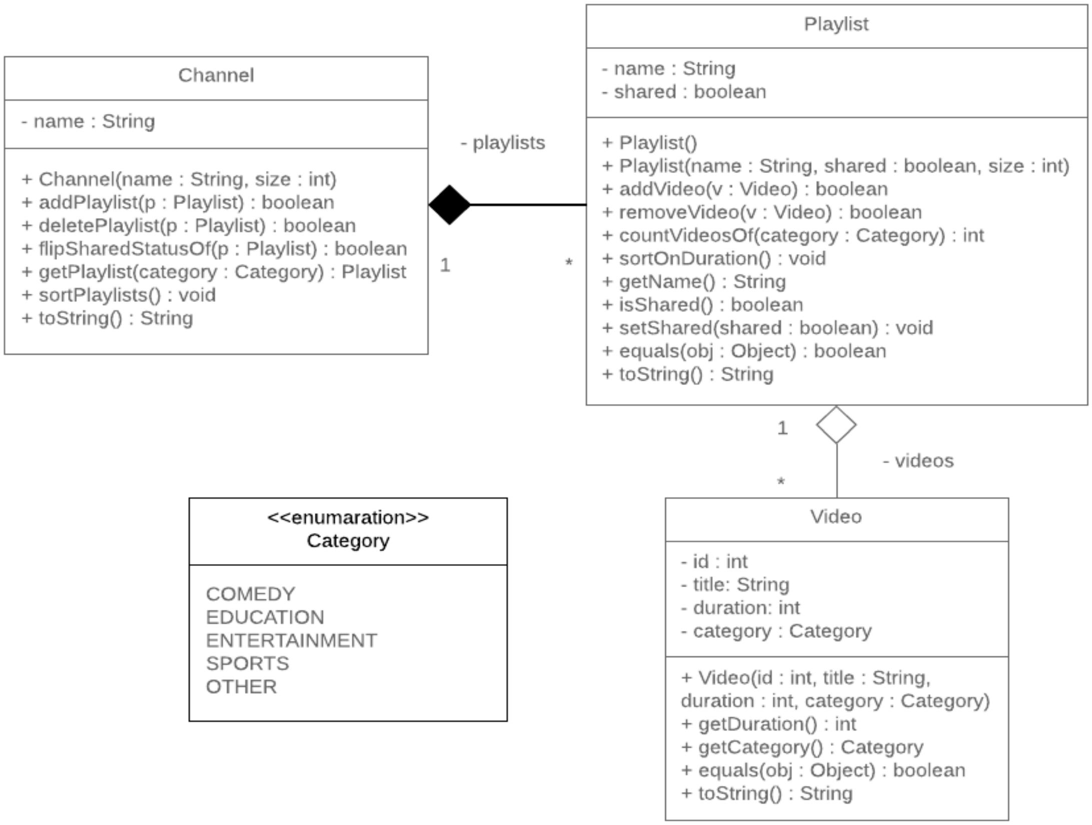
King Saud University
College of Computer and Information Sciences
Department of Computer Science
CSC113 – Computer Programming II – Composition Lab – Fall 2019

Exercise 1: Create the classes along with the functionality given in the following UML Diagram. To understand the problem, please refer to the description given after the diagram.


Channel
- name : String
+ Channel(name : String, size : int)
+ addPlaylist(p : Playlist) : boolean
+ deletePlaylist(p : Playlist) : boolean
+ flipSharedStatusOf(p : Playlist) : boolean
+ getPlaylist(category : Category) : Playlist
+ sortPlaylists() : void
+ toString() : String
- playlists
1
*
Playlist
- name : String
- shared : boolean
+ Playlist()
+ Playlist(name : String, shared : boolean, size : int)
+ addVideo(v : Video) : boolean
+ removeVideo(v : Video) : boolean
+ countVideosOf(category : Category) : int
+ sortOnDuration() : void
+ getName() : String
+ isShared() : boolean
+ setShared(shared : boolean) : void
+ equals(obj : Object) : boolean
+ toString() : String
1
- videos
*
<<enumaration>>
Category
COMEDY
EDUCATION
ENTERTAINMENT
SPORTS
OTHER
Video
- id : int
- title: String
- duration: int
- category : Category
+ Video(id : int, title : String, duration : int, category : Category)
+ getDuration() : int
+ getCategory() : Category
+ equals(obj : Object) : boolean
+ toString() : String

Video Class**:**

- Attributes:
    - ***id:*** the ID of the video
    - ***title***: the title of the video
    - ***duration***: the duration of the video in seconds
    - ***category***: the category of the video
- Methods:

  - ***Video(id:int, title:String, duration:int, category:Category):*** constructor
  - ***getDuration():*** returns the duration in seconds of the video
  - ***getCategory():*** returns the category of the video
  - ***equals(obj:Object***): compares two objects of type Video based on their ***id*** and returns the result of the equality
  - ***toString():*** this method returns a string representation of the video

Playlist Class**:**

- Attributes:
  - ***name:*** the name of the playlist
  - ***shared***: the sharing status of the playlist; true if it’s public and false if it’s private
- Methods:
  - ***Playlist()***: default constructor for an empty playlist
  - ***Playlist(name:String, shared:boolean, size:int):*** constructor
  - ***addVideo(v:Video):*** adds a video to the playlist if there’s space and the video wasn’t added before
  - ***removeVideo(v:Video):*** removes a video from the playlist if it’s there while maintaining the order of the playlist
  - ***countVideosOf(category:Category):*** returns the number of videos of a certain category currently in the playlist
  - ***sortOnDuration():*** sorts the videos in the playlist ascendingly based on the duration
  - ***getName():*** returns the name of the playlist
  - ***isShared():*** returns the shared status of the playlist
  - ***setShared(shared:boolean***): sets the value of the shared status of the playlist
  - ***equals(obj:Object***): compares two objects of type Playlist based on their ***name*** and returns the result of the equality
  - ***toString():*** this method returns a string representation of the playlist

Channel Class**:**

- Attributes:
  - ***name:*** the name of the channel
- Methods:
  - ***Channel(name:String, size:int):*** constructor
  - ***addPlaylist(p:Playlist):*** adds a playlist to the channel if there's space and the playlist wasn't added before
  - ***deletePlaylist(p:Playlist):*** removes a playlist from the channel if it's there by replacing it with the last playlist in the channel
  - ***flipSharedStatusOf(p:Playlist):*** flips the shared status of a playlist if it's there in channel
  - ***getPlaylist(category:Category***): returns the playlist in the channel having the most videos of a certain category if possible
  - ***sortPlaylists():*** sorts all videos in each playlist in the channel ascendingly based on duration
  - ***toString():*** this method returns a string representation of the channel

Exercise 2: Write a main method that tests the functionalities of the previous classes.