

LAPORAN PRAKTIKUM
JURNAL MODUL 14



Nama :

Aulia Jasifa Br Ginting 2311104060

S1SE-07-02

Dosen :

Yudha Islami Sulistya, S.Kom., M.Kom

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

I. LINK GITHUB

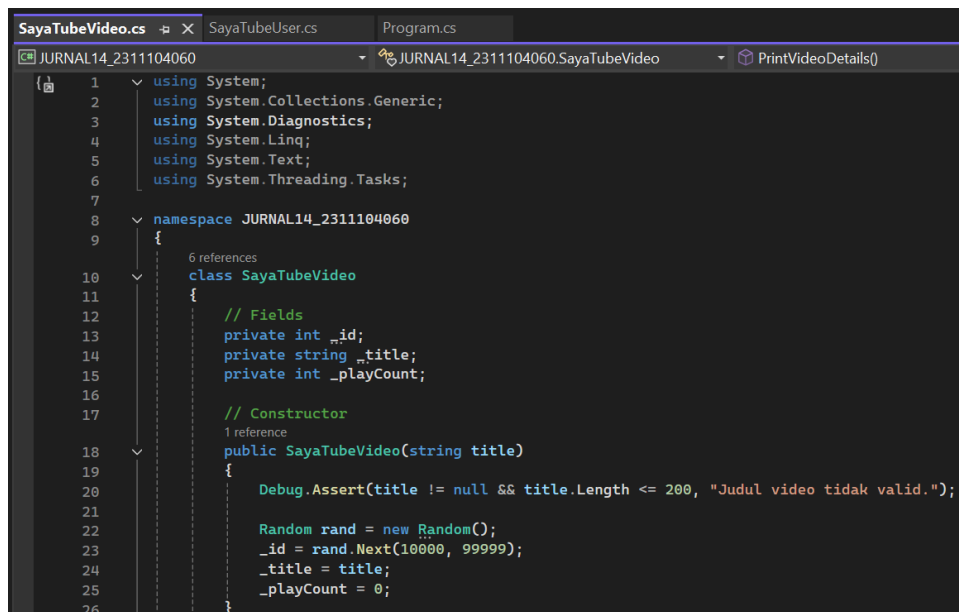
https://github.com/auliajsf06/KPL_Aulia-Jasifa-Br-Ginting_2311104060_SE-07-02

II. MEMBUAT PROJECT MODUL

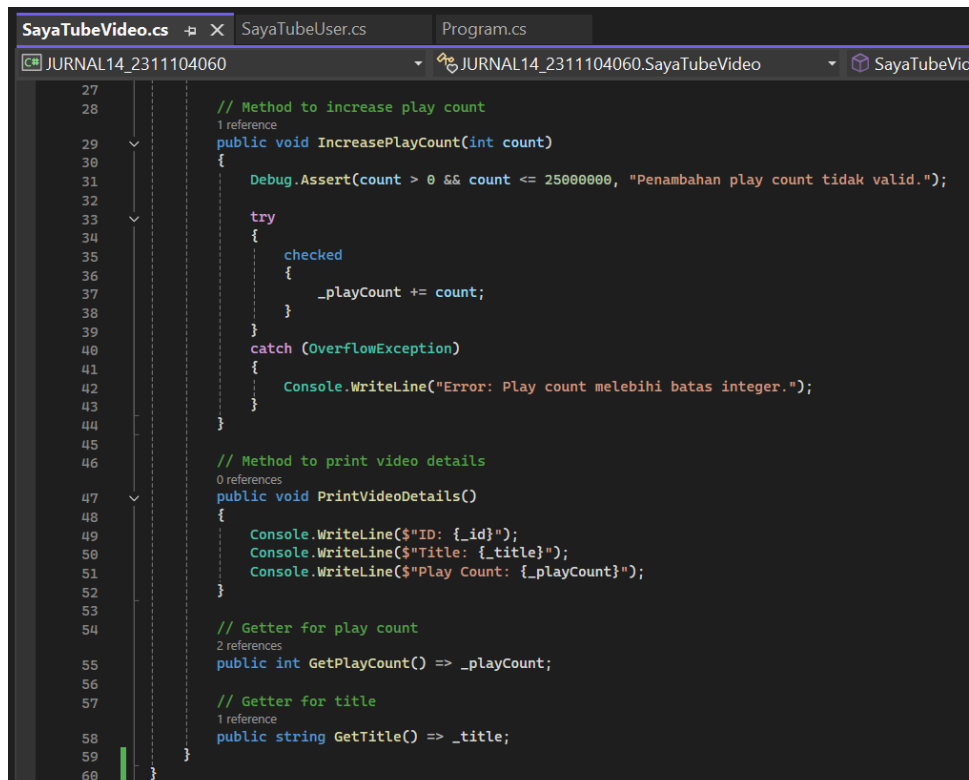
Mengambil (Copy Paste) dari Jurnal Modul 6

III. REFACTORING DENGAN STANDAR CODE

Class “SayaTubeVideo.cs”



```
1 using System;
2 using System.Collections.Generic;
3 using System.Diagnostics;
4 using System.Linq;
5 using System.Text;
6 using System.Threading.Tasks;
7
8 namespace JURNAL14_2311104060
9 {
10     class SayaTubeVideo
11     {
12         // Fields
13         private int _id;
14         private string _title;
15         private int _playCount;
16
17         // Constructor
18         public SayaTubeVideo(string title)
19         {
20             Debug.Assert(title != null && title.Length <= 200, "Judul video tidak valid.");
21
22             Random rand = new Random();
23             _id = rand.Next(10000, 99999);
24             _title = title;
25             _playCount = 0;
26         }
27     }
28 }
```



```
27
28 // Method to increase play count
29 public void IncreasePlayCount(int count)
30 {
31     Debug.Assert(count > 0 && count <= 25000000, "Penambahan play count tidak valid.");
32
33     try
34     {
35         checked
36         {
37             _playCount += count;
38         }
39     }
40     catch (OverflowException)
41     {
42         Console.WriteLine("Error: Play count melebihi batas integer.");
43     }
44 }
45
46 // Method to print video details
47 public void PrintVideoDetails()
48 {
49     Console.WriteLine($"ID: {_id}");
50     Console.WriteLine($"Title: {_title}");
51     Console.WriteLine($"Play Count: {_playCount}");
52 }
53
54 // Getter for play count
55 public int GetPlayCount() => _playCount;
56
57 // Getter for title
58 public string GetTitle() => _title;
59
60 }
```

Penjelasan Syntax “SayaTubeVideo.cs”

Kelas SayaTubeVideo merupakan representasi dari sebuah entitas video pada sistem

simulasi video seperti YouTube. Kelas ini dirancang untuk menyimpan informasi dasar mengenai sebuah video, seperti ID unik, judul video, dan jumlah tayangan (play count). Dengan membuat kelas ini, kita dapat mengenkapsulasi semua data dan logika terkait sebuah video dalam satu struktur yang rapi dan mudah digunakan di bagian program lainnya. Kelas ini juga menyertakan fitur validasi dan penanganan error guna memastikan data yang digunakan selalu berada dalam kondisi valid.

1. Field (Atribut Private)

Di dalam kelas `SayaTubeVideo`, terdapat tiga atribut (field) privat yaitu `_id`, `_title`, dan `_playCount`. Atribut `_id` digunakan untuk menyimpan identitas unik dari video yang dibuat secara acak. Atribut `_title` berfungsi untuk menyimpan judul video, sedangkan `_playCount` menyimpan jumlah tayangan video. Ketiganya bersifat `private` untuk menerapkan prinsip enkapsulasi dalam pemrograman berorientasi objek, yang bertujuan membatasi akses langsung dari luar kelas dan hanya bisa diakses melalui method tertentu.

2. Constructor

Constructor dalam kelas `SayaTubeVideo` bertugas untuk menginisialisasi objek ketika pertama kali dibuat. Pada constructor ini, terdapat validasi menggunakan `Debug.Assert` untuk memastikan bahwa judul video (title) yang dimasukkan tidak null dan panjangnya tidak melebihi 200 karakter. Setelah lolos validasi, sistem akan secara otomatis memberikan ID acak pada video menggunakan objek `Random` dengan nilai antara 10000 sampai 99999. Selain itu, nilai awal dari `playCount` juga diatur menjadi 0, menandakan bahwa video tersebut belum pernah ditonton saat pertama kali dibuat.

3. Method

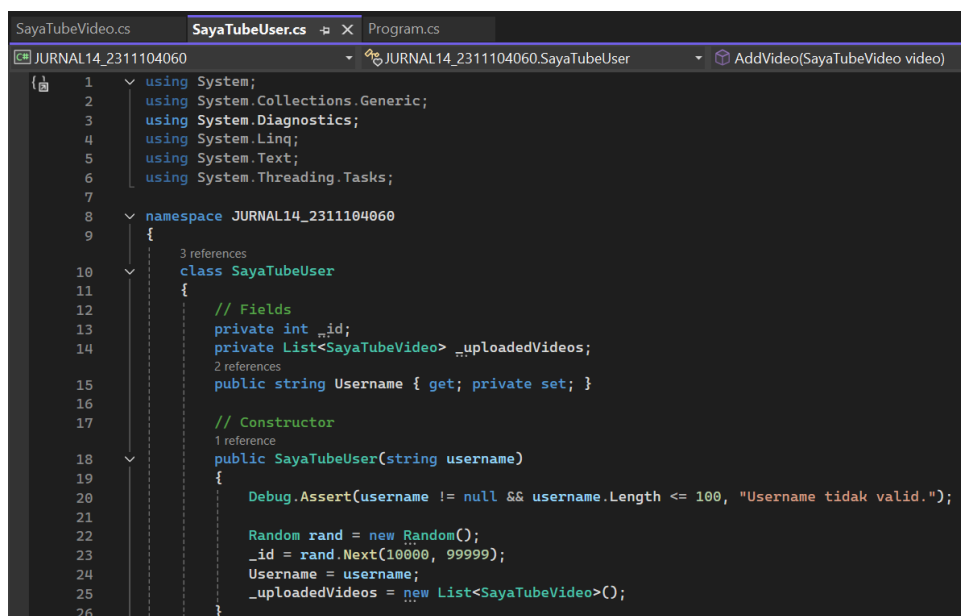
Kelas ini memiliki beberapa method penting. Pertama adalah `IncreasePlayCount(int count)`, yang digunakan untuk menambahkan jumlah tayangan video. Method ini melakukan validasi input dengan `Debug.Assert` untuk memastikan bahwa nilai count berada dalam batas yang wajar, yakni lebih dari 0 dan tidak lebih dari 25 juta. Penambahan `playCount` dilakukan dalam blok `checked` untuk mencegah terjadinya overflow pada tipe data integer. Jika terjadi overflow, program akan menangkapnya dengan `catch` dan mencetak pesan kesalahan. Kedua adalah `PrintVideoDetails()`, method yang menampilkan informasi video seperti ID, judul, dan jumlah tayangan ke konsol. Method ini memudahkan proses debugging atau pelaporan data video.

4. Getter

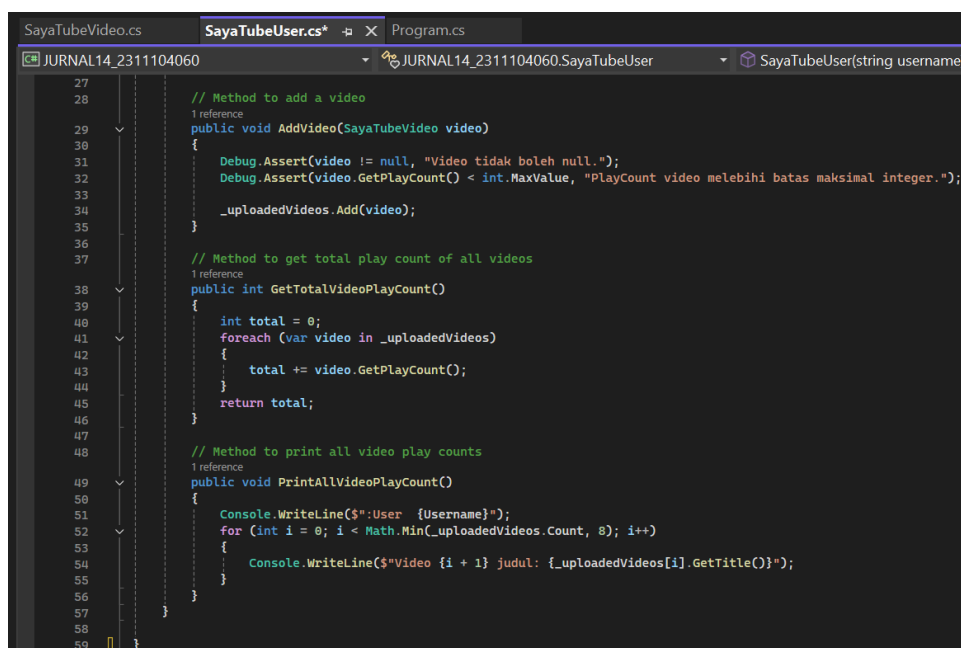
Selain method utama, kelas ini juga menyediakan dua method getter, yaitu

GetPlayCount() dan GetTitle(). Kedua method ini digunakan untuk mengakses nilai dari atribut `_playCount` dan `_title` tanpa harus memberikan akses langsung terhadap field-nya. Getter ini penting dalam menjaga prinsip enkapsulasi, karena memberikan kontrol penuh atas bagaimana data internal dari objek bisa diakses oleh bagian lain dari program tanpa membiarkan field tersebut dimodifikasi sembarangan dari luar kelas.

Class “SayaTubeUser.cs”



```
1  using System;
2  using System.Collections.Generic;
3  using System.Diagnostics;
4  using System.Linq;
5  using System.Text;
6  using System.Threading.Tasks;
7
8  namespace JURNAL14_2311104060
9  {
10     class SayaTubeUser
11     {
12         // Fields
13         private int _id;
14         private List<SayaTubeVideo> _uploadedVideos;
15         public string Username { get; private set; }
16
17         // Constructor
18         public SayaTubeUser(string username)
19         {
20             Debug.Assert(username != null && username.Length <= 100, "Username tidak valid.");
21
22             Random rand = new Random();
23             _id = rand.Next(10000, 99999);
24             Username = username;
25             _uploadedVideos = new List<SayaTubeVideo>();
26         }
27     }
28 }
```



```
27
28 // Method to add a video
29 public void AddVideo(SayaTubeVideo video)
30 {
31     Debug.Assert(video != null, "Video tidak boleh null.");
32     Debug.Assert(video.GetPlayCount() < int.MaxValue, "PlayCount video melebihi batas maksimal integer.");
33     _uploadedVideos.Add(video);
34 }
35
36 // Method to get total play count of all videos
37 public int GetTotalVideoPlayCount()
38 {
39     int total = 0;
40     foreach (var video in _uploadedVideos)
41     {
42         total += video.GetPlayCount();
43     }
44     return total;
45 }
46
47 // Method to print all video play counts
48 public void PrintAllVideoPlayCount()
49 {
50     Console.WriteLine($"User {Username}");
51     for (int i = 0; i < Math.Min(_uploadedVideos.Count, 8); i++)
52     {
53         Console.WriteLine($"Video {i + 1} judul: {_uploadedVideos[i].GetTitle()}");
54     }
55 }
56
57
58
59 }
```

Penjelasan Syntax “SayaTubeUser.cs”

Kelas `SayaTubeUser` merupakan representasi dari seorang pengguna (user) dalam sistem simulasi berbasis video, seperti platform YouTube versi sederhana. Kelas ini bertanggung jawab untuk menyimpan data pengguna seperti ID unik, nama pengguna, serta daftar video yang telah diunggah oleh pengguna tersebut. Dengan

mengorganisasi informasi ini ke dalam satu kelas, program dapat dengan mudah mengelola dan mengakses data pengguna serta video-videonya secara terstruktur dan terenkapsulasi.

1. Field (Atribut Private dan Public)

Terdapat tiga field dalam kelas ini: `_id`, `_uploadedVideos`, dan `Username`. Field `_id` adalah identitas unik yang diberikan secara acak kepada setiap pengguna. Field `_uploadedVideos` bertipe `List<SayaTubeVideo>` dan digunakan untuk menyimpan daftar video yang diunggah oleh pengguna. Field `Username` adalah properti publik yang hanya memiliki akses `get` (melalui `private set`), sehingga nilainya dapat dibaca dari luar tetapi tidak bisa diubah sembarangan. Struktur field ini mengikuti prinsip enkapsulasi yang baik dalam pemrograman berorientasi objek.

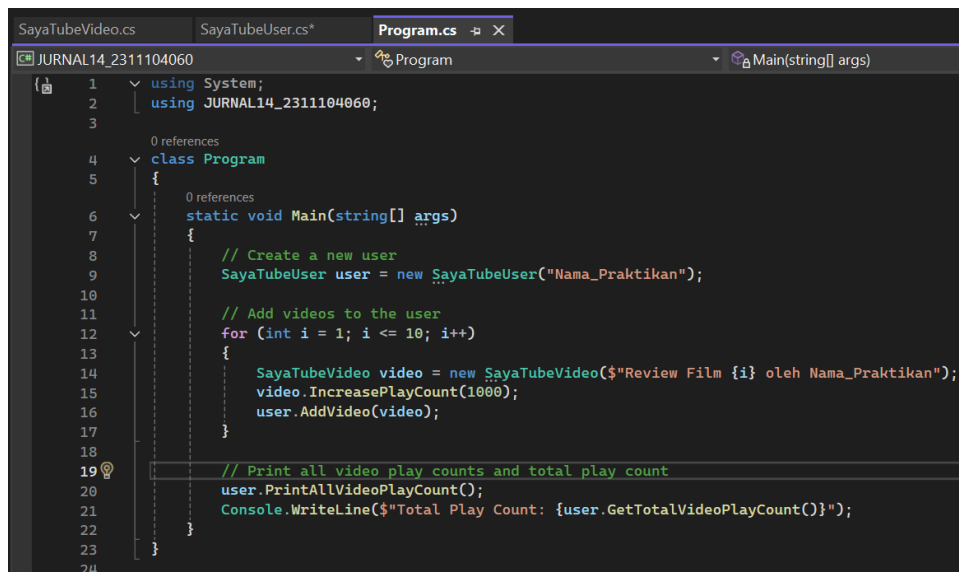
2. Constructor

Constructor `SayaTubeUser(string username)` digunakan untuk menginisialisasi objek pengguna baru. Constructor ini melakukan validasi menggunakan `Debug.Assert` untuk memastikan bahwa nama pengguna tidak `null` dan panjangnya tidak melebihi 100 karakter. Jika validasi berhasil, sistem akan memberikan ID acak kepada pengguna menggunakan objek `Random`, dan menetapkan username berdasarkan parameter yang diberikan. Selain itu, sebuah objek `List` kosong dari `SayaTubeVideo` juga dibuat untuk menyimpan video yang nantinya akan diunggah oleh pengguna tersebut.

3. Method

Kelas ini memiliki tiga method penting. Pertama, `AddVideo(SayaTubeVideo video)` digunakan untuk menambahkan video ke dalam daftar `uploadedVideos`. Method ini memeriksa bahwa objek video yang diberikan tidak `null` dan jumlah `play count` video tidak melebihi batas maksimum dari tipe data `integer` (`int.MaxValue`). Kedua, `GetTotalVideoPlayCount()` digunakan untuk menghitung total jumlah tayangan dari semua video yang telah diunggah oleh pengguna. Ini dilakukan dengan menjumlahkan `play count` dari setiap video dalam daftar menggunakan perulangan `foreach`. Ketiga, `PrintAllVideoPlayCount()` berfungsi menampilkan daftar hingga 8 video pertama milik pengguna, dengan menampilkan judul masing-masing video. Informasi ini dicetak ke konsol sebagai bentuk pelaporan aktivitas pengguna.

Class “Program.cs”



```
1 using System;
2 using JURNAL14_2311104060;
3
4 class Program
5 {
6     static void Main(string[] args)
7     {
8         // Create a new user
9         SayaTubeUser user = new SayaTubeUser("Nama_Praktikan");
10
11         // Add videos to the user
12         for (int i = 1; i <= 10; i++)
13         {
14             SayaTubeVideo video = new SayaTubeVideo($"Review Film {i} oleh Nama_Praktikan");
15             video.IncreasePlayCount(1000);
16             user.AddVideo(video);
17         }
18
19         // Print all video play counts and total play count
20         user.PrintAllVideoPlayCount();
21         Console.WriteLine($"Total Play Count: {user.GetTotalVideoPlayCount()}");
22     }
23 }
24
```

Penjelasan Syntax “Program.cs”

Kelas Program berisi method utama (Main) yang merupakan titik masuk (entry point) dari program ketika dijalankan. Di dalam method Main, dilakukan serangkaian proses untuk mensimulasikan penggunaan objek SayaTubeUser dan SayaTubeVideo dalam konteks sederhana, yaitu menambahkan video ke pengguna dan menampilkan datanya. Kode ini berfungsi untuk menguji sekaligus mendemonstrasikan integrasi antara kelas SayaTubeUser dan SayaTubeVideo yang telah didefinisikan sebelumnya.

1. Instansiasi Objek User

Pada bagian awal method Main, sebuah objek dari kelas SayaTubeUser dibuat dengan nama pengguna “Nama_Praktikan”. Ini menunjukkan bahwa satu entitas pengguna telah terdaftar di sistem. Proses ini juga memicu constructor SayaTubeUser, yang secara otomatis menetapkan ID acak dan memvalidasi username yang diberikan.

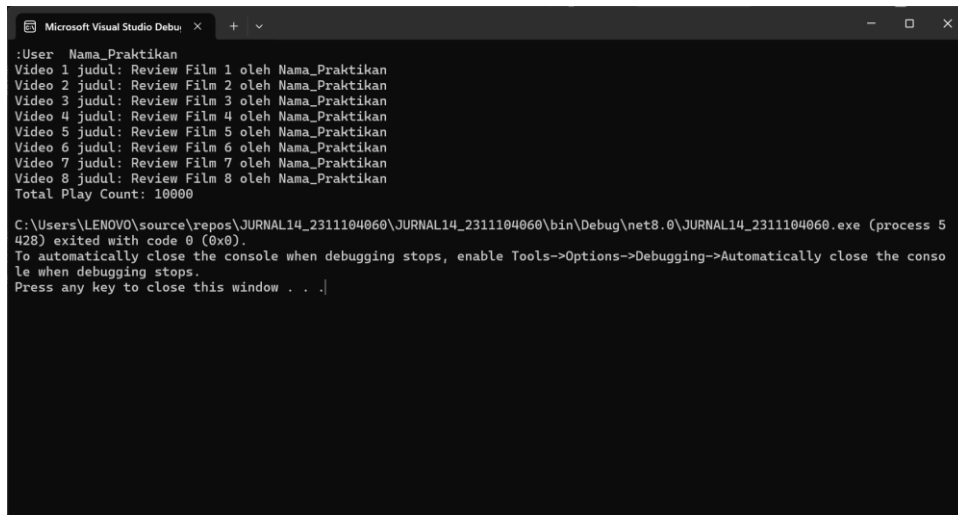
2. Penambahan Video ke Pengguna

Setelah membuat pengguna, dilakukan perulangan for sebanyak 10 kali, dari 1 hingga 10. Di setiap iterasi, dibuat objek SayaTubeVideo baru dengan judul yang mengikuti format “Review Film {i} oleh Nama_Praktikan”, sehingga menghasilkan judul yang unik untuk setiap video. Setiap video kemudian ditambahkan 1000 tayangan menggunakan method IncreasePlayCount(1000). Setelah itu, video tersebut ditambahkan ke daftar video milik pengguna menggunakan method AddVideo(video). Ini menunjukkan proses yang menggambarkan bagaimana seorang pengguna bisa mengunggah banyak video dengan play count tertentu.

3. Menampilkan Informasi Video dan Total Play Count

Setelah semua video ditambahkan, method `PrintAllVideoPlayCount()` dipanggil untuk menampilkan judul-judul dari maksimal 8 video pertama yang telah diunggah oleh pengguna ke layar konsol. Ini memberi gambaran umum kepada pengguna tentang video-video yang mereka miliki. Selanjutnya, program menampilkan total jumlah tayangan dari seluruh video menggunakan `Console.WriteLine` yang mencetak hasil dari pemanggilan method `GetTotalVideoPlayCount()`. Hasil ini memperlihatkan total play count secara agregat sebagai bentuk rekapitulasi aktivitas pengguna dalam platform.

II. OUTPUT NYA



```
Microsoft Visual Studio Debu x + v - □ x
:User Nama_Praktikan
Video 1 judul: Review Film 1 oleh Nama_Praktikan
Video 2 judul: Review Film 2 oleh Nama_Praktikan
Video 3 judul: Review Film 3 oleh Nama_Praktikan
Video 4 judul: Review Film 4 oleh Nama_Praktikan
Video 5 judul: Review Film 5 oleh Nama_Praktikan
Video 6 judul: Review Film 6 oleh Nama_Praktikan
Video 7 judul: Review Film 7 oleh Nama_Praktikan
Video 8 judul: Review Film 8 oleh Nama_Praktikan
Total Play Count: 10000

C:\Users\LENOVO\source\repos\JURNAL14_2311104060\JURNAL14_2311104060\bin\Debug\net8.0\JURNAL14_2311104060.exe (process 5428) 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 . . .|
```