

Novita Syahwa Tri Hapsari

2311104007

SE 07-01

```
const SayaTubeVideo = require("../SayaTubeVideo");

class SayaTubeUser {
  constructor(username) {
    if (typeof username !== "string" || username.length === 0) {
      throw new Error("Username harus berupa string yang tidak kosong");
    }
    this.id = Math.floor(10000 + Math.random() * 90000); // 5-digit random ID
    this.username = username;
    this.uploadedVideos = [];
  }

  AddVideo(video) {
    if (!(video instanceof SayaTubeVideo)) {
      console.log("Error: Hanya bisa menambahkan instance dari SayaTubeVideo");
      return;
    }
    this.uploadedVideos.push(video);
  }

  GetTotalVideoPlayCount() {
    return this.uploadedVideos.reduce((sum, video) => sum + video.playCount, 0);
  }

  PrintAllVideoPlaycount() {
    console.log(`User: ${this.username}`);
    this.uploadedVideos.forEach((video, index) => {
      console.log(`Video ${index + 1} judul: ${video.title}`);
    });
    console.log(`Total Play Count: ${this.GetTotalVideoPlayCount()}`);
  }
}

module.exports = SayaTubeUser;
```

```
class SayaTubeVideo {
  constructor(title) {
    if (typeof title !== "string" || title.length === 0) {
      throw new Error("Title harus berupa string yang tidak kosong");
    }
    this.id = Math.floor(10000 + Math.random() * 90000); // 5-digit random ID
    this.title = title;
    this.playCount = 0;
  }

  IncreasePlayCount(count) {
    if (typeof count !== "number" || count <= 0 || count > 1000000) {
      console.log("Error: Jumlah play count tidak valid");
      return;
    }
  }
}
```

```

        this.playCount += count;
    }

    PrintVideoDetails() {
        console.log(`ID: ${this.id}`);
        console.log(`Title: ${this.title}`);
        console.log(`Play Count: ${this.playCount}`);
    }
}

module.exports = SayaTubeVideo;

```

```

const SayaTubeUser = require("./SayaTubeUser");
const SayaTubeVideo = require("./SayaTubeVideo");

function testSayaTubeApp() {
    let user = new SayaTubeUser("Novita");
    let movieTitles = [
        "Review Film Interstellar oleh Novita",
        "Review Film Inception oleh Novita",
        "Review Film The Dark Knight oleh Novita",
        "Review Film Tenet oleh Novita",
        "Review Film Parasite oleh Novita",
        "Review Film The Shawshank Redemption oleh Novita",
        "Review Film Fight Club oleh Novita",
        "Review Film Pulp Fiction oleh Novita",
        "Review Film The Matrix oleh Novita",
        "Review Film Whiplash oleh Novita"
    ];

    movieTitles.forEach(title => {
        let video = new SayaTubeVideo(title);
        video.IncreasePlayCount(Math.floor(Math.random() * 1000)); // Menambah play count
        user.AddVideo(video);
    });

    user.PrintAllVideoPlaycount();
}

testSayaTubeApp();

```

Output:

```

User: Novita
Video 1 judul: Review Film Interstellar oleh Novita
Video 2 judul: Review Film Inception oleh Novita
Video 3 judul: Review Film The Dark Knight oleh Novita
Video 4 judul: Review Film Tenet oleh Novita
Video 5 judul: Review Film Parasite oleh Novita
Video 6 judul: Review Film The Shawshank Redemption oleh Novita
Video 7 judul: Review Film Fight Club oleh Novita
Video 8 judul: Review Film Pulp Fiction oleh Novita
Video 9 judul: Review Film The Matrix oleh Novita
Video 10 judul: Review Film Whiplash oleh Novita
Total Play Count: 3213

```

Penjelasan code:

Kode ini mengimplementasikan sistem manajemen video dengan dua kelas: SayaTubeVideo untuk merepresentasikan video dengan ID unik, judul, dan jumlah pemutaran, serta SayaTubeUser untuk mengelola daftar video yang diunggah oleh pengguna. Metode dalam kedua kelas memungkinkan penambahan video, peningkatan jumlah pemutaran, dan pencetakan detail. Pada main.js aplikasi diuji dengan membuat pengguna Novita dan menambahkan 10 video review film, lalu menampilkan daftar video beserta total pemutarannya.

```
class SayaTubeVideo {
  constructor(title) {
    if (!title || title.length > 200) {
      throw new Error("Judul video tidak boleh kosong dan maksimal 200 karakter.");
    }

    this.id = Math.floor(10000 + Math.random() * 90000); // 5-digit random ID
    this.title = title;
    this.playCount = 0;
  }

  IncreasePlayCount(count) {
    if (count < 0 || count > 25000000) {
      throw new Error("Play count harus antara 0 hingga 25.000.000.");
    }

    if (this.playCount + count > Number.MAX_SAFE_INTEGER) {
      throw new Error("Jumlah play count melebihi batas maksimum integer.");
    }

    this.playCount += count;
  }

  PrintVideoDetails() {
    console.log(`ID: ${this.id}`);
    console.log(`Title: ${this.title}`);
    console.log(`Play Count: ${this.playCount}`);
  }
}

module.exports = SayaTubeVideo;
```

```
const SayaTubeVideo = require('./SayaTubeVideo');

class SayaTubeUser {
  constructor(username) {
    if (!username || username.length > 100) {
      throw new Error("Username tidak boleh kosong dan maksimal 100 karakter.");
    }

    this.id = Math.floor(10000 + Math.random() * 90000); // 5-digit random ID
    this.username = username;
    this.uploadedVideos = [];
  }
}
```

```

    }

    AddVideo(video) {
        if (!video || !(video instanceof SayaTubeVideo)) {
            throw new Error("Video yang ditambahkan harus berupa instance dari
SayaTubeVideo.");
        }

        if (video.playCount > Number.MAX_SAFE_INTEGER) {
            throw new Error("Play count video tidak boleh melebihi batas maksimum
integer.");
        }

        this.uploadedVideos.push(video);
    }

    GetTotalVideoPlayCount() {
        return this.uploadedVideos.reduce((sum, video) => sum + video.playCount, 0);
    }

    PrintAllVideoPlaycount() {
        console.log(`User: ${this.username}`);
        let maxVideos = Math.min(8, this.uploadedVideos.length);
        for (let i = 0; i < maxVideos; i++) {
            console.log(`Video ${i + 1} judul: ${this.uploadedVideos[i].title}`);
        }
    }
}

module.exports = SayaTubeUser;

```

```

const SayaTubeUser = require('./SayaTubeUser');
const SayaTubeVideo = require('./SayaTubeVideo');

function testSayaTubeApp() {
    try {
        let user = new SayaTubeUser("Novita");
        let movieTitles = [
            "Review Film The Nun oleh Novita",
            "Review Film Inception oleh Novita",
            "Review Film The Dark Knight oleh Novita",
            "Review Film Next Door oleh Novita",
            "Review Film Parasite oleh Novita",
            "Review Film King tHe Land oleh Novita",
            "Review Film Fight Club oleh Novita",
            "Review Film Mariposa oleh Novita",
            "Review Film Lampor oleh Novita",
            "Review Film Perempuan ahli kubur oleh Novita"
        ];

        movieTitles.forEach(title => {
            try {
                let video = new SayaTubeVideo(title);
                video.IncreasePlayCount(Math.floor(Math.random() * 1000));
                user.AddVideo(video);
            } catch (error) {

```

```

        console.error("Error saat menambahkan video:", error.message);
    }
});

user.PrintAllVideoPlaycount();
} catch (error) {
    console.error("Error pada aplikasi:", error.message);
}
}

testSayaTubeApp();

```

Output:

```

User: Novita
Video 1 judul: Review Film The Nun oleh Novita
Video 2 judul: Review Film Inception oleh Novita
Video 3 judul: Review Film The Dark Knight oleh Novita
Video 4 judul: Review Film Next Door oleh Novita
Video 5 judul: Review Film Parasite oleh Novita
Video 6 judul: Review Film King tHe Land oleh Novita
Video 7 judul: Review Film Fight Club oleh Novita
Video 8 judul: Review Film Mariposa oleh Novita

```

Penjelasan code:

Kode ini mengimplementasikan sistem manajemen video dengan Design by Contract untuk memastikan validasi input, penanganan error, dan pembatasan output. Kelas SayaTubeVideo merepresentasikan video dengan ID unik, judul, dan jumlah pemutaran, serta memiliki validasi agar judul tidak melebihi 200 karakter dan play count maksimal 25 juta. Kelas SayaTubeUser mewakili pengguna dengan username valid dan daftar video yang diunggah, serta membatasi jumlah video yang ditampilkan maksimal 8. Pada main.js, program diuji dengan menambahkan 10 video ke pengguna "Novita", di mana setiap video mendapat play count acak. Blok try-catch digunakan untuk menangani error tanpa menghentikan program, memastikan keamanan data dan kelancaran eksekusi.