removermos a linha `tipo = tipo+1;`?
- A condição sempre será verdadeira, *tipo* sempre será ≤ 3

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= PLASTICO) {  
        System.out.println(tipo+"\t\t"+  
                            valorPiscina(area,tipo));  
  
    }  
}
```

Laços

- O que acontece se removermos a linha `tipo = tipo+1;`?
- A condição sempre será verdadeira, *tipo* sempre será ≤ 3
- Laço (ou loop) infinito

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= PLASTICO) {  
        System.out.println(tipo+"\t\t"+  
                            valorPiscina(area,tipo));  
    }  
}
```

- Sabendo que PLASTICO=3, há diferença entre o código visto anteriormente e esse?

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("Material  
                        \tValor");  
    while (tipo <= 3) {  
        System.out.println(tipo+"\t\t"+  
                            valorPiscina(area,tipo));  
        tipo = tipo+1;  
    }  
}
```

Laços

- Sabendo que PLASTICO=3, há diferença entre o código visto anteriormente e esse?
- Somente o acesso extra a PLASTICO na memória. Contudo, aumenta a legibilidade

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("Material  
                        \tValor");  
  
    while (tipo <= 3) {  
        System.out.println(tipo+"\t\t"+  
                            valorPiscina(area,tipo));  
        tipo = tipo+1;  
    }  
}
```

While

- Já temos o orçamento para um determinado tamanho e vários materiais

While

- Já temos o orçamento para um determinado tamanho e vários materiais
- Agora queremos poder comparar vários tamanhos de piscina de alvenaria, fixando o material
 - Digamos, 50m^2 , 100m^2 , 150m^2 e 200m^2

While

- Já temos o orçamento para um determinado tamanho e vários materiais
- Agora queremos poder comparar vários tamanhos de piscina de alvenaria, fixando o material
 - Digamos, 50m^2 , 100m^2 , 150m^2 e 200m^2
- Como fazer?

While

```
public static void main(String[] args) {  
    double area = 50;  
    int tipo = ALVENARIA;  
  
    System.out.println("Área\tValor");  
    while (area <= 200) {  
        System.out.println(area+"\t"+  
                             valorPiscina(area,tipo));  
        area = area+50;  
    }  
}
```

While

```
public static void main(String[] args) {  
    double area = 50;  
    int tipo = ALVENARIA;  
  
    System.out.println("Área\tValor");  
    while (area <= 200) {  
        System.out.println(area+"\t"+  
                             valorPiscina(area,tipo));  
        area = area+50;  
    }  
}
```

● E a saída é:

Área	Valor
50.0	75000.0
100.0	150000.0
150.0	225000.0
200.0	300000.0

While

Temos então os seguintes códigos:

```
public static void main(String[]  
                        args) {  
  
    double area = 100;  
    int tipo = ALVENARIA;  
  
    System.out.println("  
        Material\tValor");  
    while (tipo <= PLASTICO) {  
        System.out.println(tipo+"\t\t"  
            +valorPiscina(area,tipo));  
        tipo = tipo+1;  
    }  
}
```

```
public static void main(String[]  
                        args) {  
  
    double area = 50;  
    int tipo = ALVENARIA;  
  
    System.out.println("Área\t  
        Valor");  
    while (area <= 200) {  
        System.out.println(area+"\t"  
            +valorPiscina(area,tipo));  
        area = area+50;  
    }  
}
```

While

- Como faríamos para criar uma tabela que desse o orçamento para piscinas de várias áreas e materiais?

While

- Como faríamos para criar uma tabela que desse o orçamento para piscinas de várias áreas e materiais?

para cada uma das áreas:

para cada um dos materiais:

digamos o valor da piscina com esta área, feita com este material

While

- Como faríamos para criar uma tabela que desse o orçamento para piscinas de várias áreas e materiais?

para cada uma das áreas:

para cada um dos materiais:

digamos o valor da piscina com esta área, feita com este material

```
public static void main(String[]  
                        args) {  
    double area = 50;  
    int tipo;  
    System.out.println("Área\tMaterial  
                        \tValor");  
    while (area <= 200) {  
        tipo = ALVENARIA;  
        while (tipo <= PLASTICO) {  
            System.out.println(area+"\t"+  
                                tipo+"\t\t"+  
                                valorPiscina(area,tipo));  
            tipo = tipo+1;  
        }  
        area = area+50;  
    }  
}
```


While

```
public static void main(String[]
                        args) {
    double area = 50;
    int tipo;
    System.out.println("Área\tMaterial
                        \tValor");
    while (area <= 200) {
        tipo = ALVENARIA;
        while (tipo <= PLASTICO) {
            System.out.println(area+"\t"+
                                tipo+"\t\t"+
                                valorPiscina(area,tipo));
            tipo = tipo+1;
        }
        area = area+50;
    }
}
```

para cada uma das áreas:

para cada um dos materiais:
diga o valor da piscina com
esta área, feita com este
material

While

```
public static void main(String[]
                        args) {
    double area = 50;
    int tipo;
    System.out.println("Área\tMaterial
                        \tValor");
    while (area <= 200) {
        tipo = ALVENARIA;
        while (tipo <= PLASTICO) {
            System.out.println(area+"\t"+
                                tipo+"\t\t"+
                                valorPiscina(area,tipo));
            tipo = tipo+1;
        }
        area = area+50;
    }
}
```

para cada uma das áreas:

para cada um dos materiais:

diga o valor da piscina com
 esta área, feita com este
 material

While

```
public static void main(String[]
                        args) {
    double area = 50;
    int tipo;
    System.out.println("Área\tMaterial
                        \tValor");
    while (area <= 200) {
        tipo = ALVENARIA;
        while (tipo <= PLASTICO) {
            System.out.println(area+"\t"+
                                tipo+"\t\t"+
                                valorPiscina(area,tipo));
            tipo = tipo+1;
        }
        area = area+50;
    }
}
```

para cada uma das áreas:

para cada um dos materiais:

diga o valor da piscina com
 esta área, feita com este
 material

While

- Laço *while* aninhado

```
public static void main(String[]
                        args) {
    double area = 50;
    int tipo;
    System.out.println("Área\tMaterial
                        \tValor");
    while (area <= 200) {
        tipo = ALVENARIA;
        while (tipo <= PLASTICO) {
            System.out.println(area+"\t"+
                                tipo+"\t\t"+
                                valorPiscina(area,tipo));
            tipo = tipo+1;
        }
        area = area+50;
    }
}
```

While

- Laço *while* **aninhado**
- Note que a cada iteração do laço externo (area), temos que **reiniciar o laço interno** (tipo)

```
public static void main(String[]
                        args) {
    double area = 50;
    int tipo;
    System.out.println("Área\tMaterial
                        \tValor");
    while (area <= 200) {
        tipo = ALVENARIA;
        while (tipo <= PLASTICO) {
            System.out.println(area+"\t"+
                                tipo+"\t\t"+
                                valorPiscina(area,tipo));
            tipo = tipo+1;
        }
        area = area+50;
    }
}
```

While

```
public static void main(String[] args)
{
    double area = 50;
    int tipo=0;
    System.out.println("Área\tMaterial
                        \tValor");
    while (area <= 200) {
        tipo = ALVENARIA;
        while (tipo <= PLASTICO) {
            System.out.println(area+"\t"+
                                tipo+"\t\t"+
                                valorPiscina(area,tipo));
            tipo = tipo+1;
        }
        area = area+50;
    }
}
```

Área	Material	Valor
50.0	0	75000.0
50.0	1	55000.0
50.0	2	37500.0
50.0	3	25000.0
100.0	0	150000.0
100.0	1	110000.0
100.0	2	75000.0
100.0	3	50000.0
150.0	0	225000.0
150.0	1	165000.0
150.0	2	112500.0
150.0	3	75000.0
200.0	0	300000.0
200.0	1	220000.0
200.0	2	150000.0
200.0	3	100000.0

While

```
public static void main(String[] args)
{
    double area = 50;
    int tipo=0;
    System.out.println("Área\tMaterial
                        \tValor");
    while (area <= 200) {

        while (tipo <= PLASTICO) {
            System.out.println(area+"\t"+
                                tipo+"\t\t"+
                                valorPiscina(area,tipo));
            tipo = tipo+1;
        }
        area = area+50;
    }
}
```

E se removermos
tipo = ALVENARIA;?
Qual a saída?

While

```
public static void main(String[] args)
{
    double area = 50;
    int tipo=0;
    System.out.println("Área\tMaterial
                        \tValor");
    while (area <= 200) {

        while (tipo <= PLASTICO) {
            System.out.println(area+"\t"+
                               tipo+"\t\t"+
                               valorPiscina(area,tipo));
            tipo = tipo+1;
        }
        area = area+50;
    }
}
```

E se removermos
tipo = ALVENARIA;?
Qual a saída?

Área	Material	Valor
50.0	0	75000.0
50.0	1	55000.0
50.0	2	37500.0
50.0	3	25000.0

Videoaula

https:

[//www.youtube.com/watch?v=odhAEDP2ivQ&t=2s](https://www.youtube.com/watch?v=odhAEDP2ivQ&t=2s)

e

https:

[//www.youtube.com/watch?v=mOrsP1-wt1M&t=11s](https://www.youtube.com/watch?v=mOrsP1-wt1M&t=11s)