Zad 1:

Zad2: Efekt wykonania zadania nr 2 jest widoczny w kolejnych zadaniach.

Zad3: Efekt wykonania zadania nr 3 jest widoczn3 w kolejnym zadaniu.

Zad 4:

Polecenia:

MongoDB_1   localhost:27017   AgataNowara

```
db.student.insert({name: "Agata", surname: "Nowara", presence: true, grade: null, date: new Date(), passed_subjects: ["WDI", "UNIX", "Algebra"]})
```

0.003 sec.

Inserted 1 record(s) in 3ms

MongoDB_1   localhost:27017   AgataNowara

```
db.student.insert({name: "Marek", surname: "Składak", presence: true, grade: null, date: new Date(), passed_subjects: ["WDI", "ASD", "DB"]})
```

0.002 sec.

Inserted 1 record(s) in 1ms

MongoDB_1   localhost:27017   AgataNowara

```
db.student.insert({name: "Ola", surname: "Bubik", presence: true, grade: null, date: new Date(), passed_subjects: ["Algebra", "ASD", "Python"]})
```

0.002 sec.

Inserted 1 record(s) in 1ms

Efekt:

MongoDB_1   localhost:27017   AgataNowara

```
db.getCollection('student').find({})
```

student   0.001 sec.

```
{
    "_id" : ObjectId("5e79e5557da72b9f0ee088ea"),
    "name" : "Agata",
    "surname" : "Nowara",
    "presence" : true,
    "grade" : null,
    "date" : ISODate("2020-03-24T10:47:49.101Z"),
    "passed_subjects" : [
        "WDI",
        "UNIX",
        "Algebra"
    ]
}

/* 2 */
{
    "_id" : ObjectId("5e79e63c7da72b9f0ee088eb"),
    "name" : "Marek",
    "surname" : "Składak",
    "presence" : true,
    "grade" : null,
    "date" : ISODate("2020-03-24T10:51:40.793Z"),
    "passed_subjects" : [
        "WDI",
        "ASD",
        "DB"
    ]
}

/* 3 */
{
    "_id" : ObjectId("5e79e6757da72b9f0ee088ec"),
    "name" : "Ola",
    "surname" : "Bubik",
    "presence" : true,
    "grade" : null,
    "date" : ISODate("2020-03-24T10:52:37.433Z"),
    "passed_subjects" : [
        "Algebra",
        "ASD",
        "Python"
    ]
```

Zad 5:

a)

MongoDB_1   localhost:27017   AgataNowara

```
db.business.find({stars: 5}).count()
```

0.032 sec.

5097

b) Poniżej przedstawiono jedynie fragment wyniku

MongoDB_1   localhost:27017   AgataNowara

db.business.aggregate([
    { $unwind: "$categories" },
    { $match: { categories: "Restaurants" }},
    { $group: { _id: "$city", total: { $sum: 1 } } },
    { $sort: {total: -1}}
])

business   0.146 sec.   0   50

/* 1 */
{
    "_id" : "Las Vegas",
    "total" : 3855.0
}

/* 2 */
{
    "_id" : "Phoenix",
    "total" : 2493.0
}

/* 3 */
{
    "_id" : "Edinburgh",
    "total" : 1049.0
}

/* 4 */
{
    "_id" : "Scottsdale",
    "total" : 1023.0
}

/* 5 */
{
    "_id" : "Mesa",
    "total" : 693.0
}

/* 6 */
{
    "_id" : "Madison",
    "total" : 679.0
}

/* 7 */

c)

MongoDB_1   localhost:27017   AgataNowara

db.business.aggregate([
    { $unwind: "$categories"},
    { $match: { categories: "Hotels"}},
    { $unwind: "$attributes" },
    { $match: { "attributes.Wi-Fi": "free"}},
    { $match: { stars: {$gte: 4.5}}},
    { $group: { _id: "$state", total: { $sum: 1 } } }
])

business   0.094 sec.   0   50

/* 1 */
{
    "_id" : "AZ",
    "total" : 33.0
}

/* 2 */
{
    "_id" : "NV",
    "total" : 10.0
}

/* 3 */
{
    "_id" : "WI",
    "total" : 10.0
}

/* 4 */
{
    "_id" : "EDH",
    "total" : 13.0
}

/* 5 */
{
    "_id" : "MLN",
    "total" : 1.0
}

/* 6 */
{
    "_id" : "ON",
    "total" : 2.0
}

Zad 6:

a)

```java
public long count5StarsBusinesses(){
    MongoCollection<Document> business = db.getCollection("business");
    long counter = business.countDocuments(Filters.eq("stars", 5));
    return counter;
}
```

```java
public static void main(String[] args) throws UnknownHostException {
    MongoLab mongoLab = new MongoLab();
    System.out.println(mongoLab.count5StarsBusinesses());
}
```

b) Poniżej przedstawiono jedynie fragment wyniku

```java
public void restaurantsPerCity(){
    MongoCollection <Document> business = db.getCollection("business");
    AggregateIterable aggregationResult = business.aggregate(Arrays.asList(
            Aggregates.match(Filters.eq("categories", "Restaurants")),
            Aggregates.group("$city", Accumulators.sum("count",1)),
            Aggregates.sort(Sorts.descending("count"))
    ));

    for(Object result: aggregationResult){
        System.out.println(result);
    }
}
```

```java
public static void main(String[] args) throws UnknownHostException {
    MongoLab mongoLab = new MongoLab();
    mongoLab.restaurantsPerCity();
}
```

c)

```java
public void hotelsPerState(){
    MongoCollection<Document> business = db.getCollection("business");
    AggregateIterable aggregationResult = business.aggregate(Arrays.asList(
            Aggregates.match(Filters.and(Filters.eq("categories", "Hotels"),
                    Filters.eq("attributes.Wi-Fi", "free"),
                    Filters.gte("stars", 4.5)
            )),
            Aggregates.group("$state", Accumulators.sum("count", 1))
    ));
    for(Object result: aggregationResult){
        System.out.println(result);
    }
}
```

```java
public static void main(String[] args) throws UnknownHostException {
    MongoLab mongoLab = new MongoLab();
    mongoLab.hotelsPerState();
}
```

```
Problems  @ Javadoc  Declaration  Console ⊠
<terminated> MongoLab [Java Application] /Library/Java/JavaVirtualMachines/jdk-13.jdk/Contents/Home/bin/java  (24 mar 2020, 12:42:00 – 12:42:01)
mar 24, 2020 12:42:01 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Cluster created with settings {hosts=[127.0.0.1:27017], mode=SINGLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30000 ms', maxWaitQu
mar 24, 2020 12:42:01 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Cluster description not yet available. Waiting for 30000 ms before timing out
mar 24, 2020 12:42:01 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:1, serverValue:157}] to 127.0.0.1:27017
mar 24, 2020 12:42:01 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Monitor thread successfully connected to server with description ServerDescription{address=127.0.0.1:27017, type=STANDALONE, state=CONNECTED,
mar 24, 2020 12:42:01 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:2, serverValue:158}] to 127.0.0.1:27017
Document{{_id=WI, count=10}}
Document{{_id=EDH, count=13}}
Document{{_id=MLN, count=1}}
Document{{_id=AZ, count=33}}
Document{{_id=NV, count=10}}
Document{{_id=ON, count=2}}
```

Zad 7:

```java
String bestUser(){
    MongoCollection<Document> review = db.getCollection("review");
    Object userID = review.aggregate(Arrays.asList(
            Aggregates.match(Filters.gte("stars", 4.5)),
            Aggregates.group("$user_id", Accumulators.sum("count", 1)),
            Aggregates.sort(Sorts.descending("count"))
    )).first().get("_id");
    MongoCollection<Document> user = db.getCollection("user");
    return user.find(Filters.eq("user_id", userID.toString())).first().get("name").toString();
}
```

```java
public static void main(String[] args) throws UnknownHostException {
    MongoLab mongoLab = new MongoLab();
    System.out.println(mongoLab.bestUser());
}
```

```
Problems  @ Javadoc  Declaration  Console ⊠
<terminated> MongoLab [Java Application] /Library/Java/JavaVirtualMachines/jdk-13.jdk/Contents/Home/bin/java  (24 mar 2020, 12:53:21 – 12:53:26)
mar 24, 2020 12:53:24 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Cluster created with settings {hosts=[127.0.0.1:27017], mode=SINGLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30000 ms', maxWaitQu
mar 24, 2020 12:53:24 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Cluster description not yet available. Waiting for 30000 ms before timing out
mar 24, 2020 12:53:24 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:1, serverValue:161}] to 127.0.0.1:27017
mar 24, 2020 12:53:24 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Monitor thread successfully connected to server with description ServerDescription{address=127.0.0.1:27017, type=STANDALONE, state=CONNECTED,
mar 24, 2020 12:53:24 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:2, serverValue:162}] to 127.0.0.1:27017
Rand
```

Zad 8:

```java
    List<String> countVotes(){
        MongoCollection<Document> review = db.getCollection("review");
        List<String> votesCount = new ArrayList<>();

        long funny = review.countDocuments(Filters.gte("votes.funny", 1));
        votesCount.add("Funny: " + funny);

        long cool = review.countDocuments(Filters.gte("votes.cool", 1));
        votesCount.add("Cool: " + cool);

        long useful = review.countDocuments(Filters.gte("votes.useful", 1));
        votesCount.add("Useful:" + useful);

        System.out.println(votesCount);
        return votesCount;
    }
```
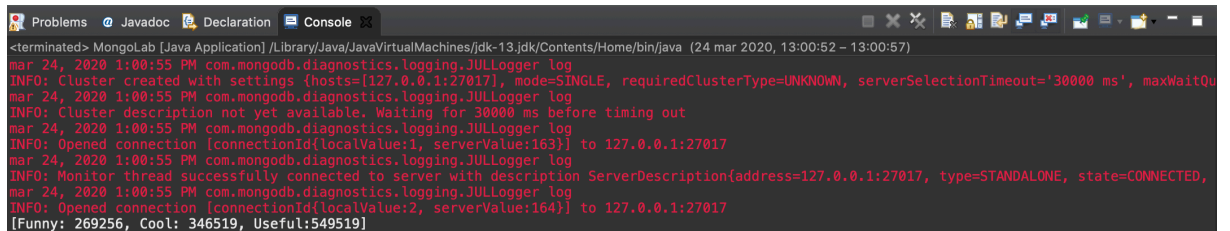
```java
    public static void main(String[] args) throws UnknownHostException {
        MongoLab mongoLab = new MongoLab();
        mongoLab.countVotes();
    }
```

Problems Javadoc Declaration Console

<terminated> MongoLab [Java Application] /Library/Java/JavaVirtualMachines/jdk-13.jdk/Contents/Home/bin/java (24 mar 2020, 13:00:52 – 13:00:57)
```
mar 24, 2020 1:00:55 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Cluster created with settings {hosts=[127.0.0.1:27017], mode=SINGLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30000 ms', maxWaitQu
mar 24, 2020 1:00:55 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Cluster description not yet available. Waiting for 30000 ms before timing out
mar 24, 2020 1:00:55 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:1, serverValue:163}] to 127.0.0.1:27017
mar 24, 2020 1:00:55 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Monitor thread successfully connected to server with description ServerDescription{address=127.0.0.1:27017, type=STANDALONE, state=CONNECTED,
mar 24, 2020 1:00:55 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:2, serverValue:164}] to 127.0.0.1:27017
[Funny: 269256, Cool: 346519, Useful:549519]
```