**Database Project (SWE3033) (Fall 2023)**

**Homework #7 (50pts, Due date: 11/15)**

**Student ID**: 2020315798

**Student Name**: Choi Jin Woo

**Instruction:** We provided datasets and Jupyter notebook file. Data is data about film reviews and is provided in the form of JSON file. Please write your code to get correct result. **If you edit the "do not edit here" part, you will not receive any points.**

Submit two files as follows:

* ‘DBP\_Homework7\_STUDENTID.zip’
  + Code: DBP\_Homework7\_STUDENTID.ipynb
  + Document: DBP\_Homework7\_STUDENTID.pdf

1. **[20pts]** Using the given **movie\_data.json** file, write the code of index mapping, write and execute the code to create index and documents to elastic search, and report the image that include the number of hits and the data view visualized through Kibana.

Answer: Enter your **code and result** here. You must show your result, either an image or text.

Hints: Write your index mapping to work with the queries below.

|  |
| --- |
| **[5pts]** [Index mapping] |
| **[10pts]** [create index & documents] |
| **[5pts]** [elastic search result] |

1. **[30pts]** Write a query code for the given questions and report the results.

Answer: Enter your **code and result** here. You must show your result, either an image or text.

* 1. **[10pts]** What is the total **number of reviews** for films that contain the **both Action and Crime** genres?

|  |
| --- |
| [query code]  query = {  "size": 1000,  "query": {  "bool": {  "filter": [  {"match": {"genres": "Action"}},  {"match": {"genres": "Crime"}}  ]  }  }  } |
| [result] |

* 1. **[10pts]** What is the total number of **reviews** for films that contain the **Sci-fi** genre and have a **rating 4.0 or more but 5.0 or less** and report the list of the movie.

|  |
| --- |
| [query code]  query = {  "size": 1000,  "query": {  "bool": {  "must": [  {"match": {"genres": "Sci-Fi"}},  {"range": {"rating": {"gte": 4.0, "lte": 5.0}}}  ]  }  }  } |
| [result] |

* 1. **[10pts]** Find the **10 lowest rated reviews** between **2005 and 2010**, in order.

|  |
| --- |
| [query code]  query = {  "size": 10,  "query": {  "bool": {  "filter": [  {"range": {"timestamp": {"gte": "2005-01-01", "lte": "2010-12-31"}}}  ]  }  },  "sort": [  {"rating": {"order": "asc"}}  ]} |
| [result] |