YouTube Downloader Python Project

In this project I will create a program that downloads youtube videos to my computer using pytube3. pytube is a small, independent library (and command-line tool) for downloading YouTube videos.

1. Download and import the libraries
    a. Download pytube3 library in your system
        i. `Pip install pytube3`
    b. Next we need to import the library in our program for using its functionalities.
        i. `from pytude import YouTube`
2. Accepting input from User
    a. Ask the user for the link to the video we need to download from YouTube.
        i.
```
link = input("Enter the link: ")
yt = YouTube(link)
```
        ii. So, we have accepted input from the user and passed on the link to our YouTube class. It will help us reveal all the information about the video and also will let us download it.
3. Revealing various information about the video
    a. Now, we can play with the link and reveal all sorts of information about the video like title, number of views, ect.

b.

```
#Title of video
print("Title: ",yt.title)

#Number of views of video
print("Number of views: ",yt.views)

#Length of the video
print("Length of video: ",yt.length,"seconds")

#Description of video
print("Description: ",yt.description)

#Rating
print("Ratings: ",yt.rating)
```

4. Looking at streams available
    a. Pytube offers a way to see the available streams for the link.
        i.
```
#printing all the available streams
print(yt.streams)
```
    b. This will print all available streams both audio and video. You can filter out only audio or video streams. You can also filter out streams based on file format.
        i.
```
print(yt.streams.filter(only_video=True))
```
    c. Progressive vs Dash streams
        i. YouTube uses Dash streams for higher-quality rendering.

ii. Progressive streams are limited to 720p and have audio and video codec files while Dash has higher quality but only has video codecs.
    1. A codec compresses or decompresses media files such as songs or videos.
iii. For higher quality, we use Dash streams for video and also download an audio stream and then later merge them using any mixing tool. No mixing with progressive
iv. To get the highest resolution progressive stream available, we can just write:

```
1. Ys = yt.streams.get_highest_resolution()
```

v. So, now we have stored our preferred stream in a variable. Now, let's download it to our system.

```
1. ys.download()
```