```
public class TV_Program {

	private double audienceRate;



	public TV_Program(double audienceRate) {

		this.audienceRate = audienceRate;

	}
public TV_Program(TV_Program p) {

		this.audienceRate = p.audienceRate;

	}

	public double getAudienceRate() {
		return audienceRate;
	}

	public void display()
	{

		System.out.println("Audience Rate: "+audienceRate);
	}



}
```

```
public interface Viewable {
	void display();
}
```

```
public abstract class TV_Channel implements Viewable{
	private int views;
	private boolean live;
	protected TV_Program arProg[];
	protected int nbProg;

	TV_Channel(int views, boolean live) {
		this.views = views;
		this.live = live;
		arProg = new TV_Program[20];
		nbProg = 0;
	}

	TV_Channel(TV_Channel tvc) {
		views = tvc.views;
		live = tvc.live;

		arProg = new TV_Program[tvc.arProg.length];
		for(int i=0; i<tvc.nbProg; i++)
			arProg[i] = new TV_Program(tvc.arProg[i]);

		nbProg = tvc. nbProg;
	}

	public int getViews() {
		return views;
	}
	public boolean isLive() {
		return live;
	}

	public abstract double CalculateRating();

	public void display() {
		System.out.println(views);
		System.out.println(live);

		for(int i=0; i<nbProg; i++)
			arProg[i].display();
	}
}



public class News extends TV_Channel {
	private int newsSegments;
	private int breakingNews;

	News(int views, boolean live,int ns, int bn ) {
		super(views, live);
		newsSegments = ns;
		breakingNews = bn;
	}

	public News(News t) {
		super(t);
```

```
            newsSegments = t.newsSegments;
            breakingNews = t.breakingNews;
      }

      public double CalculateRating() {
            double avg = 0.0;
            double rating = getViews() / breakingNews ;

            for(int i=0; i<nbProg; i++)
                  avg += arProg[i].getAudienceRate();

            if (nbProg != 0)
                  avg /= nbProg;

            return rating + avg;
      }

      public void display() {
            super.display();
            System.out.print(newsSegments + "  "+ breakingNews);
      }

      public int getBreakingNews() {
            return breakingNews;
      }
}




public class Sports extends TV_Channel {
      private int nbMatches;


      public Sports(int views, boolean live,int m ) {
            super(views, live);
            nbMatches=m;
      }

      public Sports(Sports s) {
            super(s);
            nbMatches=s.nbMatches;
      }

      public double CalculateRating() {


            return getViews()/nbMatches*1.5;
      }

      public void display() {
            super.display();
            System.out.print("No Of Matches  "+ nbMatches);
      }
```

```
	public int getNbMatches() {
		return nbMatches;
	}


}



public class TV_Group {
	private String name;

	private TV_Channel arTV[];
	private int nbTV;

	TV_Group(String name) {
		name = name;

		arTV= new TV_Channel[25];
		nbTV = 0;
	}

	public void add(TV_Channel t) {
		if(nbTV < arTV.length) {

			if(t instanceof News)
				arTV[nbTV] = new News((News) t);
			else
				arTV[nbTV] = new Sports((Sports) t);

			nbTV++;
		}
	}

	public int countLiveSport() {
		int c=0;

		for(int i=0; i < nbTV; i++)
			if(( arTV[i] instanceof Sports ) && arTV[i].isLive())
				c++;

		return c;
	}

	public void display(int n) {
		for(int i=0; i < nbTV; i++)
			if( ( arTV[i] instanceof Sports ) &&
				( ((Sports)arTV[i]).getNbMatches() > n) )
					arTV[i].display();
	}
```

```
public News[] getNews(int b) {
            News[] res = new News[nbTV];
            int k=0;

            for(int i=0; i < nbTV; i++)
                  if( ( arTV[i] instanceof News ) &&
                        ( ((News)arTV[i]).getBreakingNews() > b ) )
                        {
                              res[k] = (News) arTV[i];
                              k++;
                        }


return res;

}



            public void splitChannel(TV_Channel t1[], News[] t2) {
            int j=0, k=0;

            for(int i=0; i < nbTV; i++) {
                  if( ( arTV[i] instanceof Sports ) &&
                        ( ((Sports) arTV[i]).getNbMatches() > 20 ) ) {
                              t1[j++] = arTV[i];
                  }
                  else  {
                        if(arTV[i].getViews() > 3000) {
                              t2[k++] = (News) arTV[i];
                        }

                  }

            }
      }
}
```