

**TUGAS JURNAL
MODUL 12**



Disusun Oleh :

Izzaty Zahara Br Barus – 23111040452

Kelas :

SE-07-02

Dosen :

Yudha Islami Sulistya

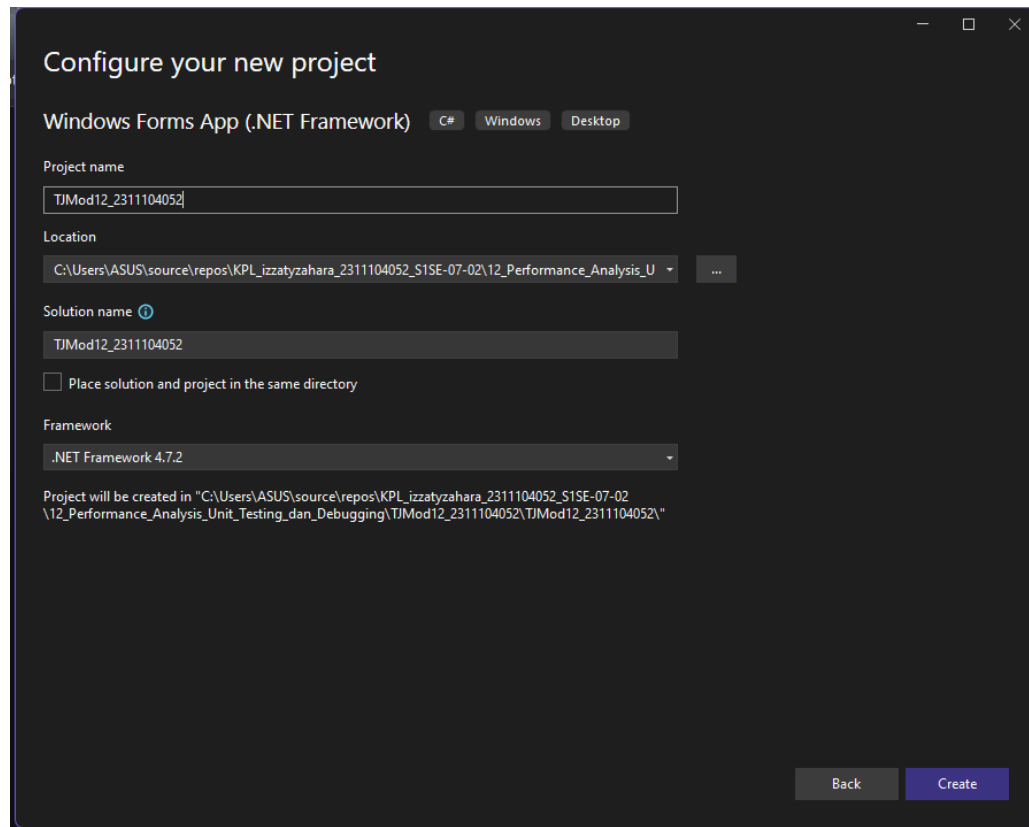
**PROGRAM STUDI SOFTWARE ENGINEERING
DIREKTORAT KAMPUS PURWOKERTO
TELKOM UNIVERSITY
PURWOKERTO
2025**

I. Link Github

- https://github.com/Izzaaaaaaaaa/KPL_izzatzahara_2311104052_S1SE-07-02.git

II. Penjelasan

1. Membuat New Project Menggunakan Blank Solution



2. Menambahkan textbox sebanyak 2 di “Form.cs [Design]”



4. Menambahkan syntax di class Form1.cs

```
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.Windows.Forms;

namespace TJMod12_2311104052
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void lbloutput_Click(object sender, EventArgs e)
        {

        }

        private void Form1_Load(object sender, EventArgs e)
        {

        }

        private void txtExponent_TextChanged(object sender, EventArgs e)
        {

        }

        private void txtBase_TextChanged(object sender, EventArgs e)
        {

        }

        public int CariNilaiPangkat(int a, int b)
        {
            if (b == 0) return 1;
            if (b < 0) return -1;
            if (b > 10 || a > 100) return -2;

            try
            {
                checked
            {
```

```

        int hasil = 1;
        for (int i = 0; i < b; i++)
        {
            hasil *= a;
        }
        return hasil;
    }
}
catch (OverflowException)
{
    return -3;
}
}

private void btnHitung_Click(object sender, EventArgs e)
{
    int a, b;

    bool isValid = int.TryParse(txtBase.Text, out a);
    bool isValid = int.TryParse(txtExponent.Text, out b);

    if (!isValid || !isValid)
    {
        lblOutput.Text = "Input tidak valid!";
        return;
    }

    int hasil = CariNilaiPangkat(a, b);
    lblOutput.Text = $"Hasil: {hasil}";
}
}
}

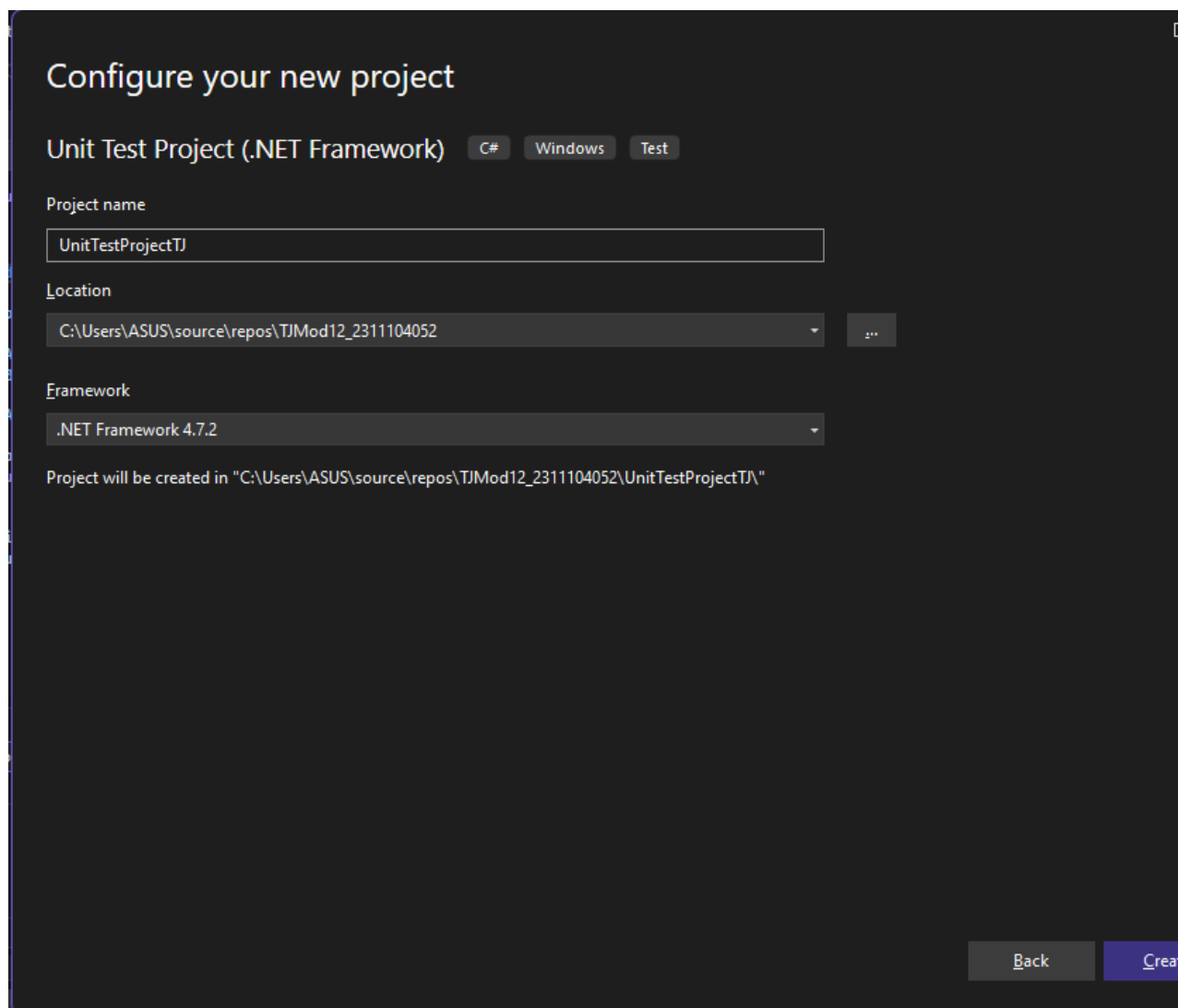
```

Penjelasan Singkat:

Program ini merupakan implementasi dari aplikasi GUI berbasis Windows Forms yang dibuat untuk menghitung nilai pangkat dari dua buah bilangan bulat, yaitu basis dan eksponen, dengan menggunakan aturan tertentu. Terdapat dua input berupa TextBox yaitu txtBase untuk nilai basis (a) dan txtExponent untuk nilai eksponen (b), satu Button bernama btnHitung untuk memproses perhitungan, serta satu Label bernama lblOutput untuk menampilkan hasilnya. Ketika tombol "Hitung" ditekan, program akan mengambil nilai dari kedua TextBox, memvalidasi bahwa input berupa bilangan bulat, lalu memanggil method CariNilaiPangkat(int a, int b) untuk menghitung hasil pangkat sesuai aturan.

Method CariNilaiPangkat merupakan method publik yang menerima dua parameter bilangan bulat, yaitu a dan b, dan mengembalikan nilai hasil pangkat a^b berdasarkan kondisi khusus yang sudah ditentukan. Jika b sama dengan nol, maka hasil selalu 1. Jika b bernilai negatif, maka akan mengembalikan nilai -1. Jika b lebih dari 10 atau a lebih dari 100, maka akan mengembalikan -2 sebagai kode kesalahan. Perhitungan pangkat dilakukan dengan iterasi dalam blok checked agar dapat mendeteksi overflow. Jika terjadi overflow (misalnya hasil terlalu besar melebihi batas integer positif), maka program akan menangkap OverflowException dan mengembalikan nilai -3. Dengan desain seperti ini, pengguna dapat mengetahui status perhitungan dan kesalahan melalui output yang ditampilkan pada Label

5. Menambah Project Baru menggunakan Unit Test Project



6. Menambah syntax Class PangkatTest.cs

```
using System;  
using Microsoft.VisualStudio.TestTools.UnitTesting;  
using TJMod12_2311104052;
```

```

[TestClass]
public class PangkatTests
{
    Form1 form = new Form1();

    [TestMethod]
    public void TestPangkatB0()
    {
        Assert.AreEqual(1, form.CariNilaiPangkat(0, 0));
    }

    [TestMethod]
    public void TestPangkatNegatif()
    {
        Assert.AreEqual(-1, form.CariNilaiPangkat(5, -2));
    }

    [TestMethod]
    public void TestBesarDari10()
    {
        Assert.AreEqual(-2, form.CariNilaiPangkat(5, 11));
    }

    [TestMethod]
    public void TestLebih100()
    {
        Assert.AreEqual(-2, form.CariNilaiPangkat(101, 3));
    }

    [TestMethod]
    public void TestOverflow()
    {
        Assert.AreEqual(-3, form.CariNilaiPangkat(9, 30));
    }

    [TestMethod]
    public void TestNormal()
    {
        Assert.AreEqual(27, form.CariNilaiPangkat(3, 3));
    }
}

```

Penjelasan Singkat:

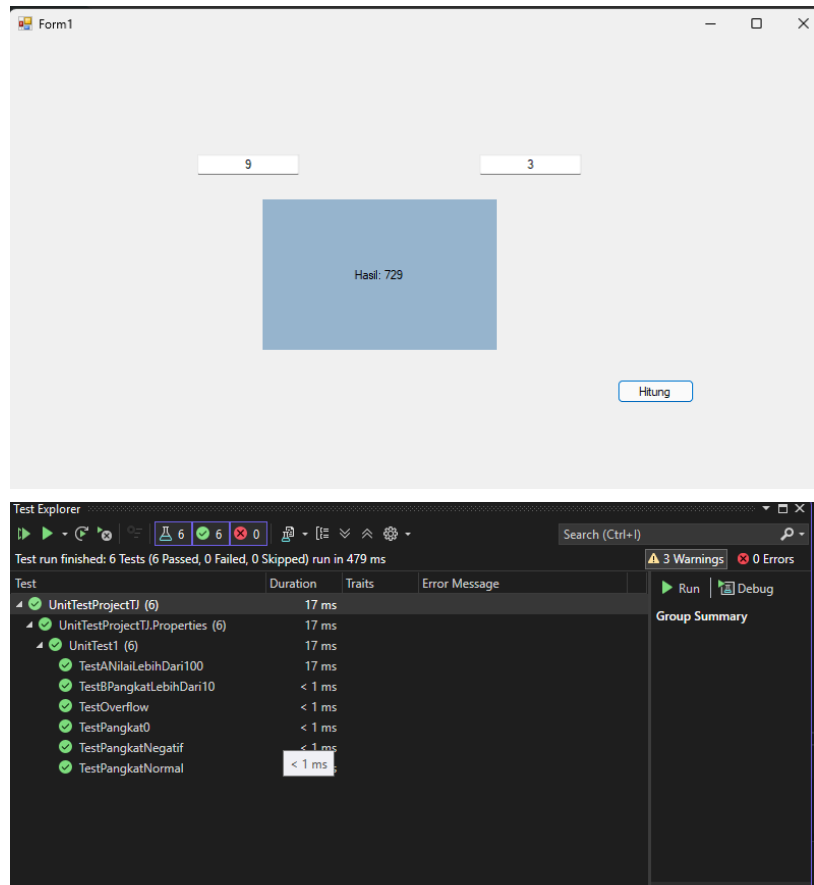
Kode unit test PangkatTests.cs merupakan bagian dari pengujian otomatis yang dibuat menggunakan framework MSTest untuk memastikan bahwa method CariNilaiPangkat dalam class Form1 berfungsi sesuai dengan spesifikasi yang ditentukan. Di dalam class ini terdapat enam metode pengujian yang masing-masing menguji kondisi khusus berdasarkan aturan perhitungan pangkat. Method TestPangkatB0 menguji kasus ketika eksponen bernilai 0, dan sesuai dengan aturan, hasil yang diharapkan adalah 1. Method TestPangkatNegatif menguji ketika eksponen negatif, yang seharusnya

menghasilkan nilai -1 sebagai penanda kesalahan. Selanjutnya, TestBesarDari10 menguji kondisi ketika eksponen melebihi angka 10, dan TestLebih100 menguji ketika nilai basis lebih dari 100, di mana keduanya harus menghasilkan -2 karena melebihi batas yang diizinkan.

Selain itu, TestOverflow menguji skenario ketika hasil pangkat terlalu besar dan menyebabkan overflow pada integer, sehingga diharapkan menghasilkan nilai -3. Terakhir, TestNormal menguji perhitungan pangkat dalam kondisi normal, yaitu 3 pangkat 3, yang seharusnya menghasilkan 27. Melalui unit test ini, semua jalur logika (branch) dalam fungsi CariNilaiPangkat telah diuji secara menyeluruh, sehingga dapat meningkatkan keandalan dan kestabilan program. Unit test ini juga berguna untuk mendeteksi kesalahan lebih awal jika terjadi perubahan pada fungsi di masa depan.

III. Hasil Running

1. Hasil Running



IV. Kesimpulan

Pengembangan aplikasi berbasis GUI menggunakan Windows Forms untuk perhitungan nilai pangkat telah berhasil dilaksanakan dengan baik. Fungsi utama CariNilaiPangkat mampu menangani berbagai kondisi input secara tepat, seperti kasus eksponen nol, eksponen negatif, batas maksimum nilai, dan overflow. Proses validasi input serta penanganan kesalahan dirancang dengan logika yang jelas dan memberikan output yang informatif bagi pengguna. Selain itu, proses pengujian menggunakan unit test dengan framework MSTest berhasil menguji seluruh cabang logika pada fungsi tersebut, sehingga keandalan dan stabilitas program dapat dipastikan. Adanya unit testing juga memberikan jaminan bahwa program tetap berfungsi sebagaimana mestinya apabila terjadi perubahan kode di masa mendatang. Secara keseluruhan, aplikasi ini tidak hanya memenuhi aspek fungsionalitas tetapi juga menunjukkan kualitas kode yang baik melalui praktik pengujian yang menyeluruh.