



**TUGAS JURNAL**  
**KONSTRUKSI PERANGKAT LUNAK**  
**MODUL 14**

**D**  
**I**  
**S**  
**U**  
**S**  
**U**  
**N**

**OLEH :**  
**TIURMA GRACE ANGELINA – 2311104042**  
**S1-SE07-02**

**Dosen :**  
**YIS**

**Prodi S1 Rekaya Perangkat Lunak**  
**Direktorat Kampus Purwokerto**

## 1. REFACTORING DENGAN STANDAR CODE

TPMOD6

Class SayaTubeVideo.cs

Sebelum :

```
using System;

public class SayaTubeVideo
{
    private int id;
    private string title;
    private int playCount;

    public SayaTubeVideo(string title)
    {
        if (string.IsNullOrEmpty(title))
            throw new ArgumentException("Judul video tidak boleh kosong atau null.");

        if (title.Length > 100)
            throw new ArgumentException("Judul video tidak boleh lebih dari 100 karakter.");

        Random random = new Random();
        this.id = random.Next(10000, 99999);
        this.title = title;
        this.playCount = 0;
    }

    public void IncreasePlayCount(int count)
    {
        if (count > 10000000)
            throw new ArgumentException("Penambahan play count tidak boleh lebih dari 10.000.000.");

        try
        {
            checked
            {
                this.playCount += count;
            }
        }
        catch (OverflowException)
        {
            Console.WriteLine("Error: Play count melebihi batas maksimum integer!");
        }
    }
}
```

```

        public void PrintVideoDetails()
        {
            Console.WriteLine($"ID Video: {id}");
            Console.WriteLine($"Judul: {title}");
            Console.WriteLine($"Jumlah Play: {playCount}");
        }
    }
}

```

Setelah :

```

using System;

/// <summary>
/// Representasi video dengan fitur penambahan play count dan cetak detail.
/// </summary>
public class SayaTubeVideo
{
    private int _id;
    private string _title;
    private int _playCount;

    public SayaTubeVideo(string title)
    {
        if (string.IsNullOrEmpty(title))
            throw new ArgumentException("Judul video tidak boleh kosong atau null.");

        if (title.Length > 100)
            throw new ArgumentException("Judul video tidak boleh lebih dari 100 karakter.");

        Random random = new Random();
        _id = random.Next(10000, 99999);
        _title = title;
        _playCount = 0;
    }

    /// <summary>
    /// Menambahkan jumlah play dengan batas maksimal 10 juta.
    /// </summary>
    public void IncreasePlayCount(int count)
    {
        if (count > 10000000)
            throw new ArgumentException("Penambahan play count tidak boleh lebih dari 10.000.000.");

        try

```

```

        {
            checked
            {
                _playCount += count;
            }
        }
    } catch (OverflowException)
    {
        Console.WriteLine("Error: Play count melebihi batas maksimum integer!");
    }
}

/// <summary>
/// Mencetak detail video.
/// </summary>
public void PrintVideoDetails()
{
    Console.WriteLine($"ID Video : {_id}");
    Console.WriteLine($"Judul : {_title}");
    Console.WriteLine($"Jumlah Play: {_playCount}");
}
}

```

Penjelasan :

Konstruktor SayaTubeVideo(string title) bertugas menginisialisasi objek video dengan:

- Validasi input title agar **tidak kosong/null** dan **maksimal 100 karakter**.
- Menggunakan objek Random untuk mengatur id secara acak di rentang 10000–99999.
- Inisialisasi playCount ke 0 sebagai nilai awal.

Ini memastikan bahwa setiap objek video memiliki identitas unik dan judul yang valid.

Method IncreasePlayCount() bertanggung jawab untuk menambah jumlah playCount video.

- Jika nilai count melebihi 10.000.000 maka dilempar ArgumentException.
- Proses penambahan dilakukan dalam blok checked untuk menangani potensi integer overflow. Bila terjadi overflow, akan ditangani dan menampilkan pesan kesalahan.

Fungsi ini memperkuat ketahanan kode terhadap input tidak wajar dan kesalahan runtime.

Method PrintVideoDetails() mencetak detail video seperti:

- ID Video
- Judul
- Jumlah Play Count

Dengan mencetak format terstruktur, method ini memudahkan proses debugging dan monitoring informasi video.

Class program.cs

Sebelum :

```
class Program
{
    static void Main()
    {
        try
        {
            SayaTubeVideo video1 = new SayaTubeVideo("Tutorial Design By
Contract – Tiorma Grace Angelina");
            video1.IncreasePlayCount(5000000);
            video1.PrintVideoDetails();

            SayaTubeVideo video2 = new SayaTubeVideo("Test Overflow");
            for (int i = 0; i < 10; i++)
            {
                video2.IncreasePlayCount(int.MaxValue);
            }
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Exception: {ex.Message}");
        }
    }
}
```

Setelah :

```
public class Program
{
    public static void Main()
    {
        try
        {
            SayaTubeVideo video1 = new SayaTubeVideo("Tutorial Design By
Contract – Tiorma Grace Angelina");
            video1.IncreasePlayCount(5000000);
            video1.PrintVideoDetails();

            SayaTubeVideo video2 = new SayaTubeVideo("Test Overflow");

            for (int i = 0; i < 10; i++)
            {
                video2.IncreasePlayCount(int.MaxValue);
            }
        }
        catch (Exception ex)
        {

```

```

        Console.WriteLine($"Exception: {ex.Message}");
    }
}
}

```

Penjelasan :

Method Main() adalah titik awal eksekusi program.

- Membuat objek SayaTubeVideo dan memanggil method IncreasePlayCount serta PrintVideoDetails.
- Kasus video2 menunjukkan simulasi overflow dengan menambahkan int.MaxValue sebanyak 10 kali.
- Seluruh blok dilindungi oleh try-catch agar error yang terjadi dapat ditangani tanpa crash program.

Melalui proses refactoring ini, program menjadi:

- **Lebih aman** dari kesalahan input dan overflow.
- **Lebih terbaca dan terstruktur**, mengikuti standar C#/.NET.
- **Lebih terawat**, karena konvensi penamaan dan dokumentasi yang baik.

Aspek	Sebelum Refactor	Setelah Refactor	Penjelasan
<b>Naming Convention</b>	title, playCount	_title, _playCount	Private attributes menggunakan _camelCase untuk membedakan dari argumen.
<b>Method Naming</b>	IncreasePlayCount()	IncreasePlayCount()	Sudah menggunakan PascalCase sesuai standar C#.
<b>Whitespace &amp; Indentation</b>	Tidak konsisten	Konsisten 4 spasi	Diselaraskan agar mudah dibaca dan terstruktur.
<b>Comments</b>	Belum ada	Ditambahkan ///	Komentar dokumentasi ditambahkan untuk menjelaskan fungsi.

Aspek	Sebelum Refactor	Setelah Refactor	Penjelasan
Exception Handling	Ada, tapi belum maksimal	Lebih lengkap	Validasi dan pengecekan overflow ditingkatkan.

Output :

The screenshot shows the Visual Studio IDE with a C# program named 'SayaTubeVideo.cs' being debugged. The program is running on 'Any CPU' and has reached a breakpoint at line 13. The output window shows the following text:

```

ID Video : 58277
Judul : Tutorial Design By Contract - Tiurma Grace Angelina
Jumlah Play: 5000000
Exception: Penambahan play count tidak boleh lebih dari 10.000.000.
C:\Users\USER\source\repos\KPL_TiurmaGraceA_2311104042_S1SE-07-02\14_Clean_Code\tjmod14_2311104042\tjmod14_2311104042\bin\Debug\net8.0\tjmod14_2311104042.exe (process 8100) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

```

The Solution Explorer on the right shows the project structure for 'tjmod14\_2311104042', including 'Dependencies', 'Program.cs', and 'SayaTubeVideo.cs'. The Output window at the bottom shows the program's output and the exception message.