**Database Project (Fall 2024)**

**Homework #6 (50pts, Due date: Nov 12)**

**Student ID**: \_\_\_\_\_2018310773\_\_\_\_

**Student Name**: \_\_\_\_임승재\_\_\_\_

**Instruction:** In this homework, we provide a dataset(airbnb-listings-newyork.json), and a jupyter notebook file(DBP\_Homework6.ipynb). You should follow the instructions in these documents carefully.

**Submission Guide:** Submit two files as follows:

-DBP\_Homwork6\_StudentID.zip

- DBP\_Homwork6\_StudentID.ipynb

- DBP\_Homwork6\_StudentID.pdf

**Data description:**

|  |  |  |
| --- | --- | --- |
| **key** | **Type** | **description** |
| \_id | Int | Airbnb’s unique identifier for the listing |
| name | String | Name of the listing |
| property\_type | String | Property type of the listing |
| room\_type | Int | Room type of the listing |
| accommodates | Float | The maximum number of people a listing can accommodate |
| beds | Int | Number of beds |
| amenities | List | Amenities included in a listing |
| price | Float | Price per night |
| text | string | Description about listing |

**1. [10pts]** A user wants to book accommodations using Airbnb. Please find and display listings that meet the following requirements for the user. **Sort the results in descending order** and display the results.

|  |  |  |  |
| --- | --- | --- | --- |
| **property\_type** | **accommodates** | **beds** | **amenities** |
| Apartment | 4 | At least 3 | Wireless Internet  Fire extinguisher  Air conditioning  TV  Dryer  Elevator in building |

**[Answer]**

Enter your code and result here. You must show your result (captured image).

|  |
| --- |
| # Load json file and save the provided documents to a collection  collection.drop()  # It may take approximately a few minutes to complete.  # ============= EDIT HERE =============  doc\_list = []  for line in open('airbnb-listings-newyork.json', 'r'):      doc\_list.append(json.loads(line))  try:      collection.insert\_many(doc\_list)  except pymongo.errrors.BulkWriteError:      pass  # =====================================  # Find and display listings that meet the requirements  # ============= EDIT HERE =============  result = collection.aggregate([      {          "$match": {              "property\_type": "Apartment",              "accommodates": 4,              "beds": {"$gte": 3},              "amenities": {                  "$all": [                      "Wireless Internet",                      "Fire extinguisher",                      "Air conditioning",                      "TV",                      "Dryer",                      "Elevator in building"                  ]              }          }      },      {          "$sort": {              "price": -1          }      }  ])  # =====================================  for doc in result:      print(doc) |

**2. [20pts]** Count all elements within the ‘amenities’ field in the collection, **sort by descending order, and display the top 10 with the highest counts**. The output format should look like this,

***[Output]***

{'\_id': 'Smoking allowed', 'value': {'count': 40}}

{'\_id': 'Indoor fireplace', 'value': {'count': 47}}

{'\_id': 'Breakfast', 'value': {'count': 54}}

{'\_id': 'Self Check-In', 'value': {'count': 57}}

{'\_id': 'Hot tub', 'value': {'count': 60}}

{'\_id': 'Gym', 'value': {'count': 76}}

Fill in the blank and capture the code and results.

**[Answer]**

|  |  |
| --- | --- |
| **The most amenity** | **Count** |
| Wireless Internet | 972 |

Enter your code and result here. You must show your result (captured image).

|  |
| --- |
| # ============= EDIT HERE =============  top\_10 = collection.aggregate([      {          "$unwind": "$amenities"      },      {          "$group": {              "\_id": "$amenities",              "count": {"$sum": 1}          }      },      {          "$sort": {              "count": -1          }      },      {          "$limit": 10      },      {          "$project": {              "\_id": 1,              "value": {                  "count": "$count"              }          }      }  ])  for i in top\_10:      print(i)  # ===================================== |

**3. [20pts]** Solve the Word Count using the ‘text’ field in the collection, **sort by descending order, and display the top 10 results with the most counts.** The output format should look like this,

***[Output]***

{'\_id': {'word': 'I'}, 'value': {'count': 112}}

{'\_id': {'word': 'love'}, 'value': {'count': 111}}

{'\_id': {'word': 'West'}, 'value': {'count': 110}}

{'\_id': {'word': 'will'}, 'value': {'count': 109}}

{'\_id': {'word': 'restaurants'}, 'value': {'count': 109}}

{'\_id': {'word': 'minutes'}, 'value': {'count': 108}}

Fill in the blank and capture the code and results.

**[Answer]**

|  |  |
| --- | --- |
| **The most frequent word** | **Count** |
| and | 1989 |

Enter your code and result here. You must show your result (captured image).

|  |
| --- |
| # ============= EDIT HERE =============  top\_words = collection.aggregate([      {          "$project": {              "words": {                  "$split": ["$text", " "]              }          }      },      {          "$unwind": "$words"      },      {          "$match": {              "words": {"$regex": "^[a-zA-Z]+$"}          }      },      {          "$group": {              "\_id": {"word": "$words"},              "count": {"$sum": 1}          }      },      {          "$sort": {              "count": -1          }      },      {          "$limit": 10      },      {          "$project": {              "\_id": {"word": "$\_id.word"},              "value": {                  "count": "$count"              }          }      }  ])  for word in top\_words:      print(word)  # ===================================== |