Enum Category

```java
public enum Category {
      COMEDY, EDUCATION, ENTERTAINMENT, SPORTS, OTHER
}
```

Class Video

```java
public class Video{
	private int id;
	private String title;
	private int duration;
	private Category category;
	public Video(int id, String title, int duration, Category category) {
		this.id = id;
		this.title = title;
		this.duration = duration;
		this.category = category;
	}
	public int getDuration() {
		return duration;
	}
	public Category getCategory() {
		return category;
	}
	public boolean equals(Video v) {
		return this.id == v.id;
	}
	public boolean equals2(Object obj) {
		if(obj == this)
			return true;
		Video temp;
		if(obj instanceof Video)
			temp = (Video) obj;
		else
			return false;
		return this.id == temp.id;
	}

	public String toString() {
		return "ID: " + id + ", Title: " + title + ", Duration: " +
duration + ", Category: " + category;
	}
}
```

Class Playlist

```java
public class Playlist {
	private String name;
	private boolean shared;
	private Video videos[];
	private int nbVid;

	public Playlist() {
	}
	public Playlist(String name, boolean shared, int size) {
		this.name = name;
		this.shared = shared;
		videos = new Video[size];
		nbVid = 0;
	}
	public Playlist(Playlist p) {
		this.name = p.name;
		this.shared = p.shared;
		this.videos = new Video[p.videos.length];
		for(int i = 0; i < p.nbVid; i++) {
			this.videos[i] = p.videos[i];
		}
		this.nbVid = p.nbVid;
	}
	private int searchVideo(Video v) {
		for(int i = 0; i < nbVid; i++) {
			if(videos[i].equals(v))
				return i;
		}
		return -1;
	}
	public boolean addVideo(Video v) {
		if(nbVid == videos.length || searchVideo(v) != -1)
			return false;
		videos[nbVid++] = v;
		return true;
	}
	public boolean removeVideo(Video v) {
		int index = searchVideo(v);
		if(index == -1)
			return false;
		for(int i = index; i < nbVid-1; i++)
			videos[i] = videos[i+1];
		videos[nbVid-1] = null;
		nbVid--;
		return true;
	}
```

```
public int countVideosOf(Category cat) {
      int counter = 0;
      for(int i = 0; i < nbVid; i++)
            if(videos[i].getCategory() == cat)
                  counter++;
      return counter;
}

public void sortOnDuration() {
      //Bubble sort

      for(int x = 0; x < nbVid; x++) {
            for(int i = 0; i < nbVid - 1 - x; i++) {
            if(videos[i].getDuration() > videos[i+1].getDuration()) {
                        //swap
                        Video temp = videos[i];
                        videos[i] = videos[i+1];
                        videos[i+1] = temp;
                  }
            }
      }
}
public boolean isShared() {
      return shared;
}
public void setShared(boolean shared) {
      this.shared = shared;
}
public String getName() {
      return name;
}
public boolean equals(Playlist p) {
      return this.name.equalsIgnoreCase(p.getName());
}
public boolean equals(Object obj) {
      Playlist temp;
      if(obj instanceof Playlist)
            temp = (Playlist) obj;
      else
            return false;
      return this.name.equalsIgnoreCase(temp.name);

}
```

```
    public String toString() {
        String play = "Name: " + name + ", Shared: ";
        if(shared)
            play += "Public";
        else
            play += "Private";
        play += ", Videos: \n";
        for(int i = 0; i < nbVid; i++)
            play += "Video number " + (i+1) + ": " +
videos[i].toString() + "\n";

        return play;
    }
}
```

Class Channel

```java
public class Channel {
      private String name;
      private Playlist playlists[];
      private int nbPly;
      public Channel(String name, int size) {
            this.name = name;
            playlists = new Playlist[size];
            nbPly = 0;
      }
      private int searchPlaylist(Playlist p) {
            for(int i = 0; i < nbPly; i++) {
                  if(playlists[i].equals(p))
                        return i;
            }
            return -1;
      }
      public boolean addPlaylist(Playlist p) {
            if(nbPly == playlists.length || searchPlaylist(p) != -1)
                  return false;
            playlists[nbPly++] = new Playlist(p);
            return true;
      }
      public boolean deletePlaylist(Playlist p) {
            int index = searchPlaylist(p);
            if(index == -1)
                  return false;
            playlists[index] = playlists[nbPly-1];
            playlists[nbPly-1] = null;
            nbPly--;
            return true;
      }
      public boolean flipSharedStatusOf(Playlist p) {
            int index = searchPlaylist(p);
            if(index == -1)
                  return false;
            playlists[index].setShared(! playlists[index].isShared());
            return true;
      }
      public Playlist getPlaylist(Category cat) {
            Playlist max = playlists[0];
            for(int i = 1; i < nbPly; i++) {
            if(max.countVideosOf(cat) < playlists[i].countVideosOf(cat))
                        max = playlists[i];
            }
            return max;
      }
      public void sortPlaylists() {
            for(int i = 0; i < nbPly; i++)
                  playlists[i].sortOnDuration();
      }

      public String toString() {
            String ch = "Name: " + name;
            ch += ", Playlists: \n";
            for(int i = 0; i < nbPly; i++)
ch += "Playlist number " + (i+1) + ": " + playlists[i].toString() + "\n";
            return ch;
      }
}
```

Class test

```
public class test {
      public static void main(String[] args) {

            Channel c1 = new Channel("KSU", 10);

            Playlist p1 = new Playlist("CCIS", true, 15);
            p1.addVideo(new Video (1, "CCIS_Seminar", 350,
      Category.EDUCATION ));
            p1.addVideo(new Video (2, "CCIS_Football", 200,
      Category.SPORTS));
            p1.addVideo(new Video (3, "CCIS_Public_Lecture", 300,
      Category.EDUCATION));
            p1.removeVideo(new Video (3,"CCIS_Public_Lecture", 300,
      Category.EDUCATION));
            c1.addPlaylist(p1);

            Playlist p2 = new Playlist("Business", false, 10);
            p2.addVideo(new Video (1, "Bussiness_Tennis_Tournament", 800,
Category.SPORTS));
            p2.addVideo(new Video (2, "Bussiness_Comedy_Club", 400,
Category.COMEDY));
            p2.addVideo(new Video (3, "Bussiness_Seminar", 200,
Category.EDUCATION));
            c1.addPlaylist(p2);

            Playlist p3 = new Playlist("Engineering", true, 20);
            p3.addVideo(new Video (1, "ENG_Field_Work", 300,
Category.OTHER));
            p3.addVideo(new Video (2, "ENG_Gaming_Competition", 700,
Category.ENTERTAINMENT));
            c1.addPlaylist(p3);

            c1.sortPlaylists();

            c1.flipSharedStatusOf(p2);

            Playlist temp = c1.getPlaylist(Category.ENTERTAINMENT);

            System.out.println("Channel Info:");
            System.out.println(c1.toString());
      System.out.println("Playlist with the most ENTERTAINMENT videos is:");
            System.out.println(temp);
      }
}
```