



COLÉGIO PEDRO II – U. E. SÃO CRISTÓVÃO III

Professor: João Luiz Lagôas

Coordenação: Flávio Costa

LINGUAGEM DE PROGRAMAÇÃO 4

NOME: _____ TURMA: _____

EXERCÍCIO 01 – FORMMENU.CS

Implemente o método que trata o evento click do botão **btnMontagemTime**.

- `private void btnMontagemTime_Click(object sender, EventArgs e)`

Esse método deverá criar uma instância do formulário FormTeam e torná-lo ativo.

EXERCÍCIO 02 – FORMIMC.CS

O FormIMC é responsável por calcular o **Índice de Massa Corporal (IMC)** de uma pessoa, dados os seguintes parâmetros:

- Sexo (selecionado por RadioButton)
- Idade (NumericUpDown)
- Peso (NumericUpDown)
- Altura (NumericUpDown)

O cálculo leva em conta a faixa etária (maiores de 18 anos) e aplica classificações distintas conforme o sexo.

Atualmente, o IMC é calculado **apenas** quando o botão **btnCalcular** é clicado, o que chama o método:

- `private void btnCalcular_Click(object sender, EventArgs e)`

Sabendo disso, implemente o comportamento reativo no formulário, ou seja, crie os métodos que tratam os eventos dos seguintes controles de modo a atualizar a mensagem do IMC sempre que forem lançados.

- **numIdade.ValueChanged**
- **numPeso.ValueChanged**
- **numAltura.ValueChanged**
- **rdbMasculino.CheckedChanged**
- **rdbFeminino.CheckedChanged**



EXERCÍCIO 03 – FORMDICEGAME.CS

- a) Caso o usuário tenha sucesso, faça com que a aplicação emita um som de sucesso. Um arquivo de som chamado **success.wav** está disponível e pode ser executado usando a seguinte instrução:

```
SoundPlayer player = new SoundPlayer("assets//success.wav");
player.Play();
```

Atenção: garanta que sua aplicação não vai quebrar caso haja algum erro com a execução deste recurso externo.

- b) Aprimore a aplicação de modo a não permitir que o usuário insira um valor mínimo maior que o valor máximo. Você pode resolver isso de algumas formas:
- Manipulando os valores das `numericUpDown` sempre que uma delas muda;
 - Avaliando após clicar em Começar. Se este for o caso, emita uma `MessageBox` dizendo para o usuário colocar um valor válido.

EXERCÍCIO 04 – FORMTEAM.CS

- a) Implemente o método que trata o evento click do botão **btnParaBanco**.

- `private void btnParaBanco_Click(object sender, EventArgs e)`

Esse método deverá transferir o Jogador selecionado da List Box **lstBanco** para **lstCampo**.

- b) Atualize os métodos baixo:

- `private void btnParaCampo_Click(object sender, EventArgs e)`
- `private void btnParaBanco_Click(object sender, EventArgs e)`

Após inserir ou remover um jogador na List box **lstCampo**, é necessário atualizar a Text Box **txtPotencial**, informando o potencial do time.

O potencial de um time é a soma da força e velocidade de todos os jogadores que estão no campo.

Lembrete: para percorrer uma lista de objetos, neste caso, de Jogadores, use a estrutura `foreach`.

```
foreach(Jogador jogador in sua_lista_de_jogadores)
{
    //...
}
```