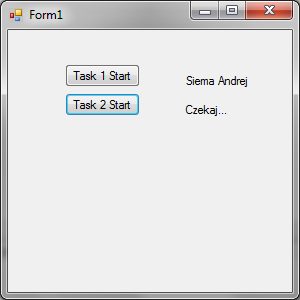
Akademia Marynarki Wojennej

|  |  |  |  |
| --- | --- | --- | --- |
| Imię i nazwisko | | Grupa | Data ćwiczenia |
| Maciej Maliszewski | | 125NCI\_A | 06.12.2015 r. |
| Tytuł ćwiczenia  **4** | **Programowanie asynchroniczne** | | |
| ***Programowanie równoległe i rozproszone***  ***dr inż.A.Zacniewski*** | | | |

Celem laboratorium było napisanie programu który wykonuje swoje operację asynchronicznie tworząc nowe procesy.

Program symuluje pobieranie danych z bazy danych pozostając responsywny na operacje użytkownika.



Kod programu:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Threading;

using System.Windows.Forms;

namespace OOR\_4

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

//Task.Factory.StartNew(() => Big("Elmo")).ContinueWith(t => label1.Text = t.Result, TaskScheduler.FromCurrentSynchronizationContext());

// label1.Text=Big ("Jonny");

CallBig();

label1.Text = "Czekaj...";

}

private async void CallBig()

{

var resoult = await BigAsync("Andrej");

label1.Text = resoult;

}

private Task<string> BigAsync(string name)

{

return Task.Factory.StartNew(() => Big(name));

}

private string Big(string name)

{

Thread.Sleep(2000);

return "Siema " + name;

}

private void button2\_Click(object sender, EventArgs e)

{

Task.Factory.StartNew(() => Big("Jonny")).ContinueWith(t => label2.Text = t.Result, TaskScheduler.FromCurrentSynchronizationContext());

label2.Text = "Czekaj...";

}

}

}