

Linguagem de Programação Orientada a Objetos

**Ambiente de Programação
Microsoft .Net
C# (C sharp)**

Prof. Laércio Silva

Controle de fluxo

- Maneira de simplificar os códigos quando se tem um número muito grande de comparações a serem feitas.
- A estrutura switch – case corresponde na verdade a vários if encadeados.
- Tudo que pode ser feito nessa estrutura poderá ser feito com sequências de ifs aninhados.

Controle de fluxo

Sintaxe:

```
switch ( < expressão >)
```

```
{
```

```
    case 1: sequência de instruções; break;
```

```
    case 2: sequência de instruções; break;
```

```
    case 3: sequência de instruções; break;
```

```
    case 4: sequência de instruções; break;
```

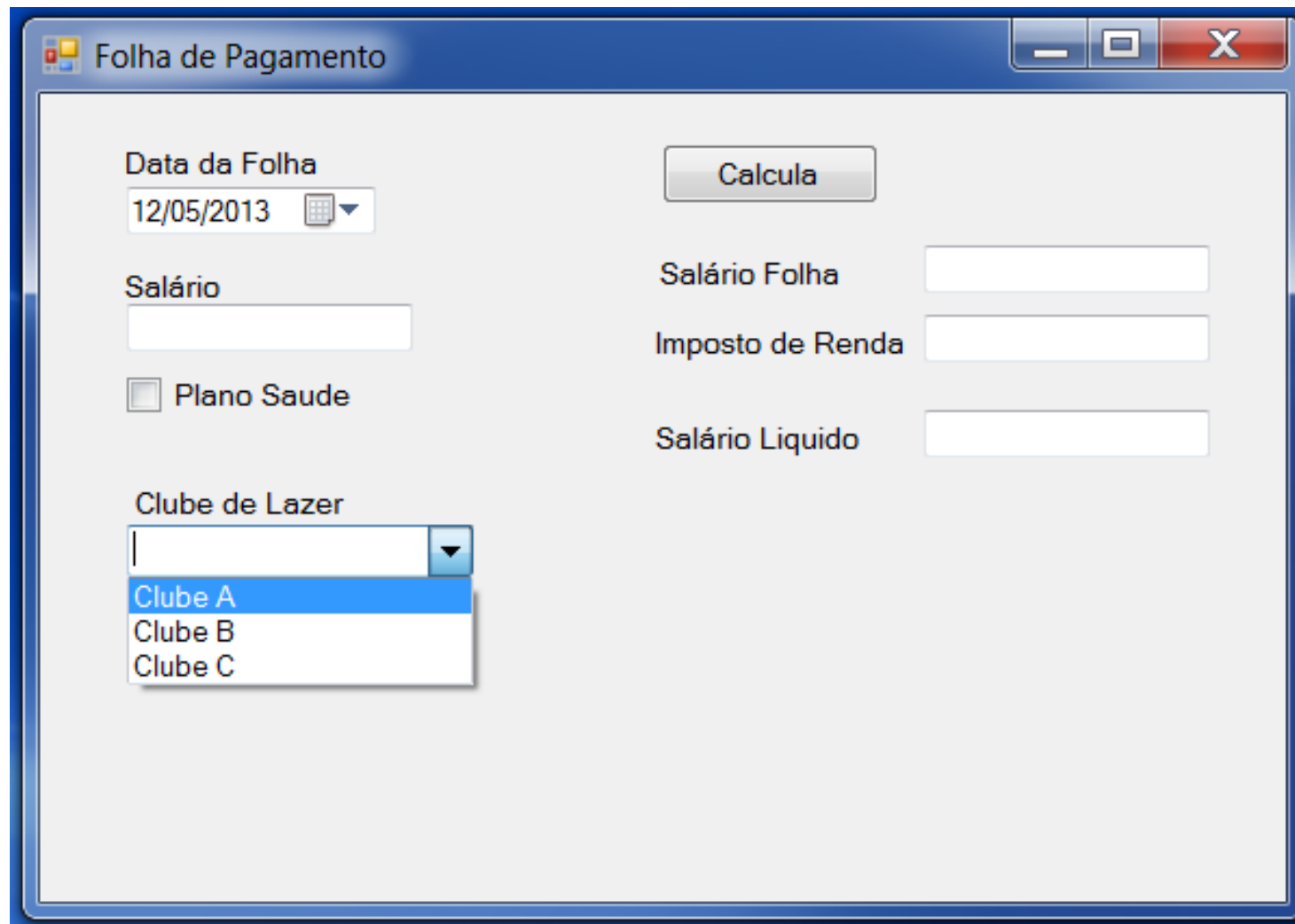
```
    Default : sequência de instruções;
```

```
}
```

Controle de fluxo

```
switch (comboBox1.Text)
{
    case "Clube A":
        vSalario = vSalario - 100;
        break;
    case "Clube B":
        vSalario = vSalario - 50;
        break;
    case "Clube C":
        vSalario = vSalario - 10;
        break;
    default:
        break;
}
```

Controle de fluxo - Exemplo

A screenshot of a software application window titled "Folha de Pagamento". The window has a blue title bar with standard Windows window controls (minimize, maximize, close). The main area is light gray and contains several input fields and a button. On the left side, there is a "Data da Folha" field with the date "12/05/2013" and a calendar icon, a "Salário" text input field, a checkbox labeled "Plano Saude", and a "Clube de Lazer" dropdown menu with a list showing "Clube A", "Clube B", and "Clube C". On the right side, there is a "Calcula" button, and three more text input fields labeled "Salário Folha", "Imposto de Renda", and "Salário Liquido".

Folha de Pagamento

Data da Folha
12/05/2013

Salário

☐ Plano Saude

Clube de Lazer
Clube A
Clube B
Clube C

Calcula

Salário Folha

Imposto de Renda

Salário Liquido

Estrutura de Repetição - While

- **Faz a comparação lógica antes de executar as instruções do seu corpo, de tal forma que as instruções serão executadas se a condição for verdadeira.**
- ***Isso significa que as instruções do laço não serão executadas se a condição nunca se tornar verdadeira;***

Sintaxe:

While (condição lógica)

{

<sequência de instruções>

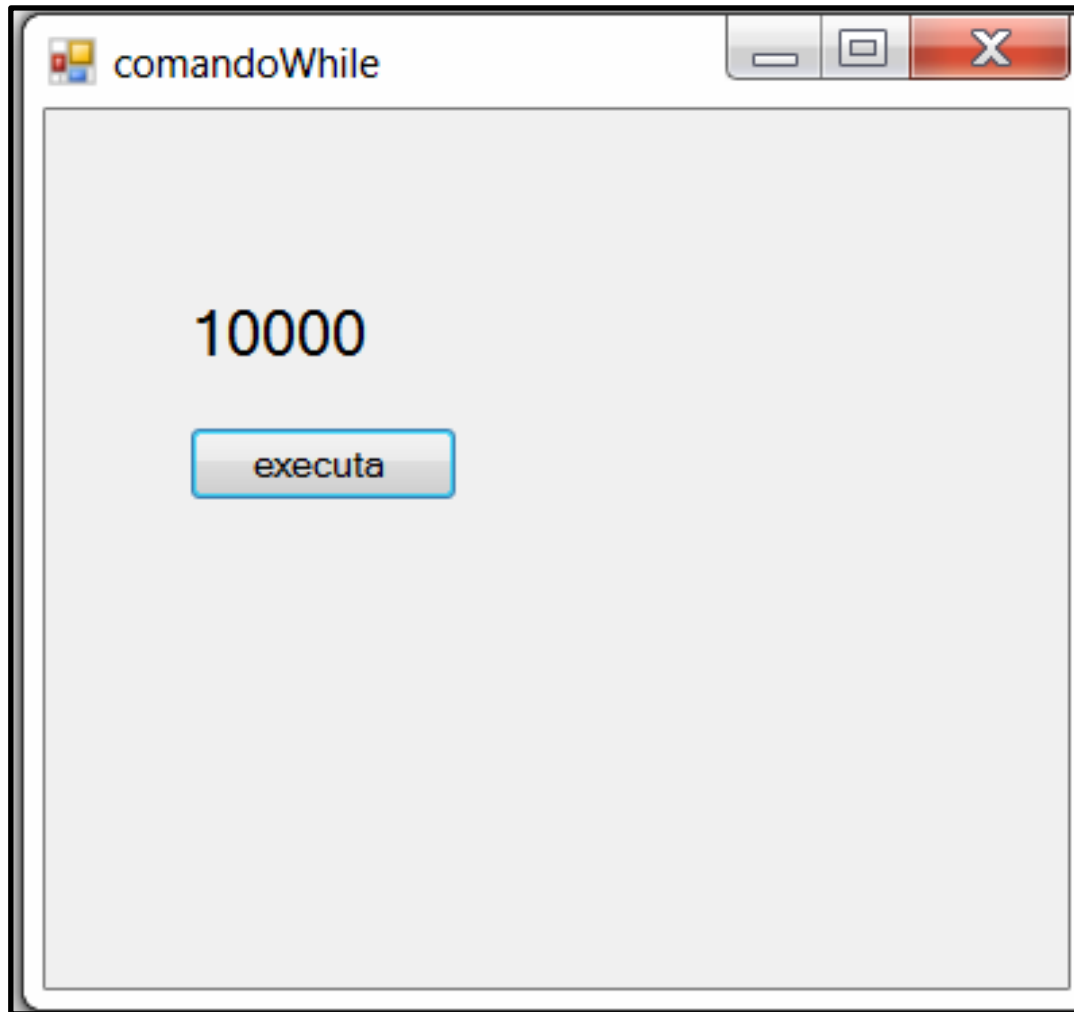
}



Estrutura de Repetição - While

```
private void button1_Click(object sender,  
EventArgs e)  
{  
    int c = 0;  
  
    while (c < 10000)  
    {  
        c = c + 1;  
        label1.Text = Convert.ToString(c);  
        label1.Refresh();  
    }  
}
```


Estrutura de Repetição - While



Estrutura de Repetição - do While



Executa as instruções enquanto a condição for verdadeira;

Nesse tipo de laço ***as instruções são executadas pelo menos uma vez até que a condição de controle seja avaliada;***

Estrutura de Repetição - do While

```
private void button1_Click(object sender, EventArgs e)
{
    int c = 0;

    do
    {
        c += 5;
        label1.Text = Convert.ToString(c);
        label1.Refresh();
    } while (c < 10000);
}
```

Estrutura de Repetição - do While





Estrutura de Repetição - for

Operação que deve ser repetida incondicionalmente em um determinado número de vezes para que se obtenha um resultado ou função.



Estrutura de Repetição - for

Sintaxe:

```
for( <variável de controle> <valor inicial> ; <valor limite> ; <incremento da variável> )  
{  
  
    itens que serão repetidos;  
  
}
```

Estrutura de Repetição - for

- **Variável de controle:** é uma variável numérica de tipo inteiro que vai controlar a repetição;
- **Valor inicial:** define o ponto de partida da contagem;
- **Valor limite:** é aquele com o qual o estado atual da variável, vai ser comparado para determinar se a repetição pára ou continua. A repetição cessa quando a variável de controle atinge o valor definido como limite;
- **Incremento da variável:** representa acréscimo que determinada variável irá sofrer cada vez que o laço de repetição for percorrido. Em geral, mas não obrigatório, o acréscimo é $x++$ que significa $x = x + 1$.



Estrutura de Repetição - for

```
private void button1_Click(object sender, EventArgs e)
{
    int fatorial = 1;
    for (int x = 1 ; x <= Convert.ToInt32(txtNumCalcular.Text) ; x++ )
    {
        fatorial = fatorial * x;
    }
    txtResultado.Text = Convert.ToString(fatorial);
}
```




Muito Obrigado

Até a Próxima Aula

Prof. Laércio Silva
Email: Indsilva@hotmail.com