Course: IST-659

Name: Sharat Sripada

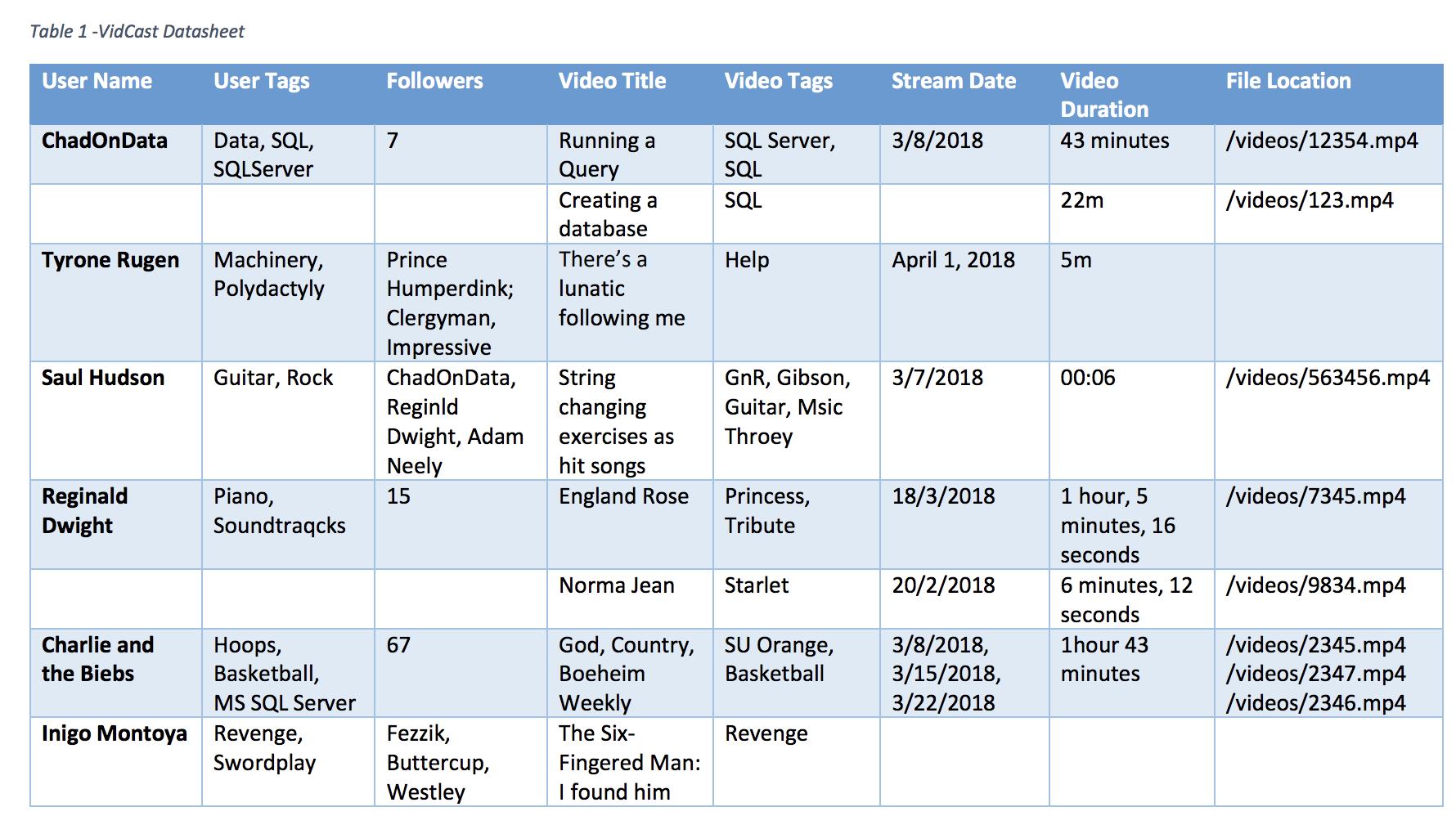
Homework #1

Due Date: 4/2/2020

Date Submitted: 4/5/2020

Topic: Lab01 – The Relational Data model

## Schema/Table Reference



## Step-1: Identify challenges to fitting Table-1 into a relational data model

1. Several attributes or columns do not have *ANY* values:
   * ‘User Name’ on row-2, row-6
   * ‘User Tags’ on row-2, row-6
   * ‘Followers’ on row-2, row-6
   * ‘Stream Date’ on row-2, row-8
   * ‘Video Duration’ on row-8
   * ‘File Location’ on row-3, row-8
2. Attribute ‘Followers’: The data in column Followers shows names in few instances and numbers in the other. An excerpt from Setup says:

“Each user will be categorized according to the number of followers they have. Users above a certain threshold of followers are invited to monetize their streams by allowing the service to periodically overlay advertising content in exchange for a share of the revenue. There will be different tiers of user status to be determined later.”

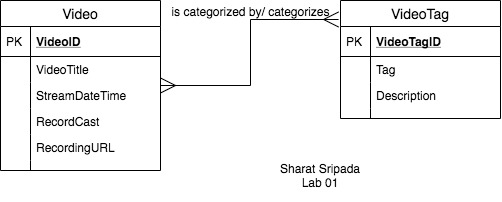
Given this, we could choose the datatype as numeric for the Followers attribute in the relational data model.

1. Attribute ‘Stream date’: The data in column Stream date is: some in <*dd/mm/yy>* format, some in *<mm/dd/yy*> while others in *<Month><Day>, <Year>* format. Defining a common format, allows us to write queries with criteria/filter as Stream date. For example: ‘select \* from <table-name> where ‘Stream date’ is after *<dd/mm/yy>*’.

**NOTE**: Also, row-7 has multiple date values. To evaluate if an append type attribute for date can be supported.

1. Attribute ‘Video Duration’: For reasons similar to that outlined in (3), the format of date in column Video Duration can be *<hr>:<min>:<sec>*
2. Redundant attributes - ‘User Tags’ and ‘Video Tags’ seem redundant with similar tag values like Guitar, SQL, SQL Server, SQLServer, BasketBall & Revenge. This could possibly be a single attribute called Tags unless there is an explicit requirement.

## Step2: Creating an Entity Relationship Diagram (ERD) using draw.io



**Entity Relationship Diagram**

Summary:

* Each VideoTag categorizes Videos (or) Each Video is categorized by VideoTag
* PK -> Primary Key for respective tables
* Arrow type indicates many to many relationships. That is, each Video can have many VideoTags and each VideoTag can be assigned to many Videos.