

2311104014

Farrel Izaz Yuwono

S1SE0701

Jurnal 06

```
import SayaTubeVideo from "./SayaTubeVideo.js";

class SayaTubeUser {
  constructor(username) {
    if (!username || username.length > 100) {
      throw new Error("Username tidak boleh null dan maksimal 100 karakter");
    }
    this.id = Math.floor(10000 + Math.random() * 90000); // Generate ID 5 digit
    this.username = username;
    this.uploadedVideos = [];
  }

  addVideo(video) {
    if (!(video instanceof SayaTubeVideo)) {
      throw new Error("Video yang ditambahkan harus instance dari SayaTubeVideo");
    }
    if (video.playCount > Number.MAX_SAFE_INTEGER) {
      throw new Error("Play count melebihi batas integer");
    }
    this.uploadedVideos.push(video);
  }

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

  printAllVideoPlayCount() {
    console.log(`User: ${this.username}`);
    this.uploadedVideos.slice(0, 8).forEach((video, index) => {
      console.log(`Video ${index + 1} judul: ${video.title} | Play Count: ${video.playCount}`);
    });
    console.log(`Total Play Count: ${this.getTotalVideoPlayCount()}`);
  }
}

export default SayaTubeUser;
```

```

class SayaTubeVideo {
  constructor(title) {
    if (!title || title.length > 200) {
      throw new Error("Judul tidak boleh null dan maksimal 200 karakter");
    }
    this.id = Math.floor(10000 + Math.random() * 90000); // Generate ID 5 digit
    this.title = title;
    this.playCount = 0;
  }

  increasePlayCount(count) {
    if (count <= 0 || count > 25000000) {
      throw new Error("Penambahan play count harus positif dan maksimal 25 juta");
    }
    try {
      let newCount = this.playCount + count;
      if (newCount > Number.MAX_SAFE_INTEGER) {
        throw new Error("Error: Play count melebihi batas integer!");
      }
      this.playCount = newCount;
    } catch (error) {
      console.error(error.message);
    }
  }

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

export default SayaTubeVideo;

```

```

ID: 96689
Title: Review Film Whiplash oleh Farel
Play Count: 1000000

ID: 17159
Title: Review Film Joker oleh Farel
Play Count: 1000000

ID: 62094
Title: Review Film Spirited Away oleh Farel
Play Count: 1000000

```

```

Username: Farel
Video 1 judul: Review Film Inception oleh Farel | Play Count: 1000000
Video 2 judul: Review Film Interstellar oleh Farel | Play Count: 1000000
Video 3 judul: Review Film The Dark Knight oleh Farel | Play Count: 1000000
Video 4 judul: Review Film Avengers: Endgame oleh Farel | Play Count: 1000000
Video 5 judul: Review Film Parasite oleh Farel | Play Count: 1000000
Video 6 judul: Review Film The Matrix oleh Farel | Play Count: 1000000
Video 7 judul: Review Film Shutter Island oleh Farel | Play Count: 1000000
Video 8 judul: Review Film Whiplash oleh Farel | Play Count: 1000000
Total Play Count: 10000000

```

Program ini merupakan implementasi dari sistem sederhana yang mensimulasikan platform berbagi video menggunakan dua kelas utama, yaitu SayaTubeVideo dan SayaTubeUser. Kelas SayaTubeVideo bertanggung jawab untuk menyimpan informasi tentang video, termasuk ID, judul, dan jumlah penayangan (play count), dengan batasan maksimal 25 juta per peningkatan. Selain itu, program juga menerapkan validasi precondition untuk memastikan bahwa judul video tidak kosong dan tidak lebih dari 200 karakter.

Sementara itu, kelas SayaTubeUser menangani daftar video yang diunggah oleh seorang pengguna dengan batasan maksimal panjang username 100 karakter. Program juga menerapkan metode untuk menambahkan video, menghitung total play count dari semua video, dan menampilkan daftar video dengan jumlah maksimal 8 video, sesuai dengan postcondition yang ditentukan.

Dalam metode utama (main()), program menguji fitur-fitur tersebut dengan membuat beberapa objek video dan menambahkannya ke dalam daftar pengguna. Untuk menguji exception handling, program mencoba melakukan penambahan play count secara berulang hingga melebihi batas integer maksimum. Jika terjadi overflow, program menangkap error dan menampilkannya tanpa menghentikan eksekusi. Dengan pendekatan ini, program dapat berjalan dengan aman dan tetap memenuhi precondition, exception handling, serta postcondition yang ditentukan.