**Unix Assignments**

**Contents**

[**Introduction** 1](#_Toc171528805)

[**Given Datasets:** 1](#_Toc171528806)

[**Set up data environment:** 3](#_Toc171528807)

**Data processing requirenents and testing data:** 3

### **Introduction**

A large streaming service company has a vast library of entertaining content. These contents are consumed by millions of users. User details, content and user engagement details are stored in text files in a Linux machine.

To increase the customer engagement and watch time, to set up the data environment and test the data for further processing, you need to execute Linux commands and shell scripts based on specifications given.

## **Given Datasets:**

1. **Subscribers data (UserDetails1.txt and UserDetails2.dat)**

**Schema(UserDetails2.dat): UserID#Age#Location#Subscription#WatchHistory**

**Schema(UserDetails2.dat): UserID,Age,Location,Subscription,WatchHistory**

UserID - unique subscriber id

Age - subscriber age

Location - location details

Subscription - subscription tier details

WatchHistory – ShowID;TimeStamp;Rating.

**UserID,Age,Location,Subscription,WatchHistory**

U001,25,New York,Standard,M1;2004-12-07;2|M3;2004-12-07;4|M5;2004-12-07;3

U002,32,Los Angeles,Premium,M2;2012-05-06;4|M4;2012-05-06;3|M6;2012-05-06;5

U003,40,Chicago,Standard,M7;2010-03-26;5|M1;2010-03-26;4|M9;2010-03-26;2

U004,18,Seattle,Free,M8;2002-04-07;4|M2;2002-04-07;5|M5;2002-04-07;2

U005,50,Austin,Premium,M9;2008-01-03;4|M7;2008-01-03;2|M3;2008-01-03;3

1. **Content details ( ShowDetails.txt)**

**Schema: ShowID,Genre,Actors,Director,Release\_Year,Synopsis**

ShowID - unique id of each content

Genre - content genre

Actors - details of actors in the show

Director - show director

Release year - year of release

Synapsis - description of show

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Show ID** | **Genre** | **Actors** | **Director** | **Release Year** | **Synopsis (brief summary)** |
| M1 | Comedy | Adam Sandler, Jennifer Aniston | David Miller | 2023 | Two estranged friends reunite for a wild weekend trip. |
| M2 | Action | Chris Hemsworth, Tessa Thompson | Taika Waititi | 2022 | A thrilling adventure with a team of superheroes. |
| M3 | Romance | Ryan Reynolds, Sandra Bullock | Adam Shankman | 2024 | A chance encounter leads to a second chance at love. |
| M4 | Sci-Fi | Scarlett Johansson, Chris Evans | Joe Russo | 2021 | A group of astronauts on a dangerous mission to save humanity. |
| M5 | Drama | Viola Davis, Chadwick Boseman | George C. Wolfe | 2020 | A powerful story about a family facing injustice. |

1. **Engagement data (Engagement\_data.csv)**

**Schema: UserID,ShowID,PlaybackStarted,PlaybackStopped,CompletionPercent**

UserID - unique subscriber id

ShowID - unique id of the show

PlaybackStarted – play start time

PlaybacksStopped – play stop time

CompletionPercent - % of completion

**UserID,ShowID,PlaybackStarted,PlaybackStopped,CompletionPercent**

U001,M1,12/7/2004,7/12/2005,76

U002,M2,5/6/2012,1/1/2013,63

U003,M7,3/26/2010,11/4/2010,67

U004,M8,4/7/2002,1/25/2003,33

U005,M9,1/3/2008,8/20/2008,38

## **Set up data environment:**

1. Create a function which takes User details file names as input and create file called **UserDetailsCombined.txt** by combining the data present in all input files with below schema.

**UserID,Age,Location,Subscription,ShowID,TimeStamp,Rating**

**Data cleaning:**

* Remove all empty records
* Remove all records with any missing value
* Remove the characters present in Age column along with Space character
* Remove trailing and leading spaces from Location column
* Replace consecutive white spaces with a single white space which is in between the data
* Convert all values of Subscription column in to uppercase
* Split the watch history column in to 3 columns (Show ID, Timestamp, rating). Represent the converted watch history records for every user as separate rows, sample data is shown here

**UserID,Age,Location,Subscription,ShowID,TimeStamp,Rating**

U001,25,New York,Standard,M1,2004-12-07,2

U001,25,New York,Standard,M3,2004-12-07,4

U001,25,New York,Standard,M5,2004-12-07,3

U002,32,Los Angeles,Premium,M2,2012-05-06,4

U002,32,Los Angeles,Premium,M4,2012-05-06,3

U002,32,Los Angeles,Premium,M6,2012-05-06,5

**Note: If the input file does not exist display appropriate message**

1. Create a Shell script to clean user engagement data and create a file **ued.txt** by using below data cleaning requirements.

* Remove all empty records.
* Remove all records where user id is ‘?’.
* Remove special chars present in the completion percentage.
* If CompletionPercent is empty, then replace it with 0.

## **Data processing requirements and testing data:**

1. Write a single Linux command to solve below requirements.
   1. Display count of subscribers present in each subscription tier.
   2. Display total number of subscribers between the age 20 to 30(both inclusive)
   3. Display unique genres available in alphabetical order
   4. Display User ID, Show ID and completion percentage of the shows with second highest completion percentage.
   5. Display the number of unique users who have watched each show or movie.
2. **Write a shell script to display the show ID, which are watched with less than 50% average completion rate.**
3. **Write a shell script to display the average rating for each genre.**
4. **Write a shell script to display the genre with minimum average completion rate.**

**Note: completion rate is having Space and % symbol, clean it wherever necessary**