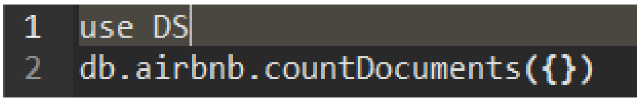
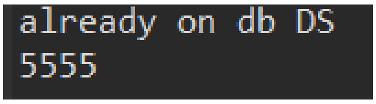
## CS/CE 457/464 - Homework Assignment 5: NoSQL

## Owais Aijaz – oa07610

1. **the count of total number of records in the collection.**



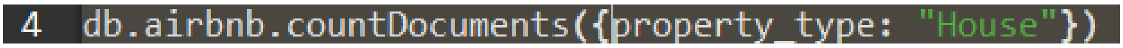
```
use DS
db.airbnb.countDocuments({})
```



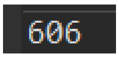
```
already on db DS
5555
```

This output shows that there are 5555 total number of records in our collection.

2. **the count of records for property_type "House".**



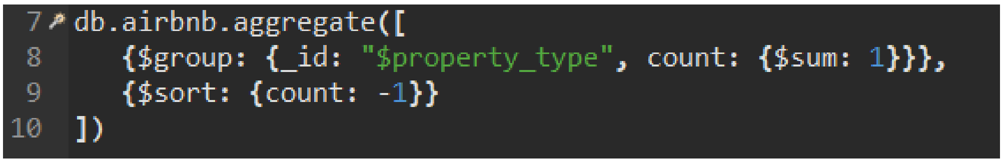
```
db.airbnb.countDocuments({property_type: "House"})
```



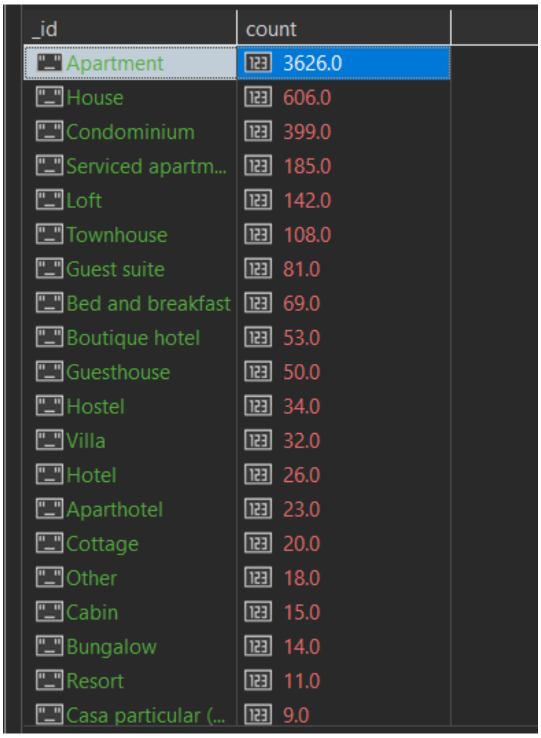
```
606
```

This means that there are 606 records/listings which are Houses (the rest can be Apartments or Condominiums, Lofts, etc).

3. **counts for each property_type in descending order of count (use group).**



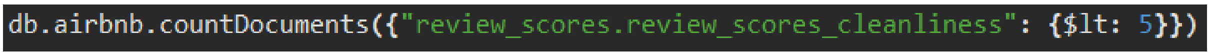
```
db.airbnb.aggregate([
    {$group: {_id: "$property_type", count: {$sum: 1}}},
    {$sort: {count: -1}}
])
```



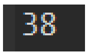
| _id | count |
|---|---|
| Apartment | 3626.0 |
| House | 606.0 |
| Condominium | 399.0 |
| Serviced apartm... | 185.0 |
| Loft | 142.0 |
| Townhouse | 108.0 |
| Guest suite | 81.0 |
| Bed and breakfast | 69.0 |
| Boutique hotel | 53.0 |
| Guesthouse | 50.0 |
| Hostel | 34.0 |
| Villa | 32.0 |
| Hotel | 26.0 |
| Aparthotel | 23.0 |
| Cottage | 20.0 |
| Other | 18.0 |
| Cabin | 15.0 |
| Bungalow | 14.0 |
| Resort | 11.0 |
| Casa particular (... | 9.0 |

The output shows 36 different types of property_types in descending order of count. So out of 5555 total listings, 3626 are Apartments, 606 are Houses, 399 are Condominium and so on.

4. **the count of records for review_scores_cleanliness less than 5.**



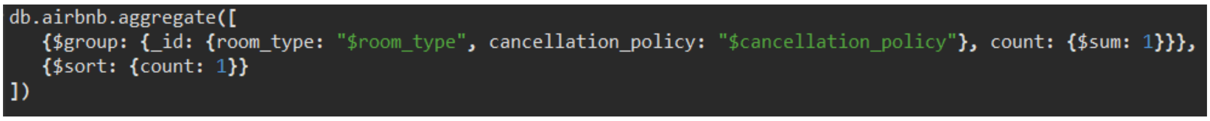
```
db.airbnb.countDocuments({"review_scores.review_scores_cleanliness": {$lt: 5}})
```



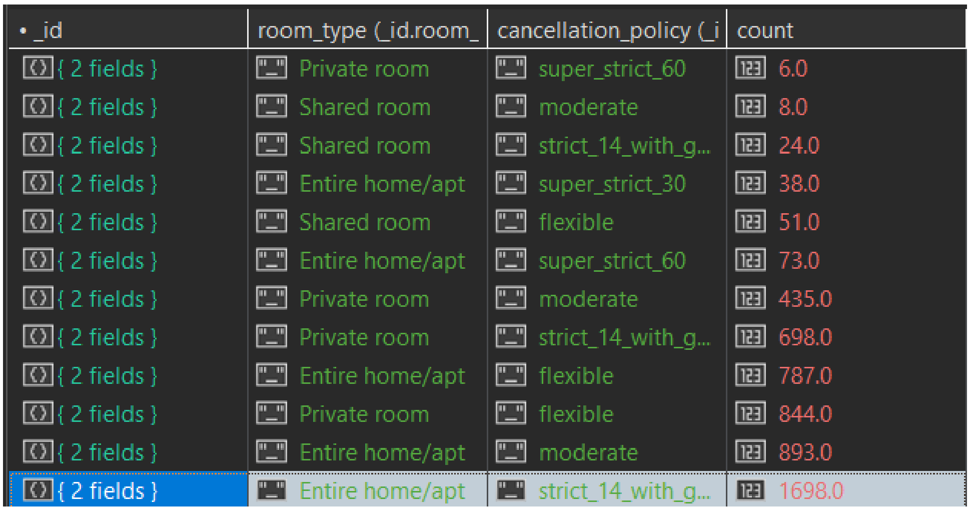
```
38
```

The output shows that there are 38 listings, which have cleanliness review score less than 5 (it means that people have reviewed that the property was not that clean).

5. **the count of records for each room_type, cancellation_policy. Order the output by count in ascending order.**



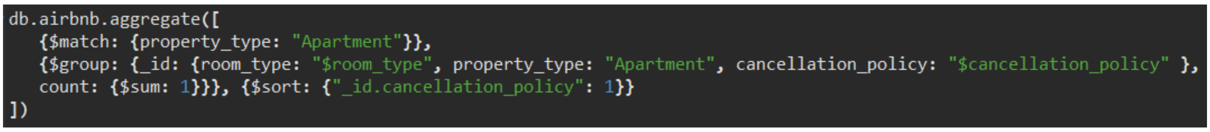
```
db.airbnb.aggregate([
   {$group: {_id: {room_type: "$room_type", cancellation_policy: "$cancellation_policy"}, count: {$sum: 1}}},
   {$sort: {count: 1}}
])
```



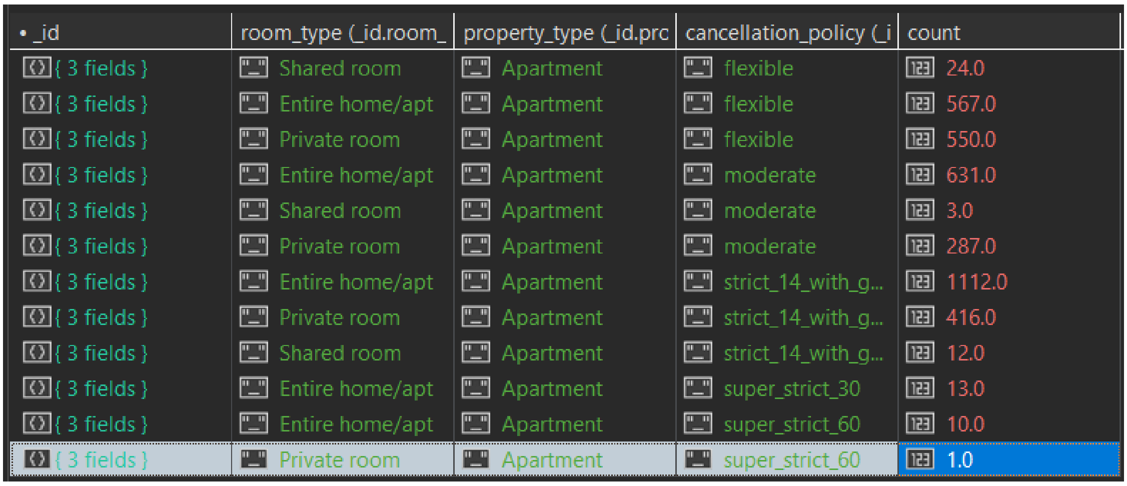
| • _id | room_type (_id.room_ | cancellation_policy (_i | count |
|---|---|---|---|
| { 2 fields } | Private room | super_strict_60 | 6.0 |
| { 2 fields } | Shared room | moderate | 8.0 |
| { 2 fields } | Shared room | strict_14_with_g... | 24.0 |
| { 2 fields } | Entire home/apt | super_strict_30 | 38.0 |
| { 2 fields } | Shared room | flexible | 51.0 |
| { 2 fields } | Entire home/apt | super_strict_60 | 73.0 |
| { 2 fields } | Private room | moderate | 435.0 |
| { 2 fields } | Private room | strict_14_with_g... | 698.0 |
| { 2 fields } | Entire home/apt | flexible | 787.0 |
| { 2 fields } | Private room | flexible | 844.0 |
| { 2 fields } | Entire home/apt | moderate | 893.0 |
| { 2 fields } | Entire home/apt | strict_14_with_g... | 1698.0 |

The output shows count of records for each room_type, cancellation_policy in ascending order. For e.g. there are 6 listings for private rooms having super_strict_60 cancellation policy. Similarly, there are 8 shared rooms with moderate cancellation policy and so on.

6. **Similar to question 5, the count of records for each room_type, cancellation_policy when property_type is "Apartment". Order the output by cancellation_policy in ascending order.**



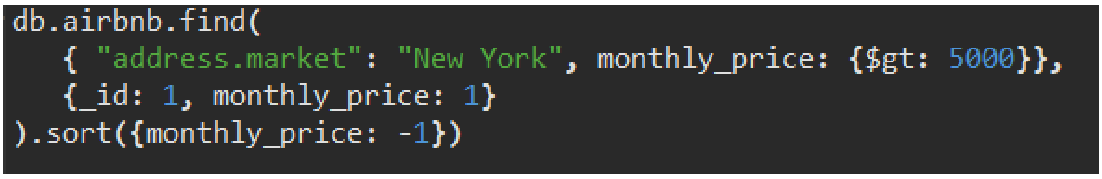
```
db.airbnb.aggregate([
   {$match: {property_type: "Apartment"}},
   {$group: {_id: {room_type: "$room_type", property_type: "Apartment", cancellation_policy: "$cancellation_policy" },
   count: {$sum: 1}}}, {$sort: {"_id.cancellation_policy": 1}}
])
```



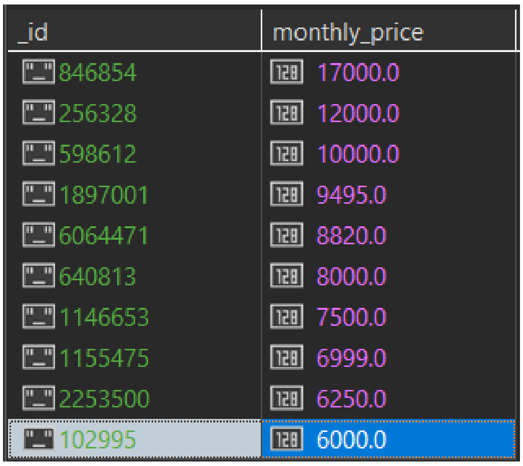
| _id | room_type (_id.room_ | property_type (_id.prc | cancellation_policy (_i | count |
|---|---|---|---|---|
| { 3 fields } | Shared room | Apartment | flexible | 24.0 |
| { 3 fields } | Entire home/apt | Apartment | flexible | 567.0 |
| { 3 fields } | Private room | Apartment | flexible | 550.0 |
| { 3 fields } | Entire home/apt | Apartment | moderate | 631.0 |
| { 3 fields } | Shared room | Apartment | moderate | 3.0 |
| { 3 fields } | Private room | Apartment | moderate | 287.0 |
| { 3 fields } | Entire home/apt | Apartment | strict_14_with_g... | 1112.0 |
| { 3 fields } | Private room | Apartment | strict_14_with_g... | 416.0 |
| { 3 fields } | Shared room | Apartment | strict_14_with_g... | 12.0 |
| { 3 fields } | Entire home/apt | Apartment | super_strict_30 | 13.0 |
| { 3 fields } | Entire home/apt | Apartment | super_strict_60 | 10.0 |
| { 3 fields } | Private room | Apartment | super_strict_60 | 1.0 |

The output shows the count of records for each room_type, cancellation_policy for Apartments. The output is sorted by cancellation policy like flexible, moderate and strict, etc. So there are 24 shared rooms in an apartment which have flexible cancellation policy and so on.

7. **Find all the records where address.market is “New York” and monthly_price is greater than 5000. Display only id and monthly_price and sort the output by monthly_price in descending order.**



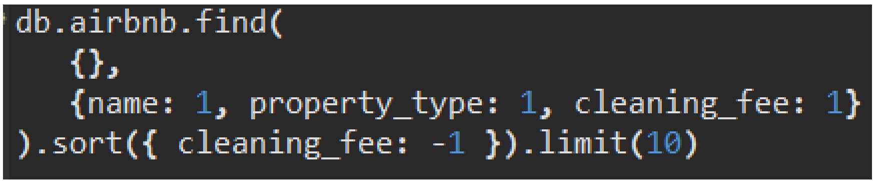
```
db.airbnb.find(
   { "address.market": "New York", monthly_price: {$gt: 5000}},
   {_id: 1, monthly_price: 1}
).sort({monthly_price: -1})
```



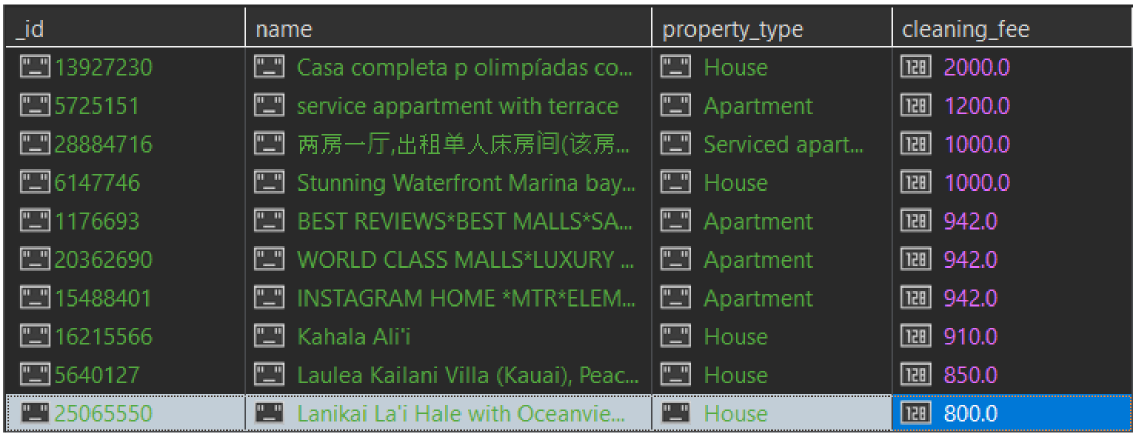
| _id | monthly_price |
|---|---|
| 846854 | 17000.0 |
| 256328 | 12000.0 |
| 598612 | 10000.0 |
| 1897001 | 9495.0 |
| 6064471 | 8820.0 |
| 640813 | 8000.0 |
| 1146653 | 7500.0 |
| 1155475 | 6999.0 |
| 2253500 | 6250.0 |
| 102995 | 6000.0 |

The output shows all listings where address.market is “New York” and monthly_price is greater than 5000, sorted by highest price first. Output shows the _id and corresponding monthly_price.

8. **Display the records with cleaning_fee in descending order. Display name, property_type and cleaning_fee. Limit records to 10 (easy)**



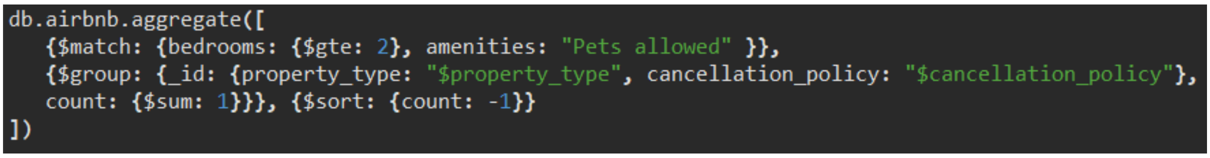
```
db.airbnb.find(
   {},
   {name: 1, property_type: 1, cleaning_fee: 1}
).sort({ cleaning_fee: -1 }).limit(10)
```



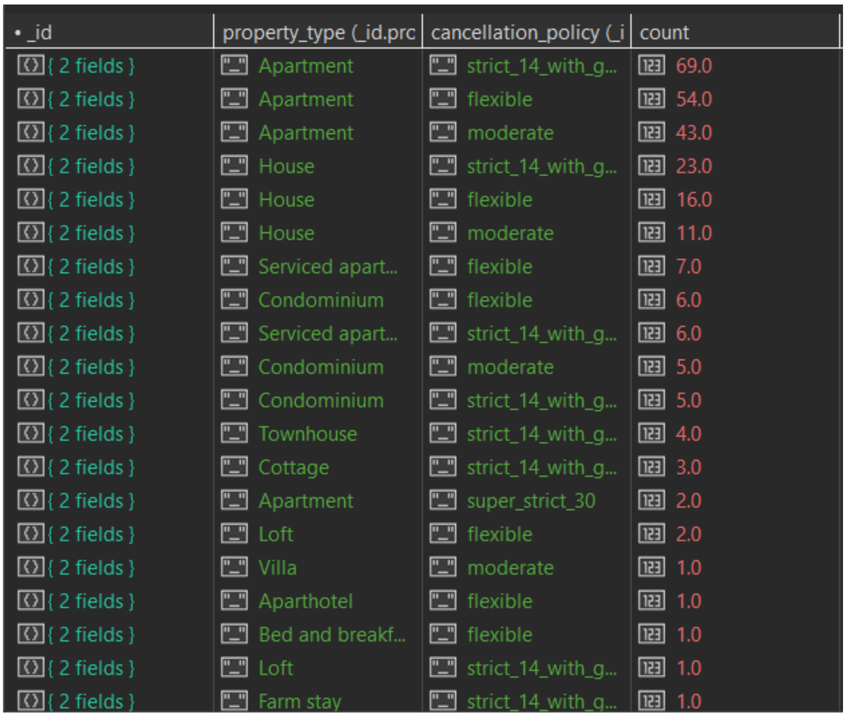
| _id | name | property_type | cleaning_fee |
|---|---|---|---|
| 13927230 | Casa completa p olimpíadas co... | House | 2000.0 |
| 5725151 | service appartment with terrace | Apartment | 1200.0 |
| 28884716 | 两房一厅,出租单人床房间(该房... | Serviced apart... | 1000.0 |
| 6147746 | Stunning Waterfront Marina bay... | House | 1000.0 |
| 1176693 | BEST REVIEWS*BEST MALLS*SA... | Apartment | 942.0 |
| 20362690 | WORLD CLASS MALLS*LUXURY ... | Apartment | 942.0 |
| 15488401 | INSTAGRAM HOME *MTR*ELEM... | Apartment | 942.0 |
| 16215566 | Kahala Ali'i | House | 910.0 |
| 5640127 | Laulea Kailani Villa (Kauai), Peac... | House | 850.0 |
| 25065550 | Lanikai La'i Hale with Oceanvie... | House | 800.0 |

These are 10 records having highest cleaning fees. For e.g. House having id 13927230 has the highest cleaning fee of 2000 USD, then second highest is 1200 and so on.

9. **Come up with your own query to show any interesting insight. Use atleast two fields for match and two fields for group.**



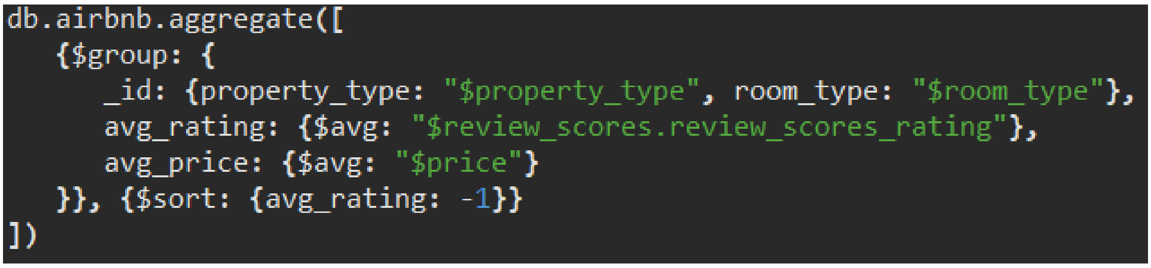
```
db.airbnb.aggregate([
   {$match: {bedrooms: {$gte: 2}, amenities: "Pets allowed" }},
   {$group: {_id: {property_type: "$property_type", cancellation_policy: "$cancellation_policy"},
   count: {$sum: 1}}}, {$sort: {count: -1}}
])
```



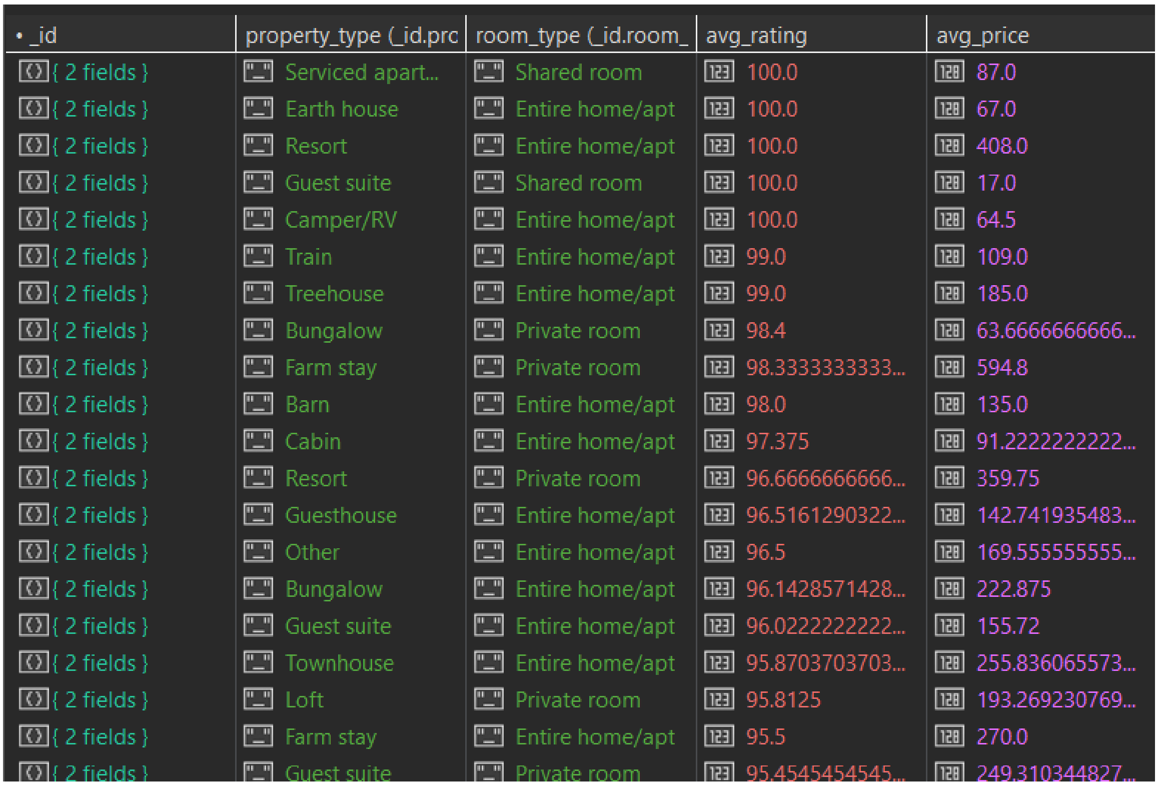
| • _id | property_type (_id.prc | cancellation_policy (_i | count |
|---|---|---|---|
| { 2 fields } | Apartment | strict_14_with_g... | 69.0 |
| { 2 fields } | Apartment | flexible | 54.0 |
| { 2 fields } | Apartment | moderate | 43.0 |
| { 2 fields } | House | strict_14_with_g... | 23.0 |
| { 2 fields } | House | flexible | 16.0 |
| { 2 fields } | House | moderate | 11.0 |
| { 2 fields } | Serviced apart... | flexible | 7.0 |
| { 2 fields } | Condominium | flexible | 6.0 |
| { 2 fields } | Serviced apart... | strict_14_with_g... | 6.0 |
| { 2 fields } | Condominium | moderate | 5.0 |
| { 2 fields } | Condominium | strict_14_with_g... | 5.0 |
| { 2 fields } | Townhouse | strict_14_with_g... | 4.0 |
| { 2 fields } | Cottage | strict_14_with_g... | 3.0 |
| { 2 fields } | Apartment | super_strict_30 | 2.0 |
| { 2 fields } | Loft | flexible | 2.0 |
| { 2 fields } | Villa | moderate | 1.0 |
| { 2 fields } | Aparthotel | flexible | 1.0 |
| { 2 fields } | Bed and breakf... | flexible | 1.0 |
| { 2 fields } | Loft | strict_14_with_g... | 1.0 |
| { 2 fields } | Farm stay | strict_14_with_g... | 1.0 |

The output shows the count of the properties with at least 2 bedrooms and that allow pets. The results are grouped by property_type and cancellation_policy and sorted in descending order with respect to count. We get 25 total results for each property_type and cancellation_policy. For e.g., there are 69 apartments with “strict_14_with_grace_ period” cancellation_policy, which have at least two bedrooms and allow pets. Similarly there are 5 Condominiums with moderate cancellation_policy which have 2 or more bedrooms, and allow pets as part of their offerings.

**10. Come up with your own query to show any interesting insight. Use atleast two fields for group.**



```
db.airbnb.aggregate([
   {$group: {
      _id: {property_type: "$property_type", room_type: "$room_type"},
      avg_rating: {$avg: "$review_scores.review_scores_rating"},
      avg_price: {$avg: "$price"}
   }}, {$sort: {avg_rating: -1}}
])
```



| • _id | property_type (_id.prc | room_type (_id.room_ | avg_rating | avg_price |
|---|---|---|---|---|
| { 2 fields } | Serviced apart... | Shared room | 100.0 | 87.0 |
| { 2 fields } | Earth house | Entire home/apt | 100.0 | 67.0 |
| { 2 fields } | Resort | Entire home/apt | 100.0 | 408.0 |
| { 2 fields } | Guest suite | Shared room | 100.0 | 17.0 |
| { 2 fields } | Camper/RV | Entire home/apt | 100.0 | 64.5 |
| { 2 fields } | Train | Entire home/apt | 99.0 | 109.0 |
| { 2 fields } | Treehouse | Entire home/apt | 99.0 | 185.0 |
| { 2 fields } | Bungalow | Private room | 98.4 | 63.6666666666... |
| { 2 fields } | Farm stay | Private room | 98.3333333333... | 594.8 |
| { 2 fields } | Barn | Entire home/apt | 98.0 | 135.0 |
| { 2 fields } | Cabin | Entire home/apt | 97.375 | 91.2222222222... |
| { 2 fields } | Resort | Private room | 96.6666666666... | 359.75 |
| { 2 fields } | Guesthouse | Entire home/apt | 96.5161290322... | 142.741935483... |
| { 2 fields } | Other | Entire home/apt | 96.5 | 169.555555555... |
| { 2 fields } | Bungalow | Entire home/apt | 96.1428571428... | 222.875 |
| { 2 fields } | Guest suite | Entire home/apt | 96.0222222222... | 155.72 |
| { 2 fields } | Townhouse | Entire home/apt | 95.8703703703... | 255.836065573... |
| { 2 fields } | Loft | Private room | 95.8125 | 193.269230769... |
| { 2 fields } | Farm stay | Entire home/apt | 95.5 | 270.0 |
| { 2 fields } | Guest suite | Private room | 95.4545454545... | 249.310344827... |

The output shows the average review_scores_rating and average price for each combination of property_type and room_type sorted by highest rating first. We get 73 results providing information on the average rating (out of 100) and average price for each combination of property and room types. For example, the top 5 highest rated listings are rated at 100 with their average price varying from 64 USD to 408 USD for different property_type and room_type combination. Similarly we can get more insights on what listings are rated higher and how their prices compare across different categories.