Starting MySQL Shell (mysqlsh)

```
Starting prompt: #

1. # mysqlsh
2. session
3. \connect root@127.0.0.1
4. session
5. Session
6. \q
7. # mysqlsh root@127.0.0.1
8. session
```

MySQL Shell - Create & Drop Schema

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  JS
```

```
1. session.createSchema("test1")
2. session.getSchemas()
3. session.runSql('show databases')
4. session.DropSchema("test1")
5. session.dropSchema("test1")
6. session.getSchemas()
7. session.createSchema("workshop")
8. \use workshop
9. db
```

MySQL Shell - Changing schemas with var or \use

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  workshop  JS
```

```
1. \use workshop
2. db
3. session.getSchema('performance_schema');
4. db
5. var db = session.getSchema('performance_schema');
6. db
7. \use information_schema
8. \use workshop
9. db
```

MySQL Shell - Setting commands as variables

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  workshop  JS
```

```
1. var sdb = session.runSql('show databases')
2. sdb
3. \q
4. # mysqlsh root@127.0.0.1
5. sdb
```

MySQL Modes

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  JS
```

```
1. \sql
2. \py
3. \js
```

Create, List, and Drop Collections (tables)

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  workshop  JS
```

```
1. \use workshop
2. db.createCollection("test1")
3. db.createCollection("test2")
4. db.getCollections()
5. session.runSql('show tables')
6. db.dropCollection("test1")
7. db.getCollections()
8. db.dropCollection("test2")
9. db.getCollections()
```

Working with Collections (tables)

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  workshop  JS
```

```
1.  \q
2.  # mysqlsh root@127.0.0.1
3.  db.createCollection("foods")
4.  \use workshop
5.  db.createCollection("foods")
6.  db.foods.add({Name_First: "Fox", Name_Last: "Mulder", favorite_food:
    {Breakfast: "eggs and bacon", Lunch: "pulled pork sandwich", Dinner: "steak
    and baked potato"}})
```

Find Documents

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  workshop  JS
```

```
1. db.foods.find()
2. db.foods.find('Name_First = "Fox"')
3. db.foods.find('Name_First = "Jason"')
4. db.foods.find('Name_First = "Fox"').fields('favorite_food')
5. db.foods.find('Name_First = "Fox"').fields("favorite_food", "Name_Last")
6. db.foods.add(
      {
         Name_First: "Dana",
         Name_Last: "Scully",
         favorite_food: {
             Breakfast: "coffee with bagel",
             Lunch: "big salad",
             Dinner: "fish with slaw"
      }
      }
      )
7. db.foods.find( 'Name_First = "Dana"' ).fields( "favorite_food.Breakfast" )
```

Modify Documents

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  workshop  JS
```

```
1.  db.foods.modify("Name_First = 'Fox'").set("favorite_food", {Lunch: "Soup in a
    bread bowl", Dinner: "steak and broccoli"})
2.  db.foods.find('Name_First = "Fox"')
3.  db.foods.modify("Name_First = 'Fox'").arrayAppend("favorite_food", {Breakfast:
    "eggs and bacon"})
4.  db.foods.find('Name_First = "Fox"')
5.  db.foods.modify("Name_First = 'Fox'").arrayAppend("favorite_food", {Breakfast:
    "eggs and bacon", Lunch: "pulled pork sandwich", Dinner: "steak and baked
    potato" })
6.  db.foods.find('Name_First = "Fox"')
7.  db.foods.modify("Name_First = 'Fox'").arrayDelete("$.favorite_food[1]")
8.  db.foods.find('Name_First = "Fox"')
9.  db.foods.find("Name_First = 'Dana'")
10. db.foods.modify("Name_First = 'Dana'").set('favorite_food.Dinner', 'Pizza')
11. db.foods.find("Name_First = 'Dana'")
12. db.foods.modify("Name_First = 'Dana'").unset("favorite_food.Dinner")
13. db.foods.find("Name_First = 'Dana'")
14. var schema = session.getSchema('workshop')
15. var collection = schema.getCollection('foods')
16. collection.modify("Name_First = 'Dana'").patch({ Name_Middle: 'Katherine' })
17. db.foods.find("Name_First='Dana'")
```

Remove Documents

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  workshop  JS
```

```
1. db.foods.find('Name_First = "Fox"')
2. db.foods.remove('Name_First = "Fox"')
3. db.foods.find('Name_First = "Fox"')
```

Importing Large Data Set. (Slides 51 and 52)

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  workshop  JS
```

```
1. session.runSql('SHOW VARIABLES like "%packet%"')
2. session.runSql('SET @@GLOBAL.mysqlx_max_allowed_packet = 1073741824')
3. session.runSql('SHOW VARIABLES like "%packet%"')
4. session.getSchemas()
5. session.runSql('SET SQL_LOG_BIN=0')
6. session.createSchema("project")
7. \use project
8. util.importJson("./workshop/Doc_Store_Demo_File.json", {schema: "project",
   collection: "GoverningPersons"})
```

Create and Drop Indexes

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  project  JS
```

```
1. \use project
2. db.GoverningPersons.find().limit(1)
3. db.GoverningPersons.find('LastName = "FRISCH"')
4. db.GoverningPersons.find('LastName = "FRISCH"')
5. db.GoverningPersons.find('LastName = "FRISCH"')
6. db.GoverningPersons.find('LastName = "VARNELL"')
7. db.GoverningPersons.find('LastName = "VARNELL"')
8. db.GoverningPersons.find('LastName = "VARNELL"')
```

```
(Optional – on another window, create the index as you talk to save time)
```

Create and Use the Index

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  project  JS
```

```
1. db.GoverningPersons.createIndex("i_Name_Last", {fields: [{field: "$.LastName",
   type: "TEXT(20)"}]})
2. session.runSql('SHOW INDEX FROM project.GoverningPersons')
3. db.GoverningPersons.find('LastName = "FRISCH"')
4. db.GoverningPersons.find('LastName = "VARNELL"')
```

Drop Index

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  project  JS
```

```
1. db.GoverningPersons.dropIndex("i_Name_Last")
2. session.runSql('SHOW INDEX FROM project.GoverningPersons')
3. db.GoverningPersons.find('LastName = "FRISCH"')
```

Session Formats

```
1. shell.options.set('resultFormat','table')
2. session.runSql('select user, host from mysql.user')
3. shell.options.set('resultFormat','json')
4. session.runSql('select user, host from mysql.user')
5. shell.options.set('resultFormat','tabbed')
6. session.runSql('select user, host from mysql.user')
7. shell.options.set('resultFormat','vertical')
8. session.runSql('select user, host from mysql.user')
```

This following commands are not in the presentation - because the presentation was already very long. This is here for your reference only.

Working with Collections (tables)

Find Record & Create Index

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  project  JS
```

```
1. db.GoverningPersons.find().limit(1)
2. db.GoverningPersons.createIndex("i_id", {fields: [{field: "$._id", type:
   "TEXT(28)"}]})
3. db.GoverningPersons.find('_id = "00005d823f920000000000000004"')
```

Modify Record

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  project  JS
```

```
1. db.GoverningPersons.modify("_id =
   '00005d823f920000000000000004'").set("MiddleName", "JOHN")
2. db.GoverningPersons.find('_id = "00005d823f920000000000000004"')
```

Find Record via Multiple Fields - and Sort

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  project  JS
```

```
1. db.GoverningPersons.find("LastName='FRISCH' AND MiddleName = 'JOHN'")
2. db.GoverningPersons.createIndex("i_Name_Last", {fields: [{field: "$.LastName",
   type: "TEXT(20)"}]})
3. db.GoverningPersons.find("LastName='FRISCH' AND MiddleName = 'JOHN'")
4. db.GoverningPersons.find("LastName='FRISCH'").fields(["LastName",
   "FirstName"]).sort("LastName", "FirstName").limit(5)
5. session.runSql('SHOW INDEX FROM project.GoverningPersons')
```

Index Creation for Two Columns & Search for Two Fields

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  project  JS
```

```
1. db.GoverningPersons.dropIndex("i_Name_Last")
2. db.GoverningPersons.find("LastName = 'FRISCH' AND Zip = '99205'")
3. db.GoverningPersons.createIndex('i_last_name_zip', {fields: [ {field:
   '$.LastName', type: 'TEXT(10)'}, {field: '$.Zip', type: 'TEXT(10)'}]})
4. db.GoverningPersons.find("LastName = 'FRISCH' AND Zip = '99205'")
5. session.runSql('SHOW INDEX FROM project.GoverningPersons')
```

Transactions

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  project  JS
```

```
1. session.startTransaction()
2. db.GoverningPersons.modify("_id =
   '00005d823f920000000000000004'").set("MiddleName", "PETER")
3. session.rollback()
4. db.GoverningPersons.modify("_id =
   '00005d823f920000000000000004'").set("MiddleName", "STEVEN")
5. session.commit()
6. db.GoverningPersons.find("State='WA' AND Zip = '99205' AND LastName =
   'FRISCH'")
```

Session Formats (Default is table for runSql)

```
Starting prompt:  MySQL  127.0.0.1:33060+ ssl  project  JS
```

```
1. shell.options.set('resultFormat','table')
2. session.runSql('select user, host from mysql.user')
3. shell.options.set('resultFormat','json')
4. session.runSql('select user, host from mysql.user limit 2')
5. shell.options.set('resultFormat','tabbed')
6. session.runSql('select user, host from mysql.user')
7. shell.options.set('resultFormat','vertical')
8. session.runSql('select user, host from mysql.user order by user desc')
```