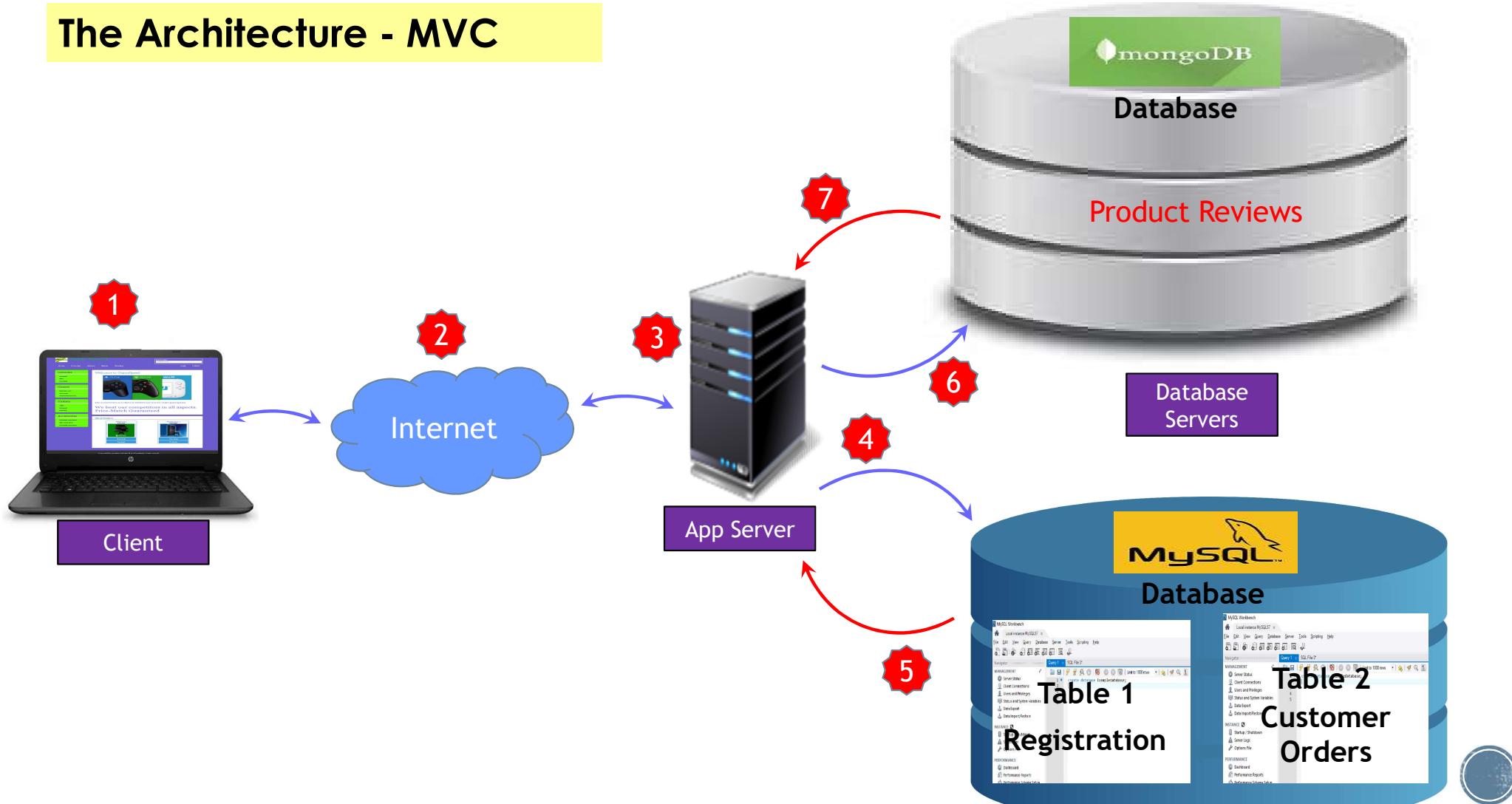


Tutorial #3 – Part A

MongoDB- Storing Reviews



The Architecture - MVC



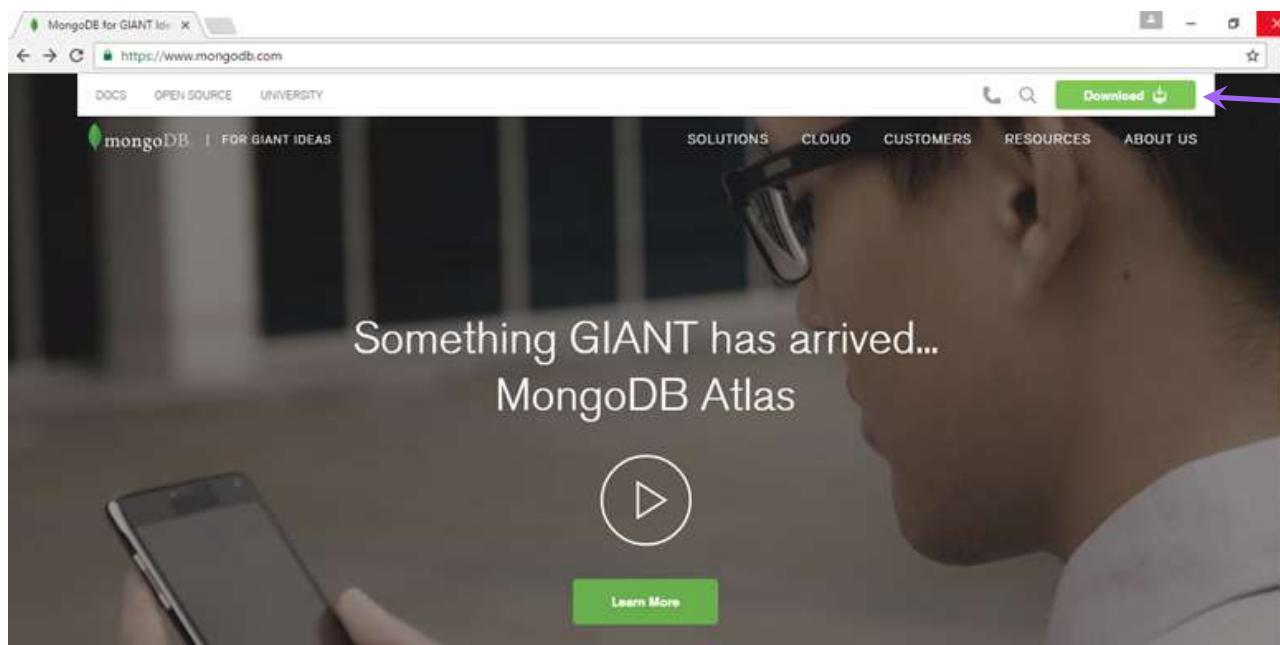
1. Mongo DB - Overview

- Mongo DB is a cross platform, document oriented database
- Mongo DB works on the concept of Collections and documents
- Terminologies:
 - Database: This is the physical container for the collections
 - Collection: Collection is a group of Mongo DB documents
 - Document: Document is a set of key - value pairs
- Advantages:
 - Schema-less: The number of fields, content and size of the document can vary from one another
 - Scalability: Mongo DB is easy to scale



2. Mongo DB - Download

- Go to <https://www.mongodb.org/> and click on the ‘Download MongoDB’ button to download Mongo DB

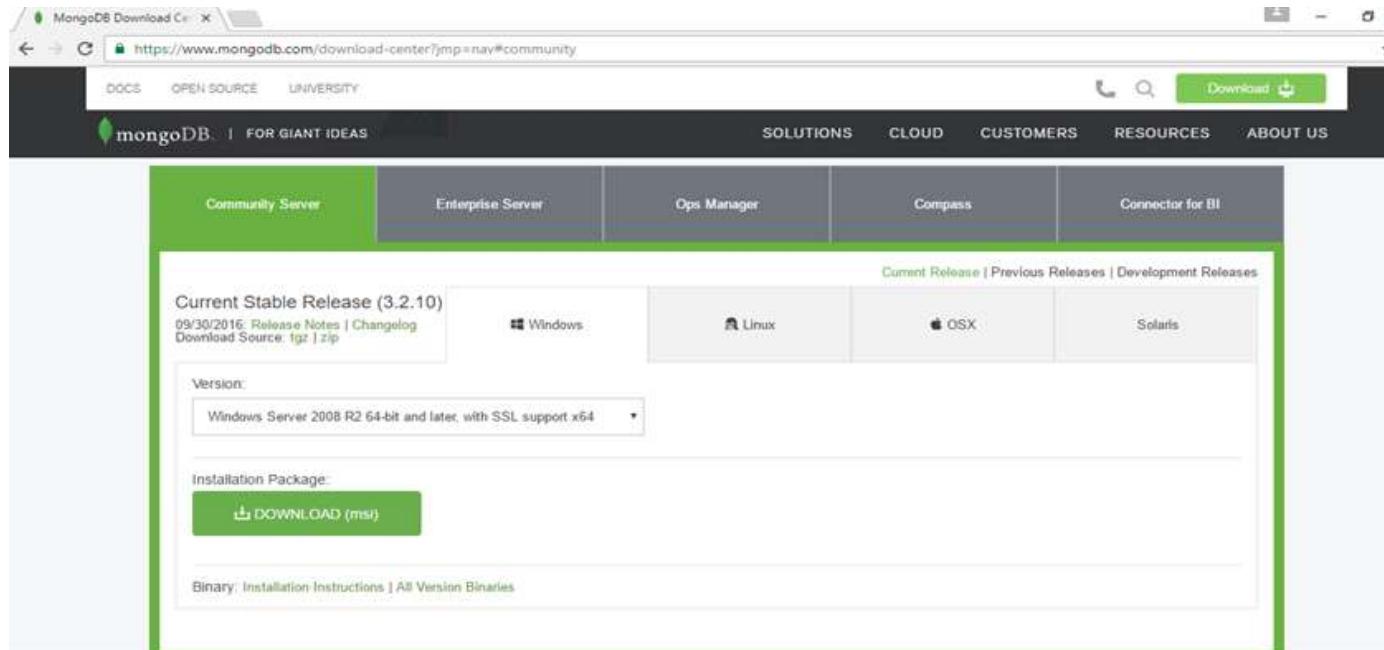


Click on the download button for downloading MongoDB



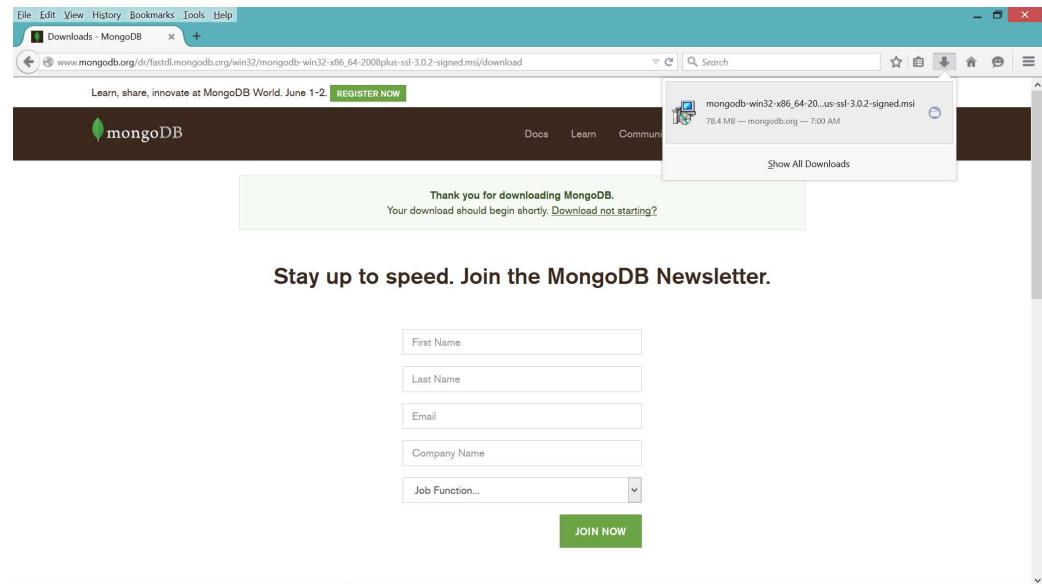
2. Mongo DB - Download

- Select the operating system as Windows and the version as ‘Windows Server 64 - bit 2008 R2 64 bit and later with SSL support x64’
- Click on the ‘Download (MSI)’ to begin the download



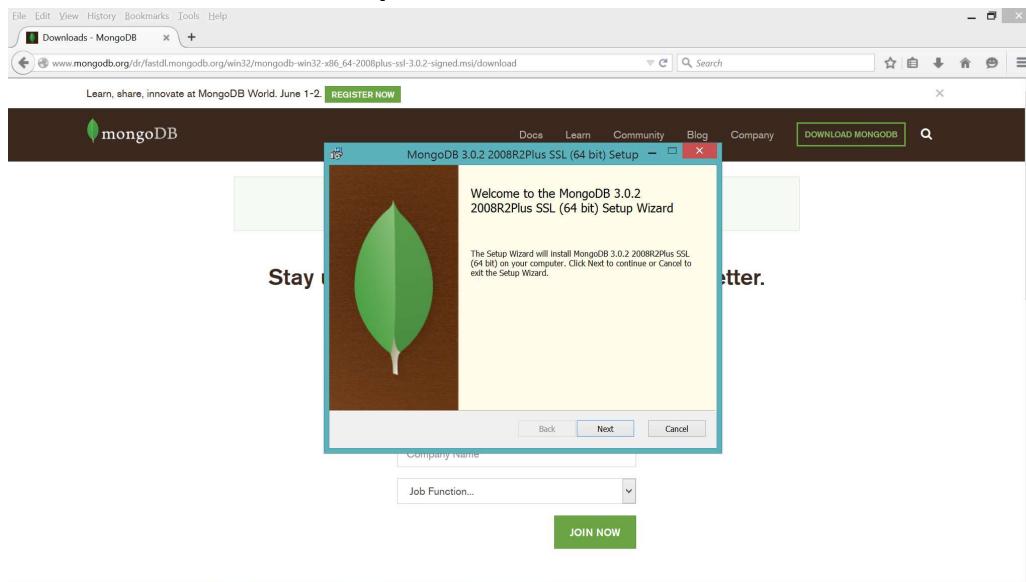
2. Mongo DB - Download

- Please note the location of the folder where MongoDB is being downloaded



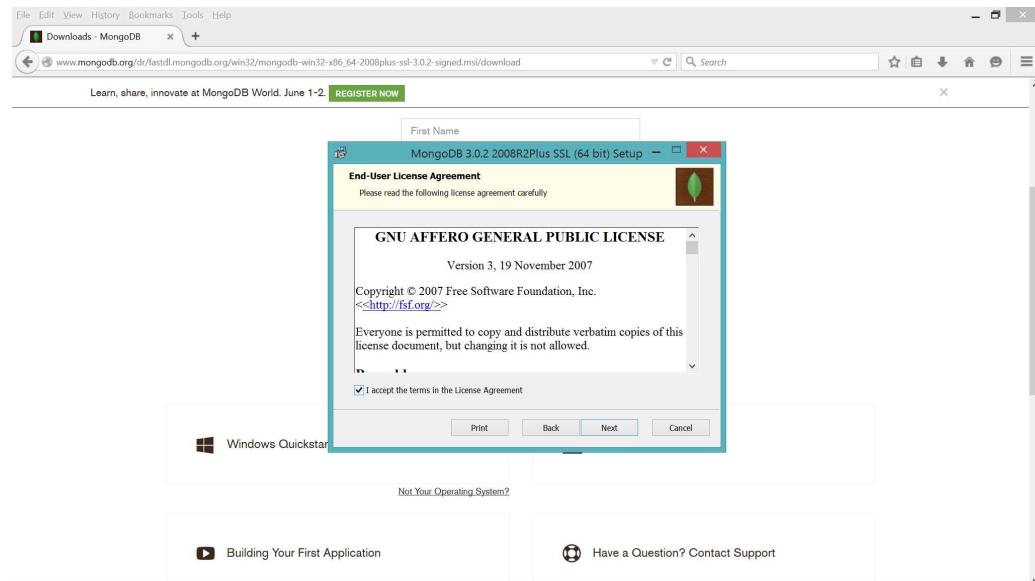
3. Mongo DB - Setup

- To start the installation, go to the folder where MongoDB has been downloaded and double click on the installation file
- This should open the MongoDB setup wizard as shown below
- Click on ‘Next’ to proceed with the installation



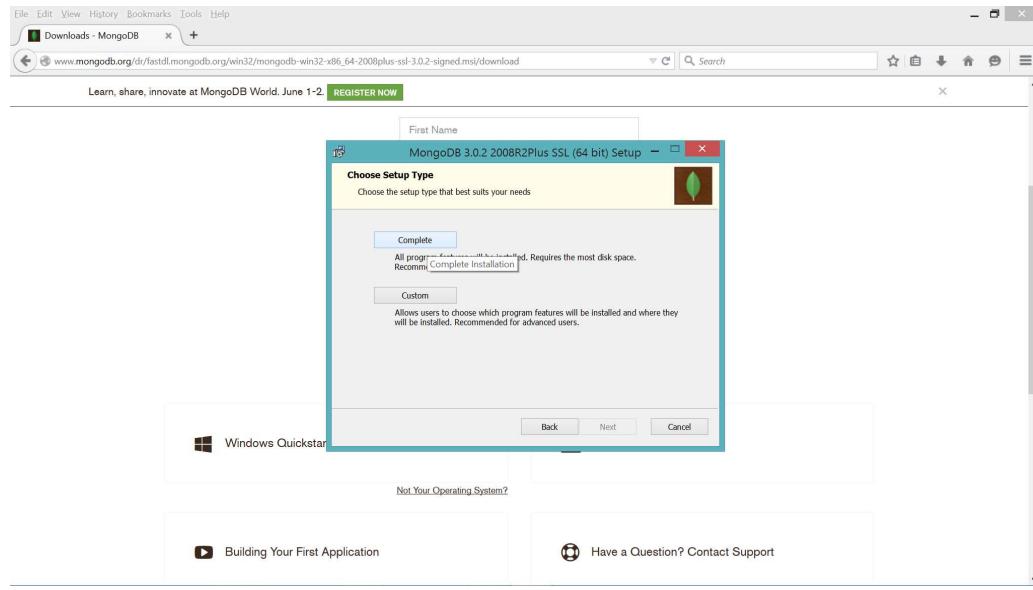
3. Mongo DB - Setup

- Accept the license agreement and proceed by clicking on ‘Next’



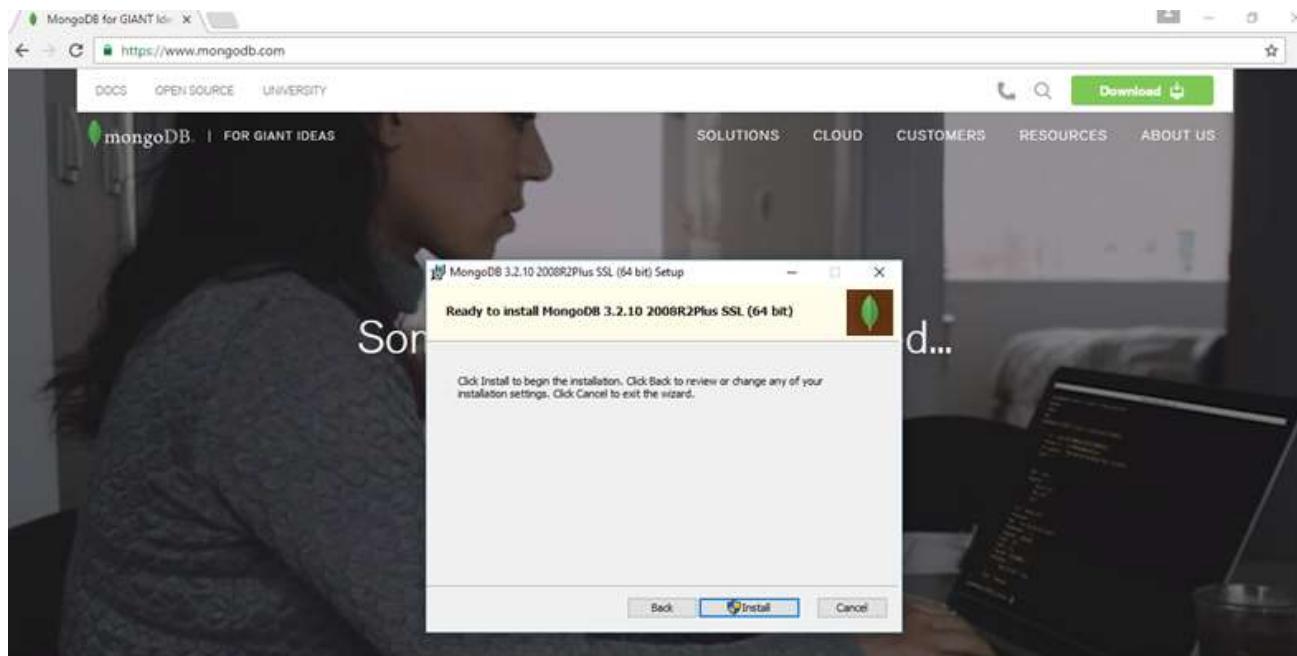
3. Mongo DB - Setup

- Select the setup type as ‘Complete’ and then click on ‘Next’
- Since we are at the beginners level with MongoDB, hence, it is recommended that you select the setup type as ‘Complete’



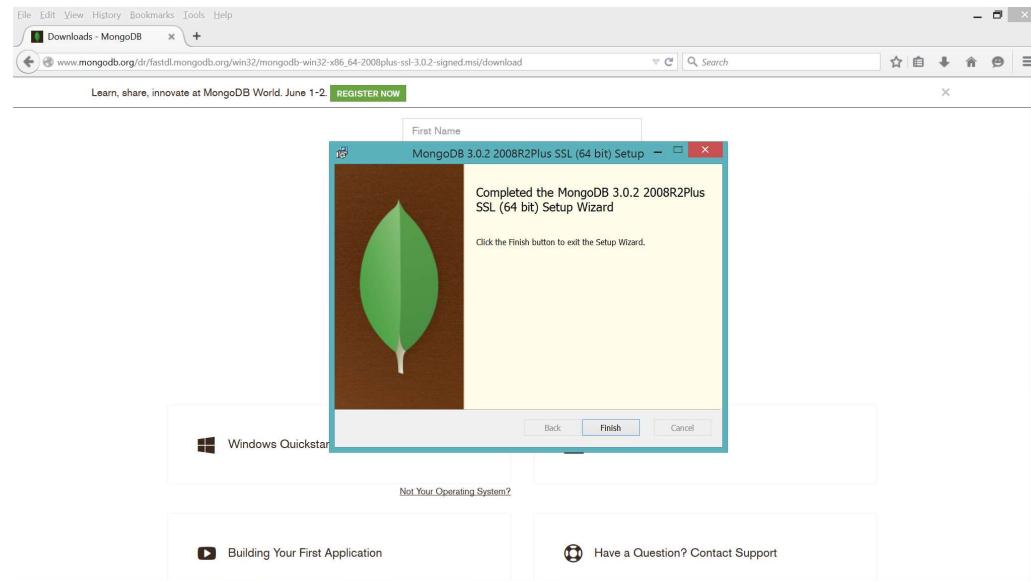
3. Mongo DB - Setup

- Click on install to install Mongo database



3. Mongo DB - Setup

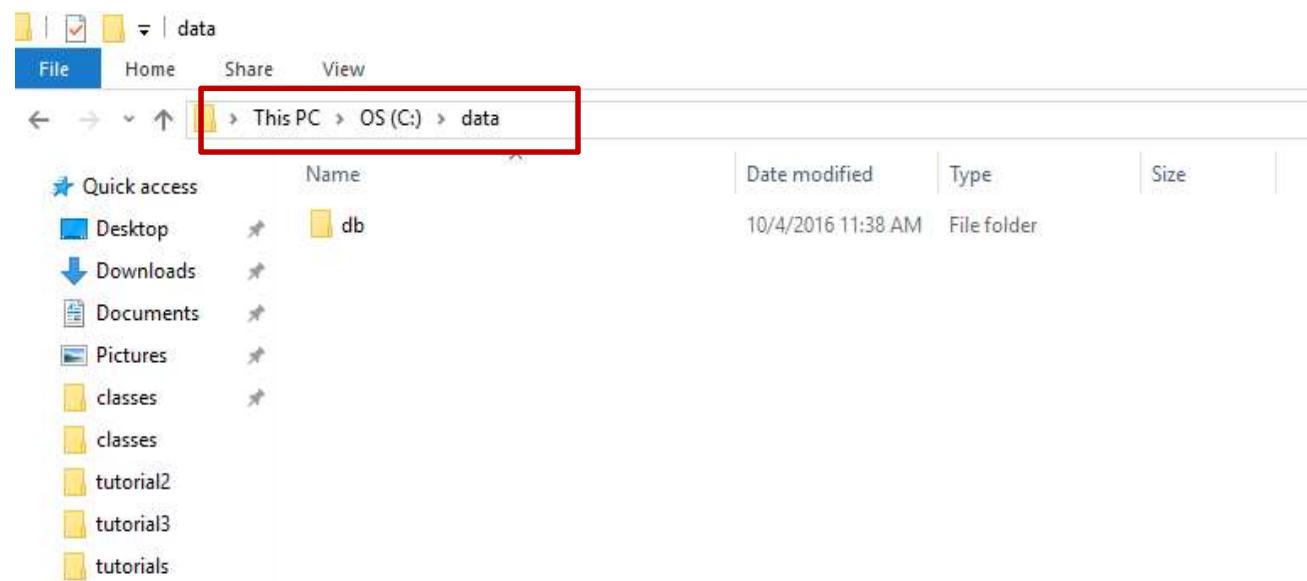
- Once the installation is complete, click on ‘Finish’ to complete the process



4. Mongo DB – Startup Instructions

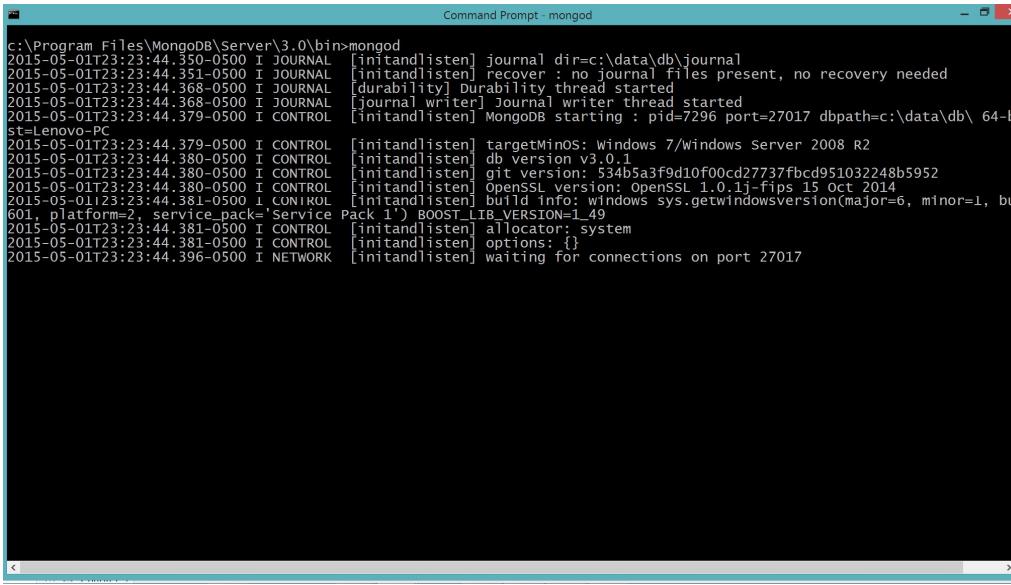
Create a data and db folder inside C drive as c:\data\db

Make sure that you directly create data\db folder inside c drive only



4. Mongo DB – Startup Instructions

- To start Mongo DB, open command prompt and enter the command ‘mongod’ .
- Mongo DB is usually installed under C:/Program Files/MongoDB
- To start Mongo DB server process , locate the “mongod.exe” stored in C:\Program Files\MongoDB\Server\3.2\bin and click it.



```
Command Prompt - mongod
c:\Program Files\MongoDB\Server\3.0\bin>mongod
2015-05-01T23:23:44.350-0500 I JOURNAL [initandlisten] journal dir=c:\data\db\journal
2015-05-01T23:23:44.351-0500 I JOURNAL [initandlisten] recover : no journal files present, no recovery needed
2015-05-01T23:23:44.368-0500 I JOURNAL [durability] durability thread started
2015-05-01T23:23:44.368-0500 I JOURNAL [journal writer] Journal writer thread started
2015-05-01T23:23:44.379-0500 I CONTROL [initandlisten] MongoDB starting : pid=7296 port=27017 dbpath=c:\data\db\ 64-bit
st=Lenovo-PC
2015-05-01T23:23:44.379-0500 I CONTROL [initandlisten] targetMinOS: Windows 7/Windows Server 2008 R2
2015-05-01T23:23:44.380-0500 I CONTROL [initandlisten] db version v3.0.1
2015-05-01T23:23:44.380-0500 I CONTROL [initandlisten] git version: 534b5a3f9d10f00cd27737fbcd951032248b5952
2015-05-01T23:23:44.380-0500 I CONTROL [initandlisten] OpenSSL version: OpenSSL 1.0.1j-fips 15 Oct 2014
2015-05-01T23:23:44.381-0500 I CONTROL [initandlisten] build info: windows sys.getwindowsversion(major=6, minor=1, bu
601, platform=2, service_pack='Service Pack 1') BOOST_LIB_VERSION=1.49
2015-05-01T23:23:44.381-0500 I CONTROL [initandlisten] allocator: system
2015-05-01T23:23:44.381-0500 I CONTROL [initandlisten] options: {}
2015-05-01T23:23:44.396-0500 I NETWORK [initandlisten] waiting for connections on port 27017
```

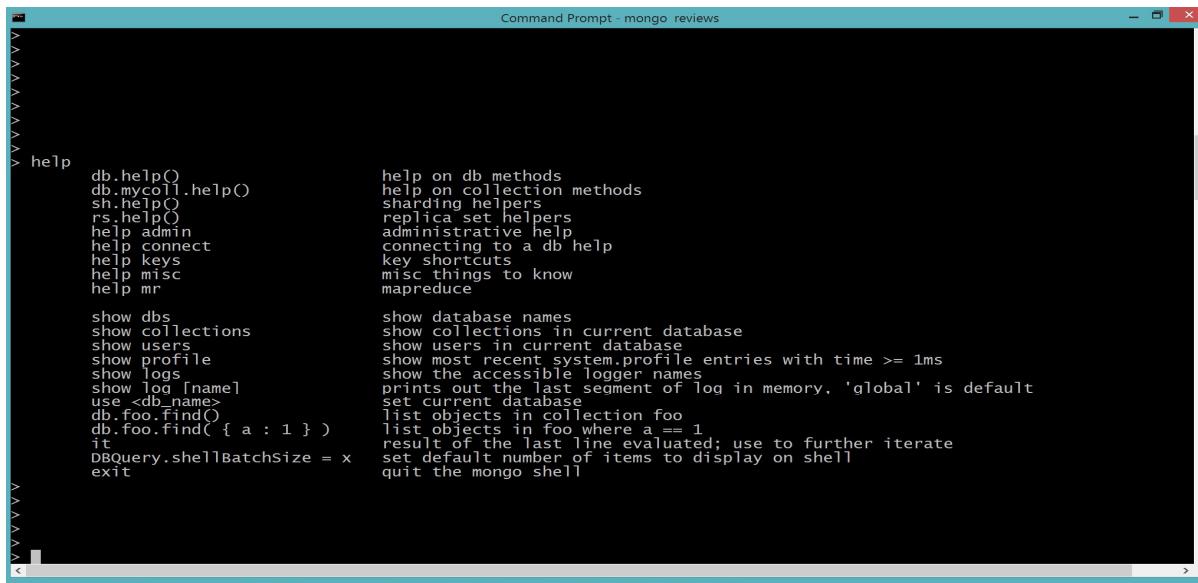
4. Mongo DB – Startup Instructions

- To start Mongo shell, open command prompt and enter the command ‘mongo’
- Mongo DB is usually installed under C:/Program Files/MongoDB
- To start Mongo shell, locate the “mongo.exe” stored in C:\Program Files\MongoDB\Server\3.2\bin and click it.



4. Mongo DB – Help command & Documentation

- The ‘Help’ command is a very handy command and can be used to check various commands available with Mongo DB
- To learn more on MongoDB Commands , visit: <https://docs.mongodb.com/manual/reference/mongo-shell/>



```
Command Prompt - mongo reviews
>
>
>
>
> help
  db.help()
  db.mycol1.help()
  sh.help()
  rs.help()
  help admin
  help connect
  help keys
  help misc
  help mr

  show dbs
  show collections
  show users
  show profile
  show logs
  show log [name]
  use <db_name>
  db.foo.find()
  db.foo.find( { a : 1 } )
  it
  DBQuery.shellBatchSize = x
  exit

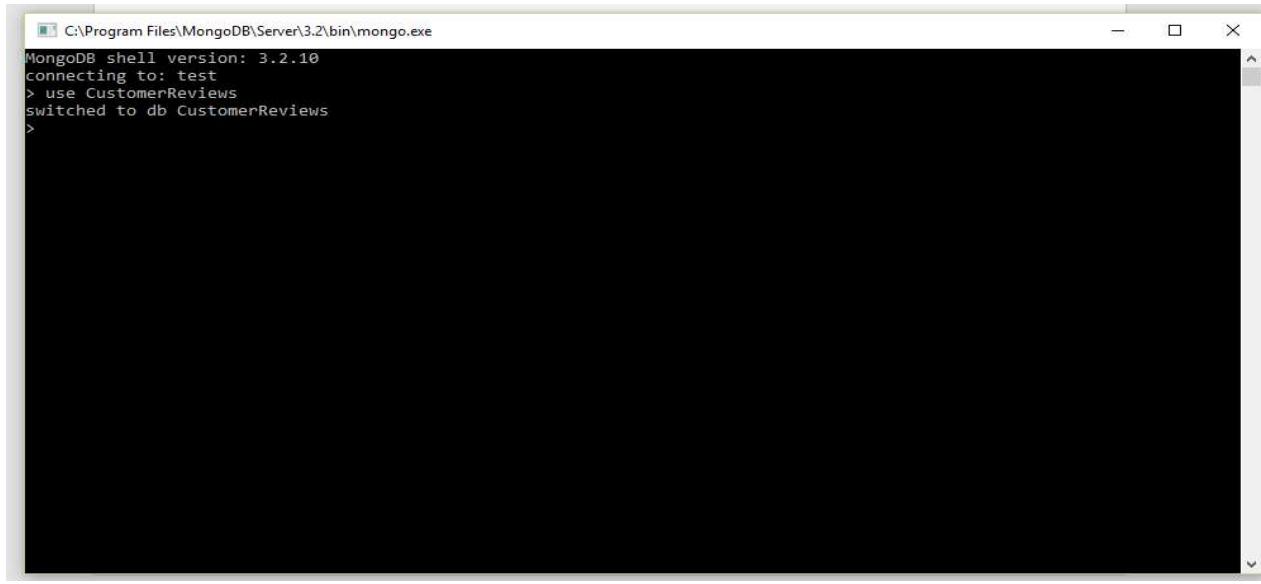
  help on db methods
  help on collection methods
  sharding helpers
  replica set helpers
  administrative help,
  connecting to a db help
  key shortcuts
  misc things to know
  mapreduce

  show database names
  show collections in current database
  show users in current database
  show most recent system.profile entries with time >= 1ms
  show the accessible logger names
  prints out the last segment of log in memory. 'global' is default
  set current database
  list objects in collection foo
  list objects in foo where a == 1
  result of the last line evaluated; use to further iterate
  set default number of items to display on shell
  quit the mongo shell
```



4. Mongo DB – Use a database

- In order to use a database, you must select it first
- To select a database along with the startup, use the command ‘use databasename’
- Example, to select the ‘CustomerReviews’ database, the command is ‘use CustomerReviews’
- You can then check the db you are in by typing db command



The screenshot shows a Windows command-line interface window titled 'C:\Program Files\MongoDB\Server\3.2\bin\mongo.exe'. The window displays the MongoDB shell version 3.2.10. A user has typed the command 'use CustomerReviews' and pressed enter. The response shows that the database has been switched to 'CustomerReviews'. The rest of the window is blacked out.

```
C:\Program Files\MongoDB\Server\3.2\bin\mongo.exe
MongoDB shell version: 3.2.10
connecting to: test
> use CustomerReviews
switched to db CustomerReviews
>
```



4. Mongo DB – Create Collections

- You can manually create collection or automatic by running your java program
- To create a collection manually type db.createCollection(collectionname)



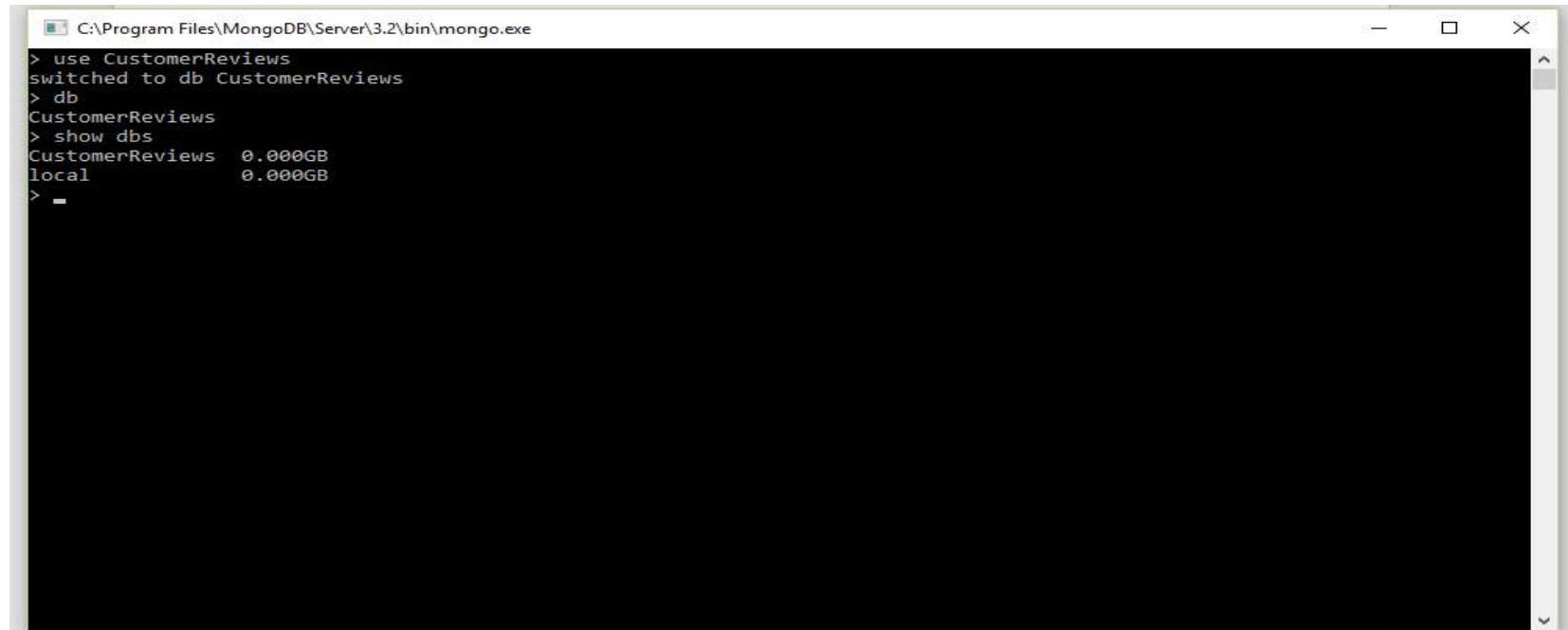
The screenshot shows a terminal window titled "C:\Program Files\MongoDB\Server\3.2\bin\mongo.exe". The window contains the following MongoDB shell commands:

```
>
> db.createCollection("myReviews")
{ "ok" : 1 }
>
```



4. Mongo DB – Display list of available databases

- To check the databases that exist, use the command ‘show dbs’
- This will show the list of available databases



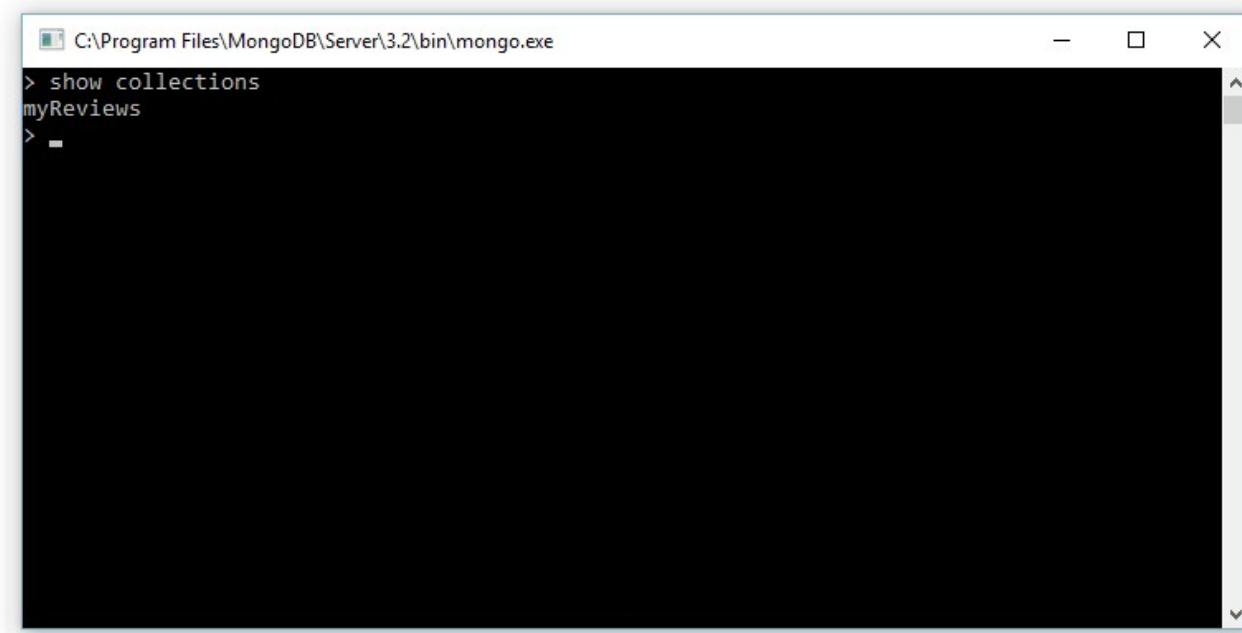
A screenshot of a terminal window titled "C:\Program Files\MongoDB\Server\3.2\bin\mongo.exe". The window contains the following text:

```
> use CustomerReviews
switched to db CustomerReviews
> db
CustomerReviews
> show dbs
CustomerReviews  0.000GB
local            0.000GB
> -
```



4. Mongo DB – Show collections

- Use the command ‘ show collections’ to view the list of available collections in the selected database



A screenshot of a terminal window titled "C:\Program Files\MongoDB\Server\3.2\bin\mongo.exe". The window contains the following text:

```
> show collections
myReviews
> -
```



4. Mongo DB – Query data

- In order to query data, use the command ‘db.COLLECTION_NAME.find()’
- The find() queries the data available in the selected collection.
- Example, to query the ‘myReviews’ collection we use the command ‘db.myReviews.find()’



The screenshot shows a terminal window titled 'C:\Program Files\MongoDB\Server\3.2\bin\mongo.exe'. The window contains the following mongo shell session:

```
>
>
> db.myReviews.find()
{ "_id" : ObjectId("57f3e06441e5be1e543b3e0d"), "title" : "myReviews", "use
rName" : "customer1", "productName" : "xbox360", "productType" : "consoles"
, "productMaker" : "microsoft", "reviewRating" : "5", "reviewDate" : "2016-
09-13", "reviewText" : " Amazing Game to Play" }
> -
```



5. Compile and run

- You need to include all the JAR files before you compile your Java program which imports external libraries (Such as Servlets, MongoDB in this tutorial)
- To download MongoDB Connector jar go to
<https://mvnrepository.com/artifact/org.mongodb/mongo-java-driver/3.1.0>
- click on the Download(JAR) link to download the MongoDB connector.
- To include these external JAR files, make the changes to the ‘CLASSPATH’ variable in your ‘env-setup-for-tomcat_backup.bat’ file
- Locate and copy the location of the JAR files on your computer and edit the ‘CLASSPATH’ variable accordingly
- **NOTE:** Make sure you have the necessary JAR files on your computer



5. Compile and run

- Download mongo-java-driver-3.1.0.jar

The screenshot shows the MVNRepository website at <https://mvnrepository.com/artifact/org.mongodb/mongo-java-driver/3.1.0>. The page displays the MongoDB Java Driver artifact version 3.1.0. A yellow box highlights a note about a new version (3.4.0-rc1). A purple callout box with the text "Click on the download button for downloading Mongo java Driver" points to the "Download (JAR) (987 KB)" button, which is also highlighted with a red box and has a blue arrow pointing to it.

Indexed Artifacts (4.78M)

Popular Categories

Aspect Oriented
Actor Frameworks
Application Metrics
Build Tools
Bytecode Libraries
Command Line Parsers
Cache Implementations
Cloud Computing
Code Analyzers
Collections

Search for groups, artifacts, categories

Home » org.mongodb » mongo-java-driver » 3.1.0

Note: There is a new version for this artifact

New Version 3.4.0-rc1

MongoDB Java Driver » 3.1.0

The MongoDB Java Driver uber-artifact, containing mongodb-driver, mongodb-driver-core, and bson

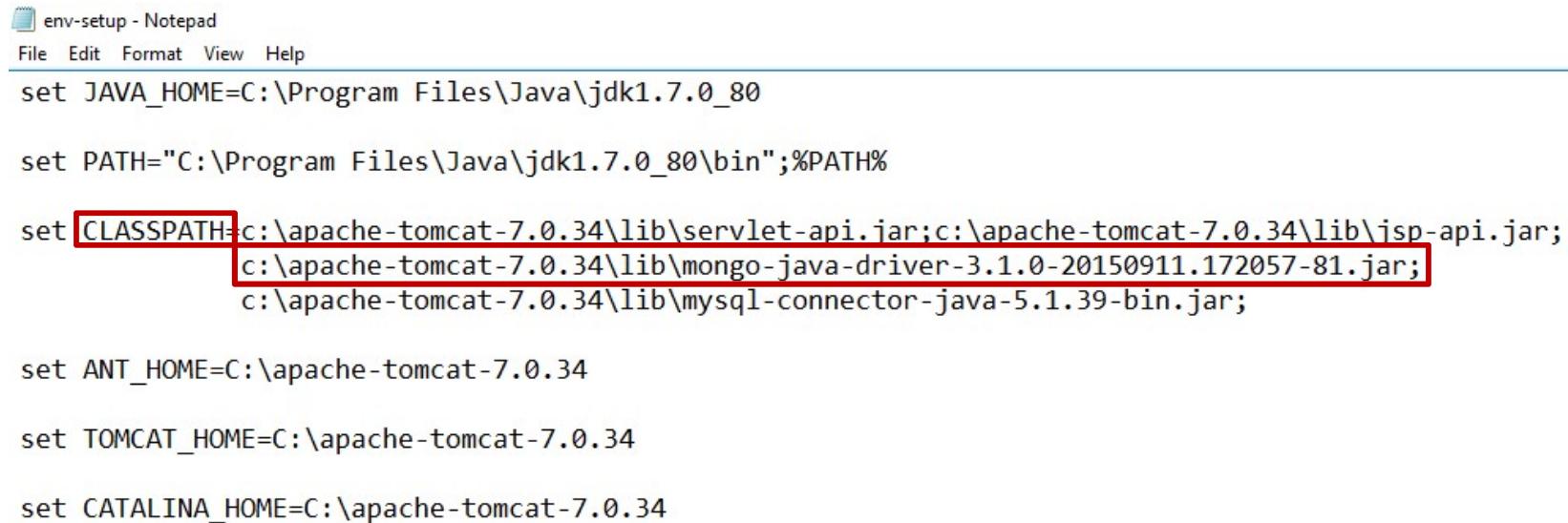
License	Apache 2.0
Categories	MongoDB Clients
HomePage	http://www.mongodb.org
Date	(Oct 06, 2015)
Files	Download (JAR) (987 KB)
Repositories	Central Sonatype Releases
Used By	713 artifacts

Maven Gradle SBT Ivy Grape Leiningen Buildr

Click on the download button for downloading Mongo java Driver

5. Compile and run

- Here is the snapshot of my ‘env-setup-for-tomcat_backup.bat’
- The location of the JAR files highlighted will differ based on where they are present on your computer
- Please make sure you do the changes accordingly



The screenshot shows a Notepad window titled "env-setup - Notepad". The menu bar includes File, Edit, Format, View, and Help. The content of the document is a batch script for setting environment variables:

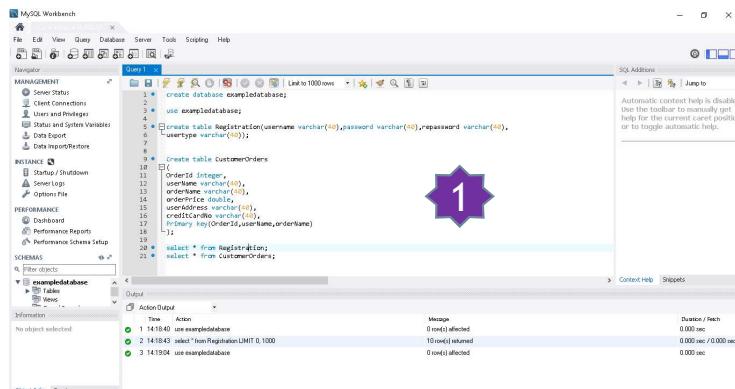
```
set JAVA_HOME=C:\Program Files\Java\jdk1.7.0_80
set PATH="C:\Program Files\Java\jdk1.7.0_80\bin";%PATH%
set CLASSPATH=c:\apache-tomcat-7.0.34\lib\servlet-api.jar;c:\apache-tomcat-7.0.34\lib\jsp-api.jar;
set ANT_HOME=C:\apache-tomcat-7.0.34
set TOMCAT_HOME=C:\apache-tomcat-7.0.34
set CATALINA_HOME=C:\apache-tomcat-7.0.34
```

The line "set CLASSPATH=c:\apache-tomcat-7.0.34\lib\servlet-api.jar;c:\apache-tomcat-7.0.34\lib\jsp-api.jar;" is highlighted with a red rectangle. The lines "c:\apache-tomcat-7.0.34\lib\mongo-java-driver-3.1.0-20150911.172057-81.jar;" and "c:\apache-tomcat-7.0.34\lib\mysql-connector-java-5.1.39-bin.jar;" are also highlighted with a red rectangle.



Things to Remember Before Running your Application in localhost:

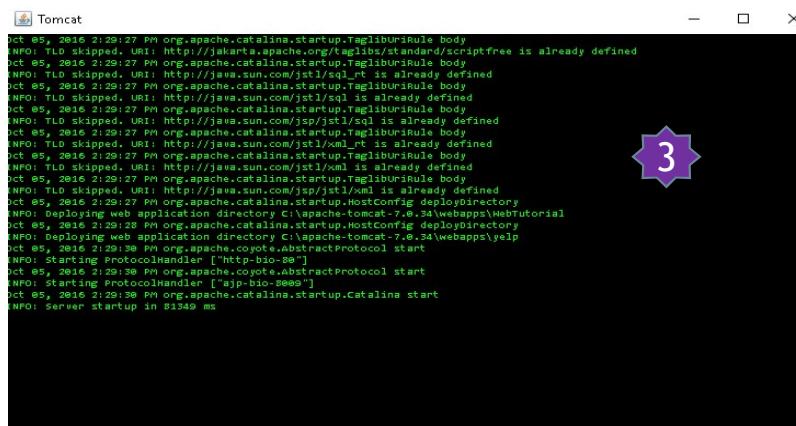
- Check **MySQL Server** is up and Running or else start the **MySQL Server** .
- Check **MongoDB Server** is up and Running or else start the **MongoDB Server** .
- Check **Apache Tomcat** is up and Running or else start the **Apache Tomcat** .



The screenshot shows a terminal window running the MongoDB command-line interface (mongod.exe). The log output includes several messages related to the startup of the MongoDB instance:

```
2016-08-05T14:23:41.202+0500 I CONTROL [initandlisten] listening on port 27017
2016-08-05T14:23:41.202+0500 I CONTROL [initandlisten] accepted connection from 127.0.0.1:486_64
2016-08-05T14:23:41.292+0500 I CONTROL [initandlisten] options: { }
2016-08-05T14:23:41.295+0500 I - [initandlisten] detected old files in C:\data\db\ created by the 'mmapv1' storage engine, so setting the active st
2016-08-05T14:23:41.300+0500 I JOURNAL [initandlisten] journal dir=C:\data\db\journal
2016-08-05T14:23:41.306+0500 I JOURNAL [initandlisten] recover : no journal files present, no recovery needed
2016-08-05T14:23:41.308+0500 I NETWORK [initandlisten] waiting for connections on port 27017
2016-08-05T14:23:41.308+0500 I JOURNAL [journal writer] Journal writer thread started
2016-08-05T14:23:41.729+0500 I NETWORK [hostNameCanonicalizationWorker] Starting hostname canonicalization worker
2016-08-05T14:23:41.729+0500 I FTDC [initandlisten] initializing full-time diagnostic data capture with directory 'C:\data\db\diagnostic.data'
2016-08-05T14:23:41.746+0500 I NETWORK [initandlisten] waiting for connections on port 27017
```

A purple starburst icon with the number '2' is overlaid on the bottom-right corner of the terminal window.



6. Example – Write Review:

To write a review for the product, click on ‘Write Review’ button on the products page

The screenshot shows a web page for a product listing. At the top, there is a navigation bar with links for Home, Consoles, Games, Tablets, Trending, Hello, Customer1, Account, Logout, and Cart(0). On the left side, there are two sidebar menus: 'Consoles' which lists Microsoft, Sony, and Nintendo; and 'Games' which lists Electronic Arts, Activision, and Take-Two Interactive. The main content area is titled 'Microsoft Consoles' and displays two products: the 'XBox 360' priced at \$299.0 and the 'XBox One' priced at \$399.0. Each product card features an image of the console and controller, a 'Buy Now' button, a 'WriteReview' button (which is highlighted with a red box), and a 'ViewReview' button.

6. Example – Write Review:

On clicking the WriteReview Button from products page user will be directed to WriteReview webpage where he can give review for product.

Click the SubmitReview button to store the review in Mongo database

ViewOrder Hello,Customer1 Account Logout Cart(1)

Review

Product Name:	xbox360
Product Type:	consoles
Product Maker:	microsoft

Review Rating: 2 ▼

Review Date: 10/04/2016

Review Text:
Nice Game to Play

SubmitReview



6. Example – Write Review:

On clicking the SubmitReview button user will get response that reviews for product is stored in database

The screenshot shows a web application interface for "Game Speed". At the top, there is a navigation bar with links for Home, Consoles, Games, Tablets, Trending, and Account, along with a Logout link and a Cart(0) button. A search bar is also present. On the left side, there are two sidebar boxes: one for "Consoles" listing Microsoft, Sony, and Nintendo, and another for "Games" listing Electronic Arts and Activision. The main content area has a heading "Review" and a message "Review for xbox360 Stored".



6. Example – Write Review:

Submitting one more Review for product

The screenshot shows a web application interface. At the top, there is a navigation bar with links for Home, Consoles, Games, Tablets, Trending, Hello, Customer1, Account, Logout, and Cart(0). On the left side, there are two sidebar boxes: 'Consoles' which lists Microsoft, Sony, and Nintendo; and 'Games' which lists Electronic Arts, Activision, and Take-Two Interactive. The main content area is titled 'Review' and contains a form. The form fields include: Product Name: xbox360, Product Type: consoles, Product Maker: microsoft, Review Rating: 5 (selected from a dropdown), Review Date: 09/13/2016, and a large text area for Review Text containing the placeholder text 'Amazing Game to Play'. A blue 'SubmitReview' button is located at the bottom of the form.

Product Name:	xbox360
Product Type:	consoles
Product Maker:	microsoft

Review Rating: 5 ▼

Review Date: 09/13/2016

Review Text:
Amazing Game to Play

SubmitReview



6. Example – Write Review:

On clicking the SubmitReview button user will get response that reviews for product is stored in database

The screenshot shows a web application interface for "Game Speed". At the top, there is a navigation bar with links for Home, Consoles, Games, Tablets, Trending, and Account, along with a Logout link and a Cart(0) button. A search bar is also present. On the left side, there are two sidebar boxes: one for "Consoles" listing Microsoft, Sony, and Nintendo, and another for "Games" listing Electronic Arts and Activision. The main content area has a heading "Review" and a message "Review for xbox360 Stored".



6. Example – View Review:

You can view the review submitted by clicking on ViewReview button on products page

The screenshot shows a web page with a purple header bar containing navigation links: Home, Consoles, Games, Tablets, Trending, Hello, Customer1, Account, Logout, and Cart(0). Below the header, there are two sidebar boxes: 'Consoles' (listing Microsoft, Sony, and Nintendo) and 'Games' (listing Electronic Arts, Activision, and Take-Two Interactive). The main content area is titled 'Microsoft Consoles' and displays two product cards: 'XBox 360 \$299.0' and 'XBox One \$399.0'. Each card features an image of the console and controller, a 'Buy Now' button, a 'WriteReview' button, and a 'ViewReview' button. The 'ViewReview' button for the XBox One card is highlighted with a red border.



6. Example – View Review:

All the reviews for the product will be retrieved from mongo db and displayed in web page

The screenshot shows a web application interface. On the left, there is a sidebar with three categories: 'Consoles', 'Games', and 'Tablets'. Under 'Consoles', there are links to Microsoft, Sony, and Nintendo. Under 'Games', there are links to Electronic Arts, Activision, and Take-Two Interactive. The main content area has a header with 'ViewOrder', 'Hello,Customer1', 'Account', 'Logout', and 'Cart(2)'. Below the header, there is a section titled 'Review' containing two tables of review data. A purple arrow points from a callout box on the right to the first table, and another purple arrow points from the same callout box to the second table.

Both the reviews are showed in web page

Product Name:	xbox360
userName:	customer1
Review Rating:	2
Review Date:	2016-10-04
Review Text:	Nice Game to Play

Product Name:	xbox360
userName:	customer1
Review Rating:	5
Review Date:	2016-09-13
Review Text:	Amazing Game to Play



6. Example – View Review:

Check in the mongo shell if the myReviews collection is created inside example database and data for two reviews is stored in it



The screenshot shows a terminal window titled "C:\Program Files\MongoDB\Server\3.2\bin\mongo.exe". The window contains the following MongoDB shell command and its results:

```
> db.myReviews.find()
{ "_id" : ObjectId("57f4495441e5be0344609c42"), "title" : "myReviews", "use
rName" : "customer1", "productName" : "xbox360", "productType" : "consoles"
, "productMaker" : "microsoft", "reviewRating" : "2", "reviewDate" : "2016-
10-04", "reviewText" : " Nice Game to Play" }
{ "_id" : ObjectId("57f4498341e5be0344609c45"), "title" : "myReviews", "use
rName" : "customer1", "productName" : "xbox360", "productType" : "consoles"
, "productMaker" : "microsoft", "reviewRating" : "5", "reviewDate" : "2016-
09-13", "reviewText" : " Amazing Game to Play" }
>
```



6. Example - Server Not Running For Registration:

Trying to Register when server is not up and running

The screenshot shows a website for "Game Speed". The header includes a green game controller icon, the site name "Game Speed", a search bar ("Search Product: search here.."), and navigation links for Home, Consoles, Games, Tablets, Trending, View Order, Login, and Cart(0). A purple callout box on the right states: "If mySql server not running gives an error message". A purple arrow points from this text to a red error message "MySQL server is not up and running" displayed above the login form. The login form contains fields for Username (filled with "customer1"), Password, Re-Password, and User Type (set to "Customer"), along with a "Create User" button.

If mySql server not
running gives an
error message

MySQL server is not up and running

Username: customer1

Password:
Re-Password:

User Type: Customer

Create User



6. Example - Server Not Running For Orders:

Trying to Place order when server is not up and running

The screenshot shows a web application for "Game Speed". The header features a green game controller icon and the text "Game Speed". A search bar is on the right. The main menu includes "Home", "Consoles", "Games", "Tablets", and "Trending". On the left, there are two green sidebar boxes: "Consoles" listing "Microsoft", "Sony", and "Nintendo"; and "Games" listing "Electronic Arts", "Activision", and "Take-Two Interactive". The central area is titled "Order" and contains a table with the following data:

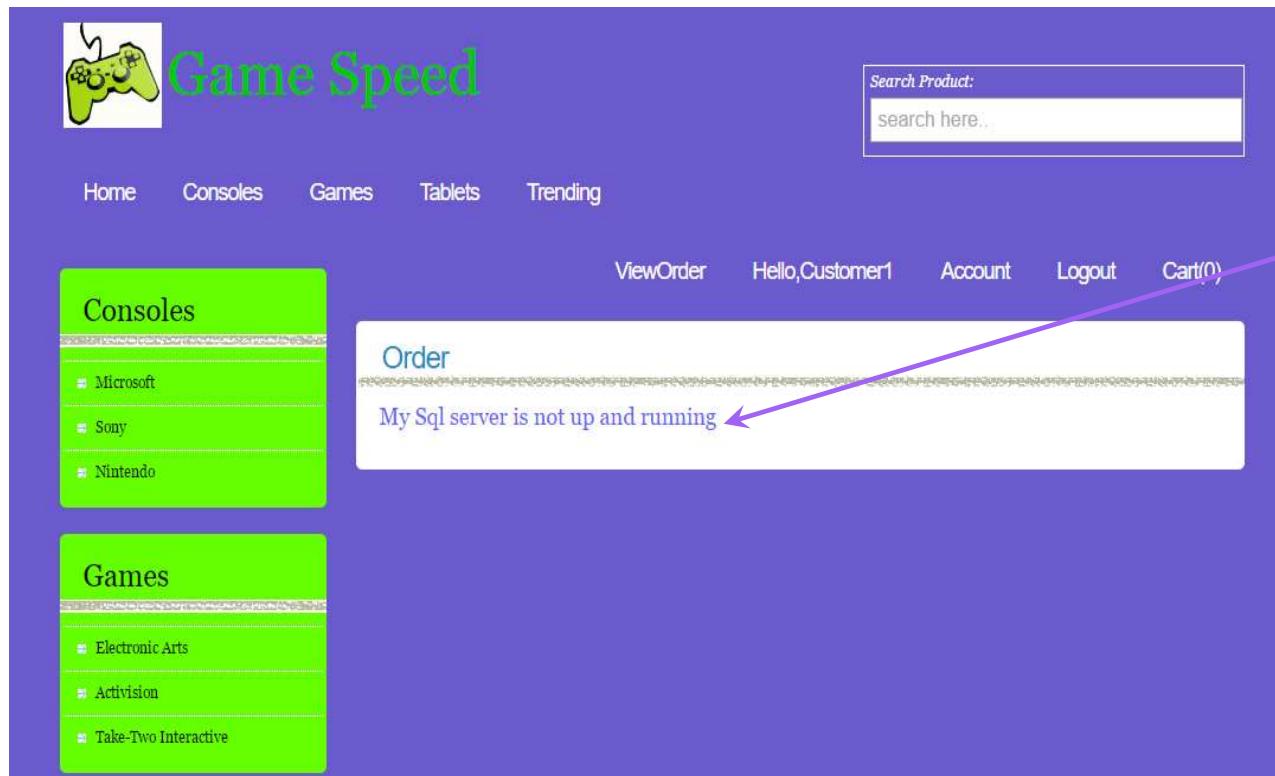
Customer Name:	customer1
Product Purchased:	xbox360
Product Price:	399.99
Total Order Cost	399.99

Below the table, there are input fields for "Credit/accountNo" containing "80456666" and "Customer Address" containing "Chicago Illinois". A blue "Submit" button is at the bottom.



6. Example - Server Not Running For Orders:

Trying to Place order when server is not up and running



If MySQL server not running gives an error message



6. Example – Write Review when MongoDB Server not running:

Trying to submit review for Product

The screenshot shows a web application interface for "Game Speed". The top navigation bar includes links for Home, Consoles, Games, Tablets, Trending, ViewOrder, Hello,Customer1, Account, Logout, and Cart(0). On the left, there are two green sidebar boxes: one for "Consoles" listing Microsoft, Sony, and Nintendo; and another for "Games" listing Electronic Arts, Activision, and Take-Two Interactive. The main content area is titled "Review" and contains a form with the following fields:
Product Name: xbox360
Product Type: consoles
Product Maker: microsoft
Review Rating: 4 ▼
Review Date: 10/07/2016
Review Text: Amazing game to play
SubmitReview



6. Example – Write Review when MongoDB Server not running:

Trying to submit review for Product

The screenshot shows a web browser displaying the 'Game Speed' website. The page has a purple header with a green game controller icon and the text 'Game Speed'. It features a navigation bar with links for Home, Consoles, Games, Tablets, and Trending. On the left, there are two sidebar boxes: 'Consoles' listing Microsoft, Sony, and Nintendo, and 'Games' listing Electronic Arts, Activision, and Take-Two Interactive. The main content area has a 'Search Product:' input field. A 'Review' button is visible. A message box contains the text 'Mongo Db is not up and running'. A purple arrow points from this message box to a callout box on the right.

Search Product:
search here...

Home Consoles Games Tablets Trending

ViewOrder Hello,Customer1 Account Logout Cart(0)

Consoles

- Microsoft
- Sony
- Nintendo

Games

- Electronic Arts
- Activision
- Take-Two Interactive

Review

Mongo Db is not up and running

If MongoDB server
not running gives an
error message



6. Example – View Review when MongoDB Server not running:

Trying to view review for Product

The screenshot shows a game store interface. At the top, there's a navigation bar with links for Home, Consoles, Games, Tablets, Trending, and a user account section for 'Hello, Customer1'. A shopping cart icon indicates 'Cart(0)'. On the left, there are two sidebar menus: 'Consoles' (listing Microsoft, Sony, and Nintendo) and 'Games' (listing Electronic Arts, Activision, and Take-Two Interactive). The main content area is titled 'Microsoft Consoles' and features two product cards. The first card is for the 'XBox 360' priced at '\$299.0', showing an image of the console and controller with 'Buy Now', 'WriteReview', and 'ViewReview' buttons below. The second card is for the 'XBox One' priced at '\$399.0', showing an image of the console and controller with similar buttons. A red box highlights the 'ViewReview' button under the Xbox One listing.



6. Example – View Review when MongoDB Server not running:

Trying to view review for Product

The screenshot shows a web application interface for 'Game Speed'. At the top left is a green game controller icon. Next to it, the site's name 'Game Speed' is displayed in a large, stylized green font. To the right of the name is a search bar with the placeholder text 'Search Product:' and 'search here...'. Below the header, there is a navigation menu with links: Home, Consoles, Games, Tablets, and Trending. On the far right of the header are links for ViewOrder, Hello,Customer1, Account, Logout, and Cart(0).

The main content area features two sidebar menus. The first sidebar, titled 'Consoles', lists Microsoft, Sony, and Nintendo. The second sidebar, titled 'Games', lists Electronic Arts, Activision, and Take-Two Interactive.

In the center of the page, there is a white rectangular box with a thin gray border. The word 'Review' is written in blue at the top left of this box. Below it, the text 'Mongo Db is not up and running' is displayed in blue. A purple arrow points from this text towards a callout box on the right side of the screen.

If MongoDB server
not running gives an
error message



7. Code Snippet

Walkthrough to get connect to
Database from Servlet



MongoDBDataStoreUtilities class to connect Database from Servlet

```
public class MongoDBDataStoreUtilities
{
    static DBCollection myReviews;
    public static void getConnection()
    {
        MongoClient mongo;
        mongo = new MongoClient("localhost", 27017);

        DB db = mongo.getDB("CustomerReviews");
        myReviews= db.getCollection("myReviews");
    }
}
```

Connecting to
CustomerReviews
database

Getting Reviews data
to DbCollection object



Walkthrough for Storing Reviews Code Snippet



Walkthrough for Storing reviews

```
public void storeReview(String productname, String producttype, String reviewrating, String reviewdate, String reviewtext)
{
    HashMap<String, ArrayList<Review>> reviews = new HashMap<String, ArrayList<Review>>();
    try
    {reviews=MongoDBDataStoreUtilities.selectReview();}
    catch(Exception e)
    {}
    if(!reviews.containsKey(productname))
    {
        ArrayList<Review> arr = new ArrayList<Review>();
        reviews.put(productname, arr);
    }
    ArrayList<Review> listReview = reviews.get(productname);
    Review review = new Review(productname,username(),producttype,reviewrating,reviewdate,reviewtext);
    listReview.add(review);
    try
    {
        MongoDBDataStoreUtilities.insertReview(productname,username(),producttype,reviewrating,reviewdate,reviewtext)
    }
    catch(Exception e)
    { }
}
```

Calling utility function to select data from database and storing reviews in hashmap

Calling utility function to inserting reviews in database



Utility Function for Selecting Review Data into Hashmap

```
public static HashMap<String, ArrayList<Review>> selectReview()
{
    getConnection();
    HashMap<String, ArrayList<Review>> reviewHashmap=new HashMap<String, ArrayList<Review>>();
    DBCursor cursor = myReviews.find();
    while (cursor.hasNext())
    {
        BasicDBObject obj = (BasicDBObject) cursor.next();
        if(! reviewHashmap.containsKey(obj.getString("productName")))
        {
            ArrayList<Review> arr = new ArrayList<Review>();
            reviewHashmap.put(obj.getString("productName"), arr);
        }
        ArrayList<Review> listReview = reviewHashmap.get(obj.getString("productName"));
        Review review =new
        Review(obj.getString("productName"),obj.getString("userNmae"),obj.getString("productType"),obj.getStr
        ing("reviewRating"),obj.getString("reviewDate"),obj.getString("reviewText"));
        listReview.add(review);
    }
    return reviewHashmap;
}
```

DBCursor used to store table data obtained from database in servlet

Iterate through Cursor and Store each review into class object

Utility Function for Writing Reviews into Mongo database

```
public static void insertReview(String productname, String username, String  
producttype, String reviewrating, String reviewdate, String reviewtext)  
{  
    getConnection();  
    BasicDBObject doc = new BasicDBObject("title", "myReviews").  
        append("userName", username).  
        append("productName", productname).  
        append("productType", producttype).  
        append("reviewRating", reviewrating).  
        append("reviewDate", reviewdate).  
        append("reviewText", reviewtext);  
    myReviews.insert(doc);  
}
```

Creating a
BasicObject to insert
data into database

Specifying each
column to insert
value

DbCollection.insert()
Will insert data into
database



MongoDB Data Analytics

Tutorial – 3, Advanced queries



Mongo DB Find/Query Data:

- Use ‘find’ method to query MongoDB to retrieve data from a collection
- ‘find’ can be used with a single collection.
- Using queries, you can either return;
 - All the documents in a collection
 - Only the documents that match a certain filter/criteria



Find all documents in a collection:

- In order to find all the documents in a collection, use the ‘find’ query without any parameters
- dbCursor - is a variable of type DBCursor, this variable can be used as an iterator and print all the values from the collection
- myReviews - Collection that has been used in the Application.
- find() - is an empty query, this will return all the documents from the given collection
- Example:
 - `DBCursor dbCursor = myReviews.find();`



Create a ‘query’

- A query can be created in the following way;

```
BasicDBObject query = new BasicDBObject();
```

- Now, once the query object is created, you can add multiple conditions in the following way;

```
query.put(Key, Value);
```

In order to find specific documents in a collection, use `find` along with some query values

- Example : Return the documents where the product name is ‘XBOX_ONE’ :

```
query.put("productName", "XBOX_ONE");
```

```
DBCursor dbCursor = myReviews.find(query);
```



Operators

- You can use different operators to conveniently filter the data based on different requirements
- Suppose, we want to filter only those reviews from our collection which have a rating of more than 3, we need to use '\$gt' (greater than)
- Example Return the documents where the review rating is above 3

```
BasicDBObject query = new BasicDBObject();
query.put("reviewRating", new BasicDBObject("$gt", 3));
DBCursor dbCursor = myReviews.find(query);
```

- Please refer this link for more information:
<https://docs.mongodb.org/manual/reference/operator/query/>



Limit and Sort

- ‘limit()’ accepts an integer value
- ‘sort()’ accepts an object of type DBObject
- Example: Return top 5 products based on maximum rating

```
int returnLimit = 5;
```

Created a new sort object

```
DBObject sort = new BasicDBObject();
```

Specify the field that you want to sort on, and the direction of the sort

```
sort.put("reviewRating",-1);
```

```
dbCursor = myReviews.find(query).limit(returnLimit).sort(sort);
```



Aggregation in MongoDB:

- The aggregate method accepts as its argument an array of stages, where each stage, processed sequentially, describes a data processing step.
- More information on aggregation can be found here:
<https://docs.mongodb.org/getting-started/java/aggregation/>



Stages in Aggregation – \$match

- \$match - This is similar to ‘Where’ in SQL
- Example Match the documents where rating is 5:
 - DBObject match = new BasicDBObject("\$match", new BasicDBObject("reviewRating", 5));
- Matching stage is optional



Stages in Aggregation – \$group

- \$group - This is similar to SQL's 'GROUP BY' clause

- Example grouping based on retailer city:

```
DBObject groupFields = new BasicDBObject("_id", 0);
groupFields.put("_id", "$retailerCity");
groupFields.put("count", new BasicDBObject("$sum", 1));
DBObject group = new BasicDBObject("$group", groupFields);
```

Group by is done on retailer city fields using _id as key to group by

Increment the count by 1 using \$sum command



Stages in Aggregation – \$project

- \$project - This is similar to ‘SELECT’ in SQL
- Vertically Slicing Data from the Original Database.

- Example Getting count based on retailer city :

```
DBObject projectFields = new BasicDBObject("_id", 0);
projectFields.put("city", "$_id");
projectFields.put("Review Count", "$count");
DBObject project = new BasicDBObject("$project", projectFields);
```

Project Fields which
we want to display in
the output



Stages in Aggregation – \$limit and \$sort

Example: Return top 5 products based on maximum rating;

```
DBObject sort = new BasicDBObject();
```

Specify the field that you want to sort on, and the direction of the sort
sort.put("reviewRating",-1);

```
DBObject limit=new BasicDBObject();
DBObject orderby=new BasicDBObject();
```

Adding sort object in DbObject
orderby=new BasicDBObject("\$sort",sort);

```
limit=new BasicDBObject("$limit",5);
aggregate = myReviews.aggregate(group,project,orderby,limit);
```



Final Stage in Aggregation:

- Now that we are done with the different stages, it is time to run the query
- Example;

```
AggregationOutput aggregate = myReviews.aggregate(match,group,project,orderby,limit);
for (DBObject result : aggregate.results()) {
    BasicDBObject bobj = (BasicDBObject) result;
    System.out.println(bobj.getString("City"));
    System.out.println(bobj.getString("Review Count"));
}
```
- Once the aggregate function is run, you can iterate through the result and print the required fields



Data Analytics:

- We can use complex queries in Mongo DB to perform data analysis on the collection.
- This tutorial will demonstrate a few scenarios where you can construct dynamic queries and display the result.
- All the queries are created dynamically based on the filters selected on the screen.



Trending link:

Clicking on the trending button will take us to the page where we will display
Top five most liked products,
Top five most reviewed products regardless of the rating
Top 5 zip code based on no of products reviewed

The screenshot shows the Game Speed website interface. At the top, there is a navigation bar with links for Home, Consoles, Games, Tablets, Trending (which is highlighted with a red box), and DataAnalytics. To the right of the navigation bar is a search bar labeled "Search Product:" with the placeholder "search here...". Below the navigation bar, there are two sidebar menus: "Consoles" (listing Microsoft, Sony, and Nintendo) and "Games" (listing Electronic Arts, Activision, and Take-Two Interactive). The main content area features a welcome message "Welcome to GameSpeed" above images of three game controllers: a PS4 controller, an Xbox One controller, and a Wii U controller. The background of the page is purple.

Trending link for user – Query Outputs :

[ViewOrder](#) [Hello,Customer1](#) [Account](#) [Logout](#) [Cart\(0\)](#)

Consoles

- » Microsoft
- » Sony
- » Nintendo

Games

- » Electronic Arts
- » Activision
- » Take-Two Interactive

Tablets

- » Apple
- » Microsoft
- » Samsung

Top 5 Products based on review rating

Product Name	Maximum Rating
WII2	5
xbox360	5
PS4	3
Xbox One	2
PS3	2

Top 5 Products based on review count

Product Name	Maximum No of Review for Product
WII2	2
xbox360	2
PS4	1
Xbox One	1
PS3	1



Trending link for user – Query Outputs :

<ul style="list-style-type: none">■ Take-Two Interactive	<p>Top 5 Products based on review count</p> <table><thead><tr><th>Product Name</th><th>Maximum No of Review for Product</th></tr></thead><tbody><tr><td>WII2</td><td>2</td></tr><tr><td>xbox360</td><td>2</td></tr><tr><td>PS4</td><td>1</td></tr><tr><td>Xbox One</td><td>1</td></tr><tr><td>PS3</td><td>1</td></tr></tbody></table>	Product Name	Maximum No of Review for Product	WII2	2	xbox360	2	PS4	1	Xbox One	1	PS3	1
Product Name	Maximum No of Review for Product												
WII2	2												
xbox360	2												
PS4	1												
Xbox One	1												
PS3	1												
<ul style="list-style-type: none">■ Tablets■ Apple■ Microsoft■ Samsung	<p>Top 5 Zip code based on Maximum no of products reviewed</p> <table><thead><tr><th>Zip Code</th><th>Maximum no of Products Reviewed in Zip Code</th></tr></thead><tbody><tr><td>690000</td><td>2</td></tr><tr><td>60616</td><td>2</td></tr><tr><td>60117</td><td>1</td></tr><tr><td>60612</td><td>1</td></tr><tr><td>60011</td><td>1</td></tr></tbody></table>	Zip Code	Maximum no of Products Reviewed in Zip Code	690000	2	60616	2	60117	1	60612	1	60011	1
Zip Code	Maximum no of Products Reviewed in Zip Code												
690000	2												
60616	2												
60117	1												
60612	1												
60011	1												



Data Analytics

- Clicking on the data analytics link will take us to data analytics page where we can perform analytics required

The screenshot shows a purple-themed website for "Game Speed". At the top left is a green video game controller icon. To its right, the word "Game Speed" is written in large green letters. On the far right of the header is a search bar with the placeholder text "Search Product: search here..". Below the header, there is a navigation bar with several items: Home, Consoles, Games, Tablets, Trending, and DataAnalytics. The "DataAnalytics" button is highlighted with a red rectangular box. To the right of the navigation bar are links for ViewOrder, Hello,Customer1, Account, Logout, and Cart(0). On the left side of the main content area, there are two green boxes. The top one is titled "Consoles" and lists "Microsoft", "Sony", and "Nintendo". The bottom one is titled "Games" and lists "Electronic Arts", "Activision", and "Take-Two Interactive". The main content area features a white box with the text "Welcome to GameSpeed" and three images of game controllers: a black PS4 DualShock 4, a black XBOX ONE controller, and a white Wii U Pro Controller. The entire website has a purple background.



Data Analytics – Store Manager

- This page will contain all fields for selection by which we can perform data analytics

The screenshot shows a web-based store management system with a purple header and sidebar, and a white main content area.

Header: Home, Consoles, Games, Tablets, Trending, DataAnalytics, ViewOrder, Hello,Customer1, Account, Logout, Cart(0)

Left Sidebar (Consoles):

- Consoles
 - Microsoft
 - Sony
 - Nintendo

Left Sidebar (Games):

- Games
 - Electronic Arts
 - Activision
 - Take-Two Interactive

Left Sidebar (Tablets):

- Tablets

Main Content Area: Data Analytics on Review

Product Name:

Product Price: Equals, Greater Than, Less Than

Review Rating:

Retailer City:

Group By: Count, Detail

Find Data

Query 1 – Print the list of all the reviews

- Press the find data button without any selection and you will get list of all reviews

The screenshot shows a web-based application interface for data analytics. At the top, there is a navigation bar with links: Home, Consoles, Games, Tablets, Trending, and DataAnalytics. On the right side of the header, there are links for ViewOrder, Hello,Customer1, Account, Logout, and Cart(0). The main content area is titled "Data Analytics on Review". It features several filter options:

- Product Name: A dropdown menu set to "All Products".
- Product Price: An input field containing "0". To its right are three radio buttons: "Equals" (selected), "Greater Than", and "Less Than".
- Review Rating: A dropdown menu set to "1".
- Retailer City: An input field.
- Group By: A dropdown menu set to "City". To its right are two radio buttons: "Count" (selected) and "Detail".

A large blue "Find Data" button is located at the bottom of the filter section. On the left side of the page, there is a sidebar with three sections: "Consoles" (listing Microsoft, Sony, Nintendo), "Games" (listing Electronic Arts, Activision, Take-Two Interactive), and "Tablets".



Result 1 - Print the list of all the reviews

The screenshot shows a web application interface. At the top, there is a navigation bar with links: Home, Consoles, Games, Tablets, Trending, and DataAnalytics. On the right side of the header, there are links for ViewOrder, Hello,Customer1, Account, Logout, and Cart(0).

The left side features a sidebar with three sections: "Consoles" (listing Microsoft, Sony, Nintendo), "Games" (listing Electronic Arts, Activision, Take-Two Interactive), and "Tablets" (listing Apple, Microsoft).

The main content area is titled "Data Analytics on Review" and contains two tables of review data:

Review	
Name:	ps4
Rating:	5
Price:	349
Retailer City:	chicago
Date:	2016-10-14
Review Text:	Good !!

Review	
Name:	ps4
Rating:	4
Price:	349
Retailer City:	austin
Date:	2016-10-13
Review Text:	Good for the Price !!



Query 2 – Print a list of reviews where rating is more than 3

- Select the filter for rating and option greater than

The screenshot shows a web-based application interface. On the left, there is a sidebar with three main sections: "Consoles", "Games", and "Tablets". Each section contains a list of brands or companies. The "Consoles" section lists Microsoft, Sony, and Nintendo. The "Games" section lists Electronic Arts, Activision, and Take-Two Interactive. The "Tablets" section lists Apple and Microsoft.

The main content area is titled "Data Analytics on Review". It contains several input fields and dropdown menus:

- A dropdown menu for "Product Name" with "All Products" selected.
- A dropdown menu for "Review Rating" with "3" selected.
- A dropdown menu for "Retailer City" which is empty.
- A dropdown menu for "Group By" with "City" selected.
- Checkboxes for "Select" and "Group By" are checked.
- Radio buttons for filtering options:
 - "Review Rating": Equals (selected), Greater Than, Less Than.
 - "Retailer City": Equals (selected), Greater Than.
 - "Group By": Count (selected), Detail.
- A blue "Find Data" button at the bottom right of the form.



Result 2 - Print a list of reviews where rating is more than 3

Only reviews with rating greater than 3 will be displayed

The screenshot shows a web-based application interface. On the left, there is a sidebar with three sections: "Consoles" (listing Microsoft, Sony, Nintendo), "Games" (listing Electronic Arts, Activision, Take-Two Interactive), and "Tablets" (listing Apple, Microsoft, Samsung). On the right, the main content area has a header with links: "ViewOrder", "Hello,Customer1", "Account", "Logout", and "Cart(0)". Below the header, the title "Data Analytics on Review" is displayed. Under this title, there are two tables, each representing a review. The first review table contains the following data:

Review	
Name:	xbox360
Rating:	5
Price:	399
Retailer City:	chicago
Date:	2016-10-15
Review Text:	Good

The second review table contains the following data:

Review	
Name:	WII2
Rating:	5
Price:	1100
Retailer City:	tenesse
Date:	2016-10-26
Review Text:	Very Interesting game to play



Query 3 - Get a list of products that got review rating 5 and price more than thousand

The screenshot shows the Game Speed website interface. At the top, there is a navigation bar with links for Home, Consoles, Games, Tablets, Trending, and DataAnalytics. A search bar is also present. On the left side, there are two sidebar menus: 'Consoles' (listing Microsoft, Sony, and Nintendo) and 'Games' (listing Electronic Arts, Activision, and Take-Two Interactive). The main content area is titled 'Data Analytics on Review' and contains several filter options:

- Product Name: A dropdown menu set to "All Products".
- Product Price: A dropdown menu set to "1000". To its right are three radio buttons: "Equals" (unchecked), "Greater Than" (checked), and "Less Than" (unchecked).
- Review Rating: A dropdown menu set to "5". To its right are three radio buttons: "Equals" (checked), "Greater Than" (unchecked), and "Less Than" (unchecked).
- Retailer City: An input field.
- Group By: A dropdown menu set to "City". To its right are two radio buttons: "Count" (checked) and "Detail" (unchecked).

A blue "Find Data" button is located at the bottom of the form.



Result 3 - Get a list of products that got review rating 5 and price more than thousand

Data will be displayed with the particular review which we added that has price greater than 1000 and rating 5

The screenshot shows a web application interface for "Game Speed". At the top, there is a logo of a green video game controller and the text "Game Speed". To the right is a search bar with placeholder text "Search Product: search here..". Below the header, there is a navigation menu with links: Home, Consoles, Games, Tablets, Trending, DataAnalytics, ViewOrder, Hello,Customer1, Account, Logout, and Cart(1). On the left side, there are two green-highlighted sections: "Consoles" listing Microsoft, Sony, and Nintendo, and "Games" listing Electronic Arts, Activision, and Take-Two Interactive. In the center, there is a white box titled "Data Analytics on Review" containing a table with the following data:

Review	
Name:	nintendo2
Rating:	5
Price:	1100
Retailer City:	chicago
Date:	2016-10-19
Review Text:	Good



Query 4 - Print a list of how many reviews for every product

Select the group by filter to get count based on products

The screenshot shows a web application interface with a purple header and sidebar, and a white content area for data analysis.

Header: Home, Consoles, Games, Tablets, Trending, DataAnalytics, ViewOrder, Hello,Customer1, Account, Logout, Cart(1)

Sidebar:

- Consoles:** Microsoft, Sony, Nintendo
- Games:** Electronic Arts, Activision, Take-Two Interactive

Data Analytics on Review Form:

- Product Name:** Select dropdown menu showing "All Products".
 - Equals
 - Greater Than
 - Less Than
- Product Price:** Select dropdown menu showing "0".
 - Equals
 - Greater Than
 - Less Than
- Review Rating:** Select dropdown menu showing "1".
 - Equals
 - Greater Than
 - Less Than
- Retailer City:** Text input field.
- Group By:** Checkboxes:
 - Product Name
 - Count
 - Detail
- Find Data:** Blue button.



Result 4 - Print a list of how many reviews for every product

The screenshot shows a web application interface for "Game Speed".

Header: A green logo featuring a video game controller and the text "Game Speed". To the right is a search bar with placeholder text "Search Product: search here..".

Main Navigation: A horizontal menu with links: Home, Consoles, Games, Tablets, Trending, and DataAnalytics.

User Session: On the right, it says "Hello,Customer1" and includes links for Account, Logout, and Cart(0).

Left Sidebar: Contains two sections: "Consoles" with links to Microsoft, Sony, and Nintendo; and "Games" with links to Electronic Arts, Activision, and Take-Two Interactive.

Central Content: A white box titled "Data Analytics on Review" contains a table showing the count of reviews for different products.

Name	Count
xbox360	2
Xbox One	1
WII2	2
PS4	1
PS3	1



Query 5 - Get the list of reviews for shoppers in Chicago

The screenshot shows a web-based application interface for data analytics. The top navigation bar includes links for Home, Consoles, Games, Tablets, Trending, and DataAnalytics. The user is logged in as "Hello,Customer1". A sidebar on the left lists categories: Consoles (Microsoft, Sony, Nintendo), Games (Electronic Arts, Activision, Take-Two Interactive), and Tablets (Apple). The main content area is titled "Data Analytics on Review" and contains the following filter settings:

- Product Name: All Products
- Product Price: 0
- Review Rating: 1
- Retailer City: chicago (selected)
- Group By: City
- Aggregation Options:
 - Equals (radio button selected)
 - Greater Than
 - Less Than
- Aggregation Options:
 - Equals (radio button selected)
 - Greater Than
- Count (radio button selected)
- Detail

A blue "Find Data" button is located at the bottom right of the filter panel.



Result 5 - Get the list of reviews for shoppers in Chicago

ViewOrder Hello,Customer1 Account Logout Cart(0)

Consoles

- Microsoft
- Sony
- Nintendo

Games

- Electronic Arts
- Activision
- Take-Two Interactive

Tablets

- Apple
- Microsoft
- Samsung

Data Analytics on Review

Review	
Name:	xbox360
Rating:	5
Price:	399
Retailer City:	chicago
Date:	2016-10-15
Review Text:	Good
Review	
Name:	Xbox One
Rating:	2
Price:	399
Retailer City:	chicago
Date:	2016-10-15
Review Text:	Not Working Properly



Sample Code: Top five zip-codes where maximum number of products sold

```
pw.print("<table id='bestseller'>");  
    groupFields = new BasicDBObject("_id", 0);  
    groupFields.put("count",new BasicDBObject("$sum",1));  
    groupFields.put("_id", "$zipCode");  
    group = new BasicDBObject("$group", groupFields);  
    sort = new BasicDBObject();  
    projectFields.put("value", "$_id");  
    projectFields.put("ReviewValue","$count");  
    project = new BasicDBObject("$project", projectFields);  
    sort.put("ReviewValue",-1);  
    orderby=new BasicDBObject("$sort",sort);  
    limit=new BasicDBObject("$limit",5);  
    aggregate = myReviews.aggregate(group,project,orderby,limit);  
    constructGroupByContent(aggregate,pw);  
pw.print("</table>");
```



Sample Code: Top five zip-codes where maximum number of products sold

```
public void constructGroupByContent(AggregationOutput aggregate, PrintWriter pw)
{
    for (DBObject result : aggregate.results()) {
        BasicDBObject bobj = (BasicDBObject) result;
        String tableData = "<tr><td> " + bobj.getString("value") + "</td>&nbsp"
                        + "<td>" + bobj.getString("ReviewValue") + "</td></tr>";
        pw.print(tableData);
    }
}
```



Sample Code for list of reviews where rating greater than 3:

```
int reviewRating = Integer.parseInt(request.getParameter("reviewRating"));
String compareRating = request.getParameter("compareRating");
String[] filters = request.getParameterValues("queryCheckBox");
myReviews=MongoDBDataStoreUtilities.getConnection();
BasicDBObject query = new BasicDBObject();
boolean noFilter = false;
boolean filterByRating = false;
if(filters != null){
    for (int i = 0; i < filters.length; i++) {
        //Check what all filters are ON
        //Build the query accordingly
        switch (filters[i]){
            case "reviewRating":
                filterByRating = true;
                if (compareRating.equals("EQUALS_TO")) {
                    query.put("reviewRating", reviewRating);
                }else{
                    query.put("reviewRating", new BasicDBObject("$gt", reviewRating));
                }
                break;}}}
DBCursor dbCursor = myReviews.find(query);
constructTableContent(dbCursor, pw);
```



Sample Code for list of reviews where rating greater than 3

```
public void constructTableContent(DBCursor dbCursor,PrintWriter pw)
{
    String tableData = "";
    pw.print("<table class='gridtable'>");
    while (dbCursor.hasNext())
    {
        BasicDBObject bobj = (BasicDBObject) dbCursor.next();
        tableData = "<tr><td align='center' colspan='2'>Review</td></tr><tr><td>Name: </td><td>" +
bobj.getString("productName") + "</td></tr>" +
        + "<tr><td>Rating:</td><td>" + bobj.getString("reviewRating") + "</td></tr>" +
        + "<tr><td>Date:</td><td>" + bobj.getString("reviewDate") + "</td></tr>" +
        + "<tr><td>Review Text:</td><td>" + bobj.getString("reviewText")+"</td></tr>";
        pw.print(tableData);
    }
    pw.print("</table>");
    //No data found
    if(dbCursor.count() == 0)
    {
        tableData = "<h2>No Data Found</h2>";
        pw.print(tableData);
    }
}
```

