

Question No:
1.a

```
mongoimport --db assignment --collection data --type json --file  
C:\Users\rrsha\Documents\PythonScripts\Assignment-folder\Among_Us_data.json --jsonArray
```

```
D:\Software\Mongo\bin>mongoimport --db assignment --collection data --type json --file C:\Users\rrsha\Documents\PythonScripts\Assignment-folder\Among_Us_data.json --jsonArray  
2022-07-18T12:18:34.960+0530    connected to: mongodb://localhost/  
2022-07-18T12:18:35.127+0530    499 document(s) imported successfully. 0 document(s) failed to import.  
D:\Software\Mongo\bin>_
```

Question No:
1.b

```
db.data.find({"game":"3"}).pretty()
```

```
> use assignment
switched to db assignment
> db.data.find({"game":"3"}).pretty()
{
  "_id" : ObjectId("62d4338f6bec06a8a9c84f59"),
  "game" : "3",
  "Game_Feed" : [
    {
      "Event" : 1,
      "Map" : "Polus",
      "Outcome" : "",
      "Player/Team" : "Keaton",
      "Action" : "kills",
      "Player" : "BK",
      "Role" : "(Crew).",
      "Game Feed" : "Keaton (Impostor) kills BK (Crew).",
      "Day" : 1,
      "Votes Off Code" : "",
      "Vote ID" : "",
      "Day 1 vote" : "",
      "Crew Alive" : 8,
      "Impostors Alive" : 2,
      "Score" : "8-2"
    },
    {
      "Event" : 2,
      "Map" : "Polus"
```

Question No:
2

```
// Saving Game 3 to a new collection  
db.game3.save( db.data.find({"game":"3"}).toArray() )
```

```
> db.game3.save( db.data.find({"game":"3"}).toArray() )  
BulkWriteResult({  
  "writeErrors" : [ ],  
  "writeConcernErrors" : [ ],  
  "nInserted" : 1,  
  "nUpserted" : 0,  
  "nMatched" : 0,  
  "nModified" : 0,  
  "nRemoved" : 0,  
  "upserted" : [ ]  
})
```

Question No:
2.a

```
db.game3.find({"game":"3"}, {"Game_Feed":1, "_id":0}).pretty()
```

```
> db.game3.find({"game":"3"}, {"Game_Feed":1, "_id":0}).pretty()
{
  "Game_Feed" : [
    {
      "Event" : 1,
      "Map" : "Polus",
      "Outcome" : "",
      "Player/Team" : "Keaton",
      "Action" : "kills",
      "Player" : "BK",
      "Role" : "(Crew).",
      "Game Feed" : "Keaton (Impostor) kills BK (Crew).",
      "Day" : 1,
      "Votes Off Code" : "",
      "Vote ID" : "",
      "Day 1 vote" : "",
      "Crew Alive" : 8,
      "Impostors Alive" : 2,
      "Score" : "8-2"
    },
    {
      "Event" : 2,
      "Map" : "Polus",
      "Outcome" : "",
      "Player/Team" : "Keaton",
```

Question No:
2.b

```
db.game3.find({"Game_Feed.Outcome":{"regex":"End"}}, {"Game_Feed.$":1,"_id":0 }).pretty()
```

```
> db.game3.find({"Game_Feed.Outcome":{"regex":"End"}}, {"Game_Feed.$":1,"_id":0 }).pretty()
{
  "Game_Feed" : [
    {
      "Event" : 10,
      "Map" : "Polus",
      "Outcome" : "3-End",
      "Player/Team" : "",
      "Action" : "Crew Win - Voting",
      "Player" : "",
      "Role" : "",
      "Game Feed" : "Crew Win - Voting",
      "Day" : 4,
      "Votes Off Code" : "",
      "Vote ID" : "",
      "Day 1 vote" : "",
      "Crew Alive" : 5,
      "Impostors Alive" : 0,
      "Score" : "5-0"
    }
  ]
}
```

Question No:
2.c

```
db.game3.find({"Game_Feed.Outcome":{"$regex":"End"}}, {"_id":0, "Game_Feed.Game  
Feed.$":1}).pretty()
```

```
> db.game3.find({"Game_Feed.Outcome":{"$regex":"End"}}, {"_id":0, "Game_Feed.Game Feed.$":1}).pretty()  
{ "Game_Feed" : [ { "Game Feed" : "Crew Win - Voting" } ] }  
>
```

Question No:
2.d

```
db.game3.find({"player_data.Color":"Black"}, {"player_data.$":1,"_id":0 })
```

```
> db.game3.find({"player_data.Color":"Black"}, {"player_data.$":1,"_id":0 })
{ "player_data" : [ { "Player" : 9, "name" : "Keaton", "Role" : "(Impostor)", "Color" : "Black" } ] }
> _
```

Question No:
2.e

```
db.game3.distinct('voting_data.Vote_Event').length
```

```
> db.game3.distinct('voting_data.Vote_Event').length  
3
```


Question No:
2.f

If you were to redesign this database to make it easier to query what changes would you make to the structure? Explain your design decision

- There are a few redundant fields in the nested fields. We can eliminate them. Eg: Game feed and Voting events hold similar information.
- Remove the nested nature of the collection. We make a separate collection for voting data and player data and add references in the game feed data. By doing this the data will become less complex to deal with.
- We can make the data more granular, each document can have one event in the game. In this case the game will identifiers will need to be added in the data.

Question No:
3.a

`db.data.count()`

```
> db.data.count()  
499  
> _
```

Question No:
3.b

```
db.data.find({"Game_Feed.Game Feed":{"regex":"Crew Win"}}).count()  
db.data.find({"Game_Feed.Game Feed":{"regex":"Impostor Win"}}).count()
```

```
> db.data.find({"Game_Feed.Game Feed":{"regex":"Crew Win"}}).count()  
323  
> db.data.find({"Game_Feed.Game Feed":{"regex":"Impostor Win"}}).count()  
176  
> _
```

Question No:
3.c

```
db.data.aggregate([ { "$unwind": "$Game_Feed" },  
                    { "$match" : {"Game_Feed.Event":1}},  
                    { "$group" : { "_id": "$Game_Feed.Map","count": { "$sum":1}}}}]);
```

```
> db.data.aggregate([ { "$unwind": "$Game_Feed" },  
...                  { "$match" : {"Game_Feed.Event":1}},  
...                  { "$group" : { "_id": "$Game_Feed.Map","count": { "$sum":1}}}}]);  
{ "_id" : "The Skeld", "count" : 88 }  
{ "_id" : "MIRA HQ", "count" : 7 }  
{ "_id" : "Polus", "count" : 404 }  
>
```

Question No:
3.d

```
db.data.aggregate([ { "$unwind": "$Game_Feed" },  
                    { "$match" : {"Game_Feed.Votes Off Code":0}},  
                    { "$group" : {"_id":null, "count": { "$sum": 1 }}}])
```

```
> db.data.aggregate([ { "$unwind": "$Game_Feed" }, { $match: {"Game_Feed.Votes Off Code":0}}, { $group: {"_id":null, "count": { "$sum": 1 }}}])  
{ "_id" : null, "count" : 693 }  
>
```

Question No:
3.e

```
db.data.aggregate([ { "$unwind": "$Game_Feed" },  
                    { "$match" : {"Game_Feed.Votes Off Code":2}},  
                    { "$group" : {"_id":null, "count": { "$sum": 1 }}}])
```

```
> db.data.aggregate([ { "$unwind": "$Game_Feed" },  
... { "$match" : {"Game_Feed.Votes Off Code":2}},  
... { "$group" : {"_id":null, "count": { "$sum": 1 }}}])  
{ "_id" : null, "count" : 639 }  
>  
_
```

Question No:
3.f

In your opinion, is the game more or less hard for impostors? Justify your answer with suitable insights from the data.

- In total crew won 323 times and Imposters won 176 times.
This statistics alone leads us to believe that the game is easier for crew.
- Across 499 matches there were, 3992 crew members and 997 imposters.
For every 10 players only 2 are imposters and 8 are crew members which means the imposters have to convince more people to vote against others.

Question No:
4.a

```
db.data.distinct("player_data.name").length
```

```
> db.data.distinct("player_data.name").length
108
>
```


Question No:
4.b

```
db.data.aggregate([
  {"$unwind": "$voting_data"},
  {"$match" : {"voting_data.Vote": {"$regex":"Impostor voted off"}} },
  {"$group" : {"_id":"$voting_data.name","Count": {"$sum":1}}} ])
```

Hint: Voting data is only stored for crew members not impostors. You can cross check this.

```
> db.data.aggregate([
...   {"$unwind": "$voting_data"},
...   {"$match" : {"voting_data.Vote": {"$regex":"Impostor voted off"}} },
...   {"$group" : {"_id":"$voting_data.name","Count": {"$sum":1}}}
... ])
{ "_id" : "DaveWilliams", "Count" : 7 }
{ "_id" : "Squirrel_Loot", "Count" : 11 }
{ "_id" : "Seven", "Count" : 16 }
{ "_id" : "Trotske", "Count" : 8 }
{ "_id" : "zlubars", "Count" : 91 }
{ "_id" : "Seabats", "Count" : 11 }
{ "_id" : "Iroknight", "Count" : 12 }
{ "_id" : "julie", "Count" : 2 }
{ "_id" : "Zhoola", "Count" : 51 }
{ "_id" : "TomRoss", "Count" : 9 }
{ "_id" : "LadyAtarka", "Count" : 55 }
{ "_id" : "Pojo", "Count" : 78 }
{ "_id" : "DoubleFried", "Count" : 54 }
{ "_id" : "LSV", "Count" : 80 }
{ "_id" : "Eirik", "Count" : 20 }
{ "_id" : "TomM", "Count" : 52 }
{ "_id" : "GGards", "Count" : 1 }
{ "_id" : "Gaby", "Count" : 24 }
{ "_id" : "Ambiance", "Count" : 4 }
{ "_id" : "LegenVD", "Count" : 8 }
Type "it" for more
```

Question No:
4.c

```
db.data.aggregate([
  {"$unwind": "$voting_data"},
  {"$match" : {"voting_data.Vote": {"$regex":"Crew voted off"}} },
  {"$group" : {"_id":"$voting_data.name","Count": {"$sum":1}}} ])
```

Hint: Voting data is only stored for crew members not impostors. You can cross check this.

```
> db.data.aggregate([
...   {"$unwind": "$voting_data"},
...   {"$match" : {"voting_data.Vote": {"$regex":"Crew voted off"}} },
...   {"$group" : {"_id":"$voting_data.name","Count": {"$sum":1}}}
... ])
{ "_id" : "Kyle", "Count" : 6 }
{ "_id" : "LeeSharpe", "Count" : 3 }
{ "_id" : "Nick", "Count" : 10 }
{ "_id" : "DaveWilliams", "Count" : 7 }
{ "_id" : "Trotske", "Count" : 2 }
{ "_id" : "Ian", "Count" : 36 }
{ "_id" : "Sam", "Count" : 60 }
{ "_id" : "Stunlock", "Count" : 1 }
{ "_id" : "Emily", "Count" : 4 }
{ "_id" : "Nairbly", "Count" : 4 }
{ "_id" : "Saku", "Count" : 3 }
{ "_id" : "SamSherman", "Count" : 2 }
{ "_id" : "GrahamLRR", "Count" : 5 }
{ "_id" : "samie", "Count" : 1 }
{ "_id" : "Bim", "Count" : 2 }
{ "_id" : "PANCAKE", "Count" : 2 }
{ "_id" : "Eric", "Count" : 7 }
{ "_id" : "Keaton", "Count" : 25 }
{ "_id" : "FoxBox", "Count" : 10 }
{ "_id" : "yoinkster", "Count" : 4 }
Type "it" for more
>
```