

Laporan Jurnal Modul 12

Nama : Kafka Putra Riyadi

Kelas : SE 07-02

NIM : 2311104041

A. Source Code Class Helper.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace JurnalModul12_2311104041
{
    public class Helper
    {
        public static 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 result = 1;
                    for (int i = 0; i < b; i++)
                    {
                        result *= a;
                    }
                    return result;
                }
            }
            catch (System.OverflowException)
            {
                return -3;
            }
        }
    }
}
```

Penjelasan :

Class Helper.cs merupakan implementasi dari fungsi CariNilaiPangkat yang berada dalam kelas Helper di namespace JurnalModul12_2311104041, yang berfungsi untuk menghitung nilai pangkat dari suatu bilangan bulat a yang dipangkatkan dengan bilangan bulat b (a^b). Fungsi ini bersifat static, sehingga dapat dipanggil langsung tanpa membuat objek dari kelas Helper.

Sebelum proses perhitungan dilakukan, fungsi ini terlebih dahulu memeriksa beberapa kondisi validasi:

1. Jika nilai b adalah 0, maka akan langsung mengembalikan 1 (karena apapun pangkat 0 adalah 1).
2. Jika nilai b negatif, maka mengembalikan -1 sebagai kode kesalahan karena pangkat negatif tidak didukung.
3. Jika nilai b lebih dari 10 atau a lebih dari 100, maka mengembalikan -2 untuk membatasi perhitungan yang terlalu besar.

Perhitungan pangkat dilakukan dalam blok try-catch dengan checked untuk mendeteksi kemungkinan overflow saat melakukan perkalian berulang. Jika terjadi overflow (hasil terlalu besar untuk ditampung oleh tipe data int), maka fungsi akan menangkap OverflowException dan mengembalikan -3 sebagai penanda kesalahan tersebut. Dengan demikian, fungsi ini tidak hanya menghitung pangkat, tetapi juga memberikan perlindungan terhadap kesalahan input dan potensi kesalahan perhitungan.

Source Code 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 JurnalModul12_2311104041
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void buttonCek_Click(object sender, EventArgs e)
        {
            if (int.TryParse(textBoxA.Text, out int a) && int.TryParse(textBoxB.Text, out int b))
            {
                int hasil = Helper.CariNilaiPangkat(a, b);
            }
        }
    }
}
```

```

        labelOutput.Text = $"Hasil Output: {hasil}";
    }
    else
    {
        labelOutput.Text = "Hasil Output: Input tidak valid!";
    }
}
}
}
}

```

Penjelasan :

Class Form1.cs merupakan bagian dari aplikasi **Windows Forms** dalam bahasa C#, yang mengatur interaksi pengguna pada form Form1 untuk menghitung nilai pangkat dari dua bilangan bulat menggunakan fungsi CariNilaiPangkat() dari kelas Helper. Ketika tombol buttonCek diklik (buttonCek_Click), program akan membaca nilai dari dua textbox (textBoxA dan textBoxB) yang mewakili bilangan pokok dan pangkat.

Jika kedua input tersebut valid (berhasil dikonversi ke tipe int menggunakan int.TryParse), maka program akan memanggil metode Helper.CariNilaiPangkat(a, b) untuk menghitung hasilnya, kemudian menampilkannya di labelOutput. Jika salah satu input tidak valid (misalnya huruf atau kosong), maka label akan menampilkan pesan kesalahan "Input tidak valid!".

Kode ini memberikan antarmuka grafis sederhana yang memungkinkan pengguna menghitung pangkat dua bilangan secara interaktif, sekaligus menangani validasi input agar program tidak crash akibat kesalahan pengguna.

Source Code Class Test1.cs

```

using Microsoft.VisualStudio.TestTools.UnitTesting;
using JurnalModul12_2311104041;

namespace UnitTest
{
    [TestClass]
    public class UnitTest1
    {
        [TestMethod]
        public void TestPangkatNormal()
        {
            Assert.AreEqual(8, Helper.CariNilaiPangkat(2, 3));
        }

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

```

```

    }

    [TestMethod]
    public void TestPangkatNegatif()
    {
        Assert.AreEqual(-1, Helper.CariNilaiPangkat(2, -3));
    }

    [TestMethod]
    public void TestInputBesar()
    {
        Assert.AreEqual(-2, Helper.CariNilaiPangkat(101, 5));
        Assert.AreEqual(-2, Helper.CariNilaiPangkat(3, 11));
    }

    [TestMethod]
    public void TestOverflow()
    {
        Assert.AreEqual(-3, Helper.CariNilaiPangkat(100, 10));
    }
}

```

Penjelasan :

Kode tersebut adalah implementasi unit test menggunakan framework MSTest untuk menguji keakuratan dan ketahanan fungsi CariNilaiPangkat() dari kelas Helper dalam namespace JurnalModul12_2311104041. Kelas UnitTest1 berisi lima metode pengujian ([TestMethod]), masing-masing menguji berbagai skenario pemanggilan fungsi:

1. TestPangkatNormal() menguji kasus normal di mana 2 pangkat 3 seharusnya menghasilkan 8.
2. TestPangkatZero() menguji kondisi dasar ketika 0^0 , yang secara konvensional dikembalikan sebagai 1.
3. TestPangkatNegatif() menguji input pangkat negatif, yang seharusnya mengembalikan -1 sesuai logika validasi fungsi.
4. TestInputBesar() menguji batas atas input, yaitu nilai a lebih dari 100 atau b lebih dari 10, yang seharusnya mengembalikan -2 sebagai kode kesalahan input terlalu besar.
5. TestOverflow() menguji skenario overflow (hasil terlalu besar), seperti 100^{10} , yang seharusnya memicu pengecualian overflow dan mengembalikan -3.

Secara keseluruhan, kode ini memastikan bahwa fungsi CariNilaiPangkat() tidak hanya memberikan hasil yang benar dalam kasus normal, tetapi juga menangani kondisi ekstrem dan kesalahan input dengan benar, yang merupakan praktik penting dalam pengujian perangkat lunak.

Source Code Class Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace JurnalModul12_2311104041
{
    static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Penjelasan :

Kode tersebut merupakan entry point atau titik awal eksekusi dari aplikasi Windows Forms berbasis .NET yang menggunakan namespace JurnalModul12_2311104041. Kode ini terletak di dalam kelas statis Program dan berisi metode Main() yang ditandai dengan atribut [STAThread], yang menunjukkan bahwa aplikasi menggunakan model *Single Threaded Apartment*—syarat wajib untuk aplikasi Windows Forms agar komponen GUI seperti Clipboard, Drag-and-Drop, atau MessageBox dapat bekerja dengan baik.

Di dalam Main(), terdapat tiga baris penting:

1. Application.EnableVisualStyles() mengaktifkan gaya visual modern Windows agar kontrol GUI (seperti tombol dan textbox) tampil sesuai dengan tema Windows saat ini.
2. Application.SetCompatibleTextRenderingDefault(false) mengatur metode rendering teks ke default yang lebih kompatibel (menggunakan GDI+).
3. Application.Run(new Form1()) meluncurkan dan menjalankan form utama (Form1) sebagai antarmuka pengguna aplikasi.

Secara keseluruhan, kode ini bertujuan untuk menyiapkan dan menjalankan antarmuka grafis aplikasi Windows Forms.

Output :

Modul 12 - Pangkat

Kalkulator Pangkat

Angka

Pangkat

Hitung Pangkat

Hasil Output: 4