1. Put headings to the preprossed data files generated using MySQL to create the input files.
   * Heading for user.csv
     + user\_id (id of the user)
     + cnt (count of distinct book rated by a user)
     + attr (C/M/F to define Couple/Married People/Friends type of users)
   * Heading for rating.csv
     + user\_id (id of the user)
     + book\_id (id of the book)
     + rating (rating given to a book by a user)
   * Heading for book.csv
     + book\_id
     + authors
     + year
     + title
     + language
2. You can find the input data files in “3. TF\_IDF (Python)/input” folder.
3. Run the TF\_IDF.py program in ““3. TF\_IDF (Python)” folder.
   * This program generate the TF\_IDF value for all the books columns. The process it uses can be understandable by the “Case Study 2” example of the following link:  
     <https://www.analyticsvidhya.com/blog/2015/08/beginners-guide-learn-content-based-recommender-systems/>
   * The “TF\_IDF.py” file also has sufficient comments to understand the step by step approach.
4. The Program generates following files:
   * TF (Term Frequency)
     + TF\_title.csv (book\_id, word, TF\_title\_val)
     + TF\_authors.csv (book\_id, author, TF\_author\_val)
     + TF\_language.csv (book\_id, language, TF\_language\_val)
     + TF\_year.csv (book\_id, year, TF\_year\_val)
   * IDF (Inverse Document Frequency)
     + IDF\_title.csv (word, IDF\_val)
     + IDF\_authors.csv (author, IDF\_val)
     + IDF\_language.csv (language, IDF\_val)
     + IDF\_year.csv (year, IDF\_val)
   * UA (Users Attribute or Profile for each term)
     + UA\_title.csv (user\_id word, UA\_title\_val)
     + UA\_authors.csv (user\_id, author, UA\_author\_val)
     + UA\_language.csv (user\_id, language, UA\_language\_val)
     + UA\_year.csv (user\_id, year, UA\_year\_val)
5. You can find the output files in “3. TF\_IDF (Python)/output” folder.