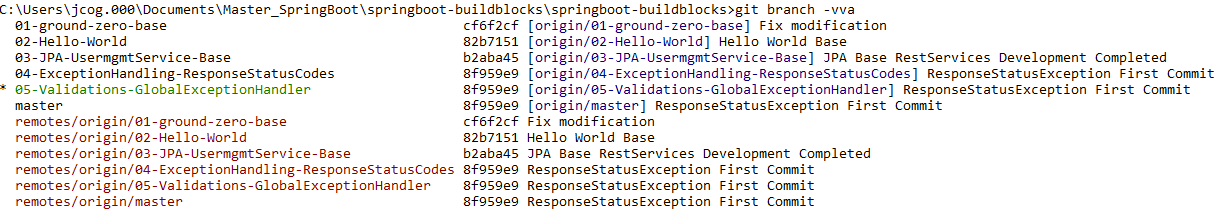
**Master SpringBoot**

**05-JPA-OneToMany Association:**

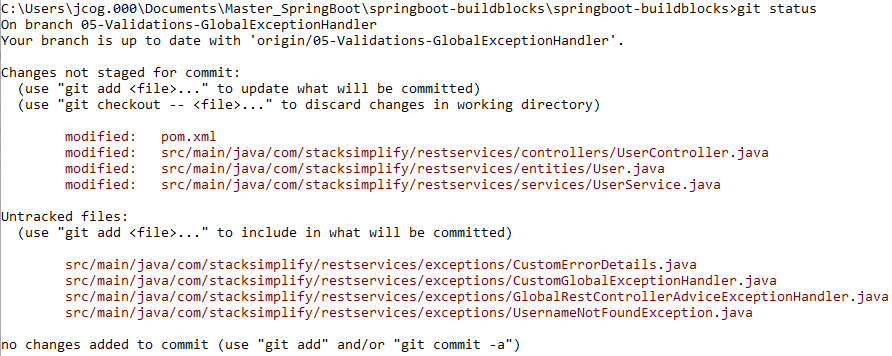
**Step-01: Create new GIT branch in local git repo and remote github repo**

* **git branch –vva**
* **git status**
* **git checkout –b 06-SpringBoot-JPA-OneToMany**
* **git push –set-upstream origin 06-SpringBoot-JPA-OneToMany**
* **git branch –a**

**git branch -vva**

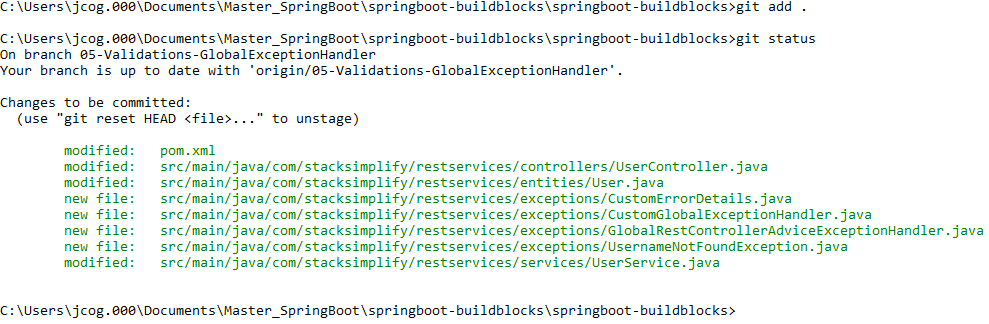


**git status**

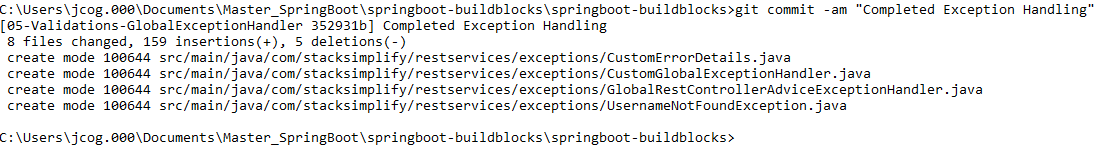


**Need to commit changes:**

**git add . git status**



**git commit –am “Completed Exception Handling”**



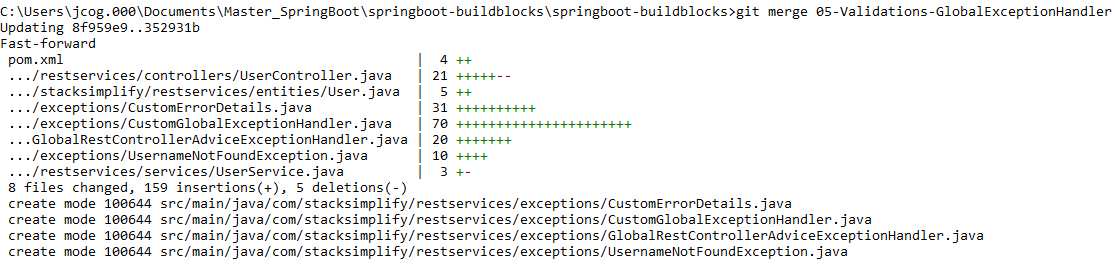
**On branch so, push to remote branch, git show-branch git push**



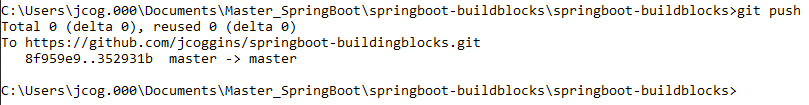
**Now move to local master and merge the changes git checkout master**



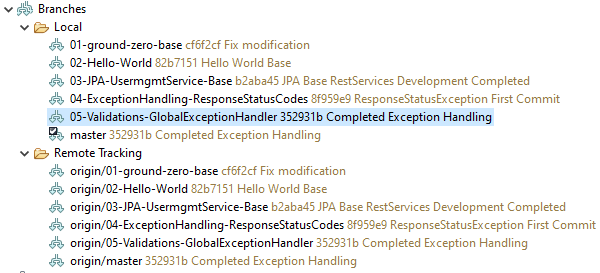
**Move the changes from Local branch to local master git merge 05-Validations-GlobalExceptionHandler**



**Now make the Remote Master the same: git push**



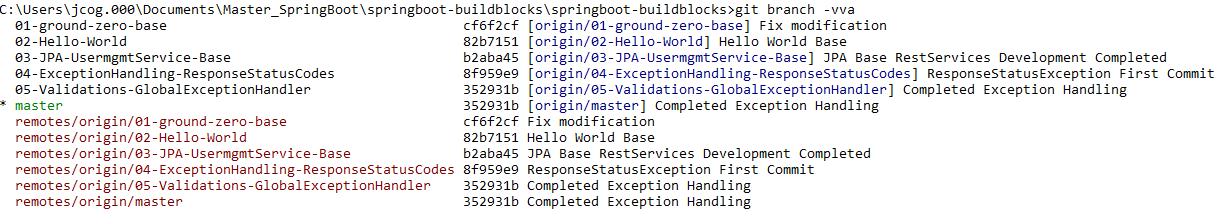
**Check is all branches are the same:**



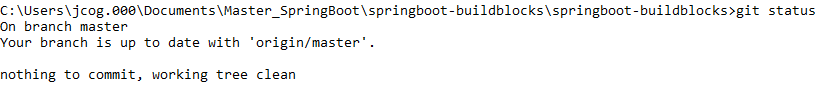
**Check the branches on GITHUB**



**git branch -vva**

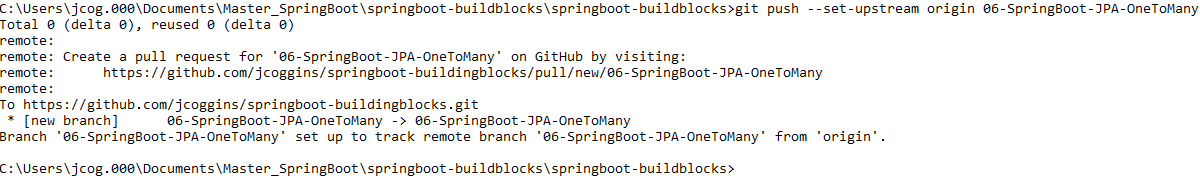


**git status**

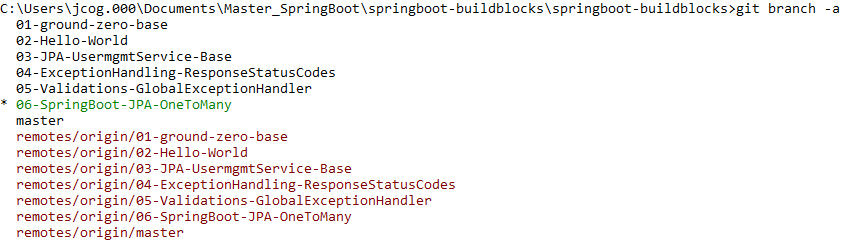
**git checkout –b 06-SpringBoot-JPA-OneToMany**



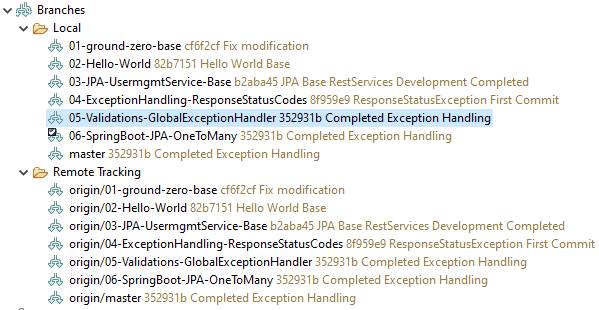
**git push –set-upstream origin 06-SpringBoot-JPA-OneToMany**



**git branch –a**



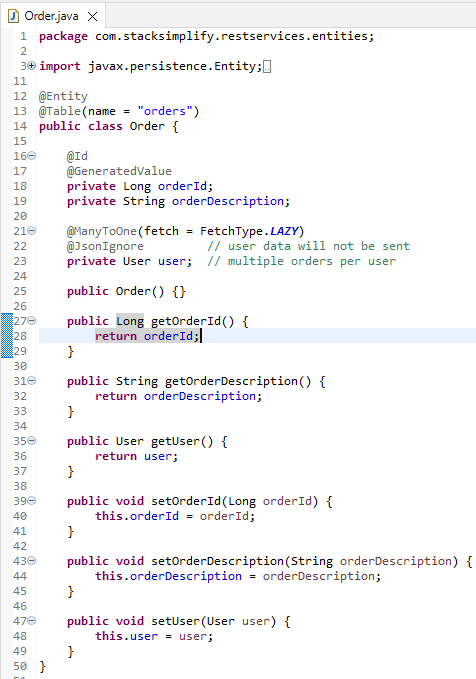
**Local branch and remote branch are the same**



**Step-02: Create Order entity and ManyToOne Mapping**

* **Entity Layer**
  + **Create Order Entity**
  + **Annotate with @Table(name = “orders”)**
  + **Add user variable**
  + **Add @ManyToOne Mapping**
  + **Add Fetch Type as Lazy**
  + **Add @JsonIgnore**
  + **Add Getters and Setters**
  + **Add NoArgument Constructor**

**Create order entity:**



**05-JPA-OneToMany Association:**

**Step-03: Update User entity with @OneToMany association**

* **Add orders variable**
* **Add @OneToMany Mapping**
* **Add MappedBy to user variable in Order Entity**
* **Add getters and setters for “orders”**
* **src/main/resources**
  + **update import.sql**

**Option#1: Verify Column and create insert query**

**Insert into orders values(2001, ‘order11’, 101);**

**Insert into orders values(2002, ‘order12’, 101);**

**Insert into orders values(2003, ‘order13’, 101);**

**Insert into orders values(2004, ‘order21’, 102);**

**Insert into orders values(2005, ‘order22’, 102);**

**Insert into orders values(2006, ‘order31’, 103);**

**Option#2: Verify Foreign Key name in DB before creating below insert queries**

**Insert into orders (orderId, orderDescription , user\_id) values(2001, ‘order11’, 101);**

**Insert into orders (orderId, orderDescription , user\_id) values(2002, ‘order12’, 101);**

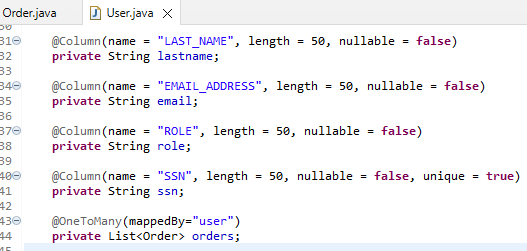
**Insert into orders (orderId, orderDescription , user\_id) values(2003, ‘order13’, 101);**

**Insert into orders (orderId, orderDescription , user\_id) values(2004, ‘order21’, 103);**

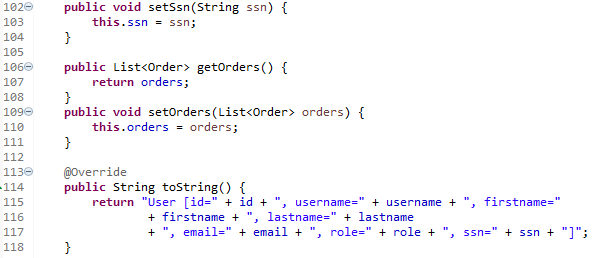
**Insert into orders (orderId, orderDescription , user\_id) values(2005, ‘order22’, 102);**

**Insert into orders (orderId, orderDescription , user\_id) values(2006, ‘order31’, 103);**

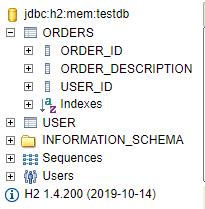
**add mapping( mappedBy = “user”) in the User Entity for the orders in the Order Entity:**



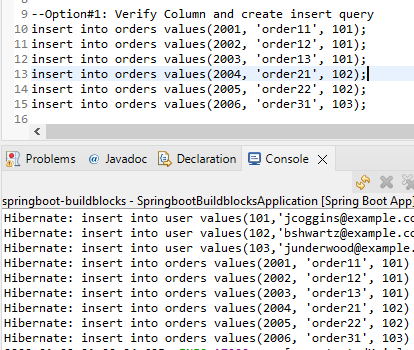
**Add the Getters and Setters for orders in User Entity:**



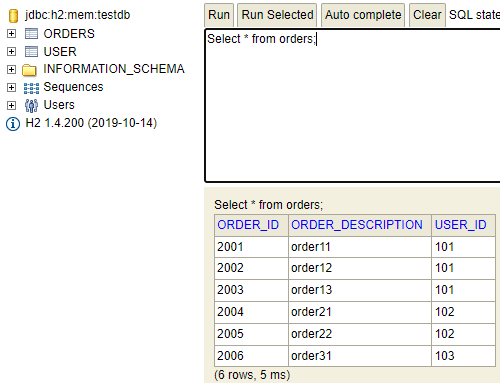
**Start up SpringBoot app and check tables:**



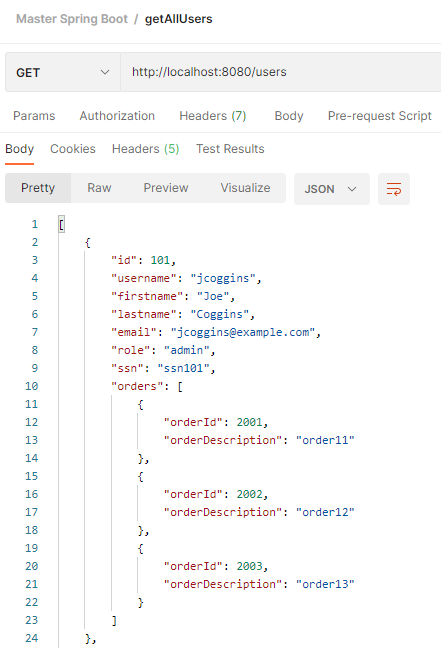
**Add data to the tables (Option#1):**



**Use a Select Statement to check the database:**



**Test with Postman:**

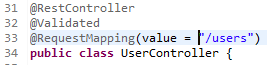


**Step-04: Implement “getAllOrders” method in OrderController**

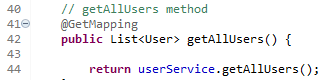
* **Controller Layer: UserController**
  + **Add @RequestMapping at class level and add “/users” context at class level**
  + **Remove “/users” at method level for all User related methods.**
  + **This will help us when creating self links in HATEOS section.**
* **Controller Layer: OrderController**
  + **Annotate with @RestController**
  + **Annotate with @RequestMapping**
  + **Method: getAllOrders**
  + **GET /users/{userid}/orders**
* **Test using Postman**
  + **Test#1 getAllOrders**
    - **GET /users/101/orders**
  + **Test#2 getAllUsers**
    - **GET /users**
  + **Test#3 getUserById**
    - **GET /users/101**

**Fix context mappings in UserController:**

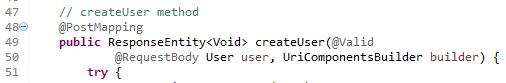
**Add at the class level:**



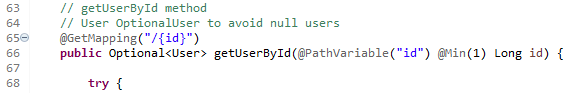
**Remove /users:**



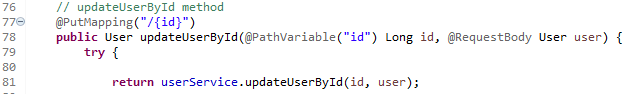
**Remove /users:**



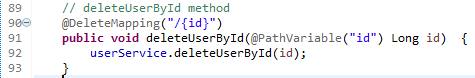
**Remove /users:**



**Remove /users:**



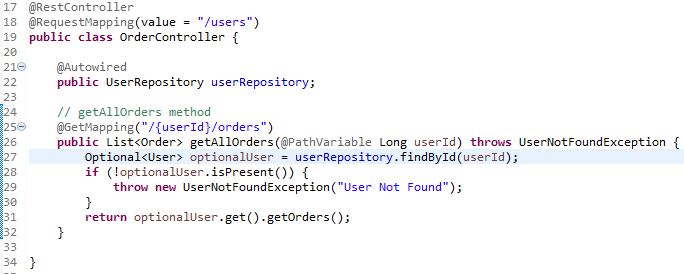
**Remove /users:**



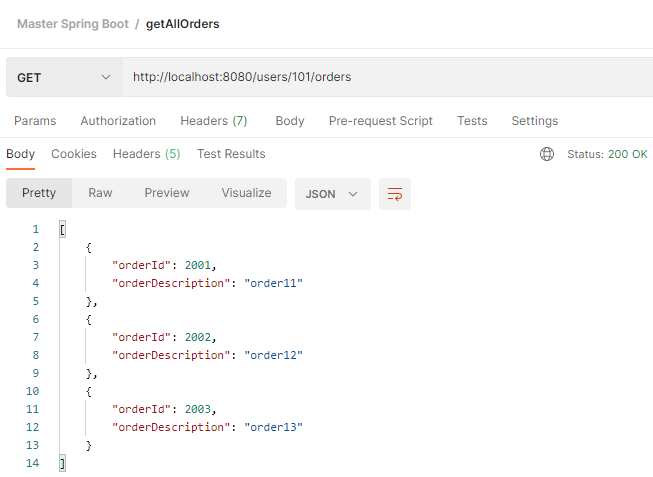
**Remove /users:**



**Add the Rest annotations to OrderController:**



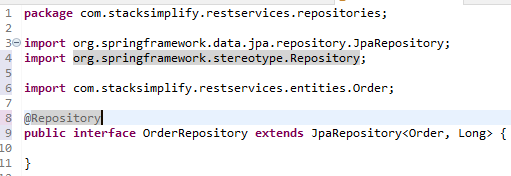
**Test with Postman:**



**Step-05: Implement “createOrder” method in OrderController**

* **Repository Layer**
  + **Create OrderRepository**
* **Controller Layer: OrderController**
  + **Method: createOrder**
  + **POST /users/{userId}/orders**
* **Test Using Postman**
  + **Test#1 createOrder**
    - **POST /users/101/orders**
  + **Test#2 getAllOrders**
    - **GET /users/101/orders**
  + **Test#3 getAllUsers**
    - **GET /users**
  + **Test#4 getUserById**
    - **GET /users/101**

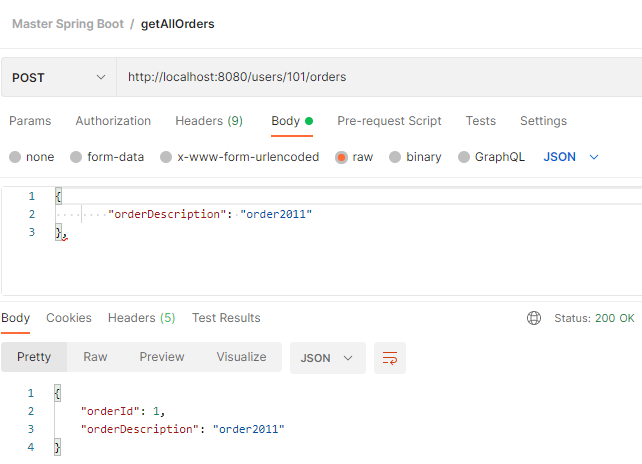
**Create an OrderRepository interface:**



**Create an OrderController class:**

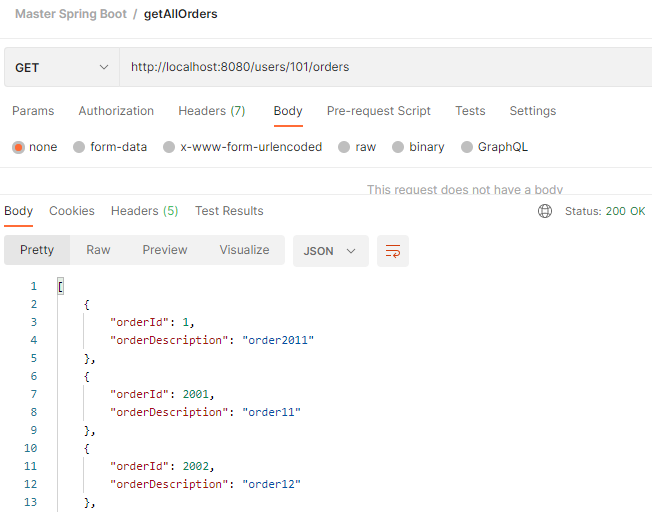


**Test with Postman:**

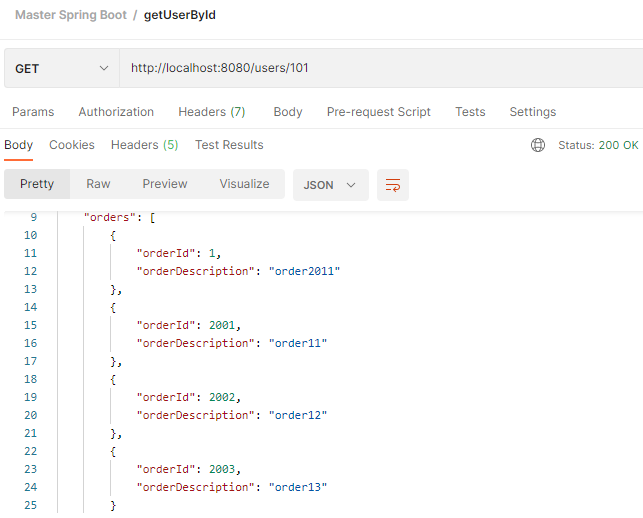


**Order created:**

**Check with getAllOrders**



**Test postman with getUserById**

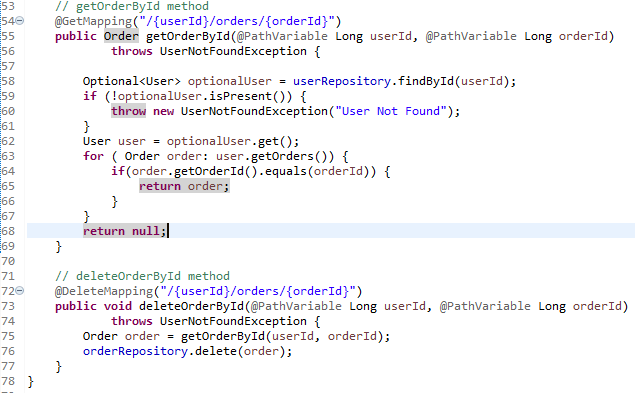


**Order id “1” appears**

**Step-06: Implement “getOrderByOrderId” method in OrderController**

* **Controller Layer: OrderController**
  + **Method: getOrderByOrderId**
  + **GET /users/{userId}/orders/{orderId}**
* **Test using POSTMAN**
  + **Test#1: getOrderByOrderId**
    - **GET /users/{userId}/orders/{orderId}**

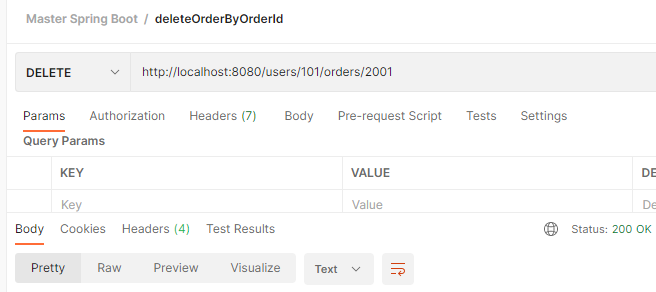
**Create getOrderByOrderId and deleteOrderById:**



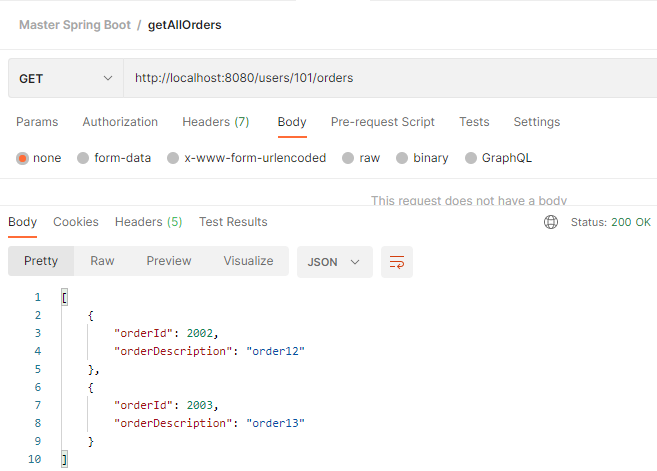
**Test with POSTMAN getOrderByOrderId**



**Test with POSTMAN deleteOrderById and check with getAllOrders for that user:**

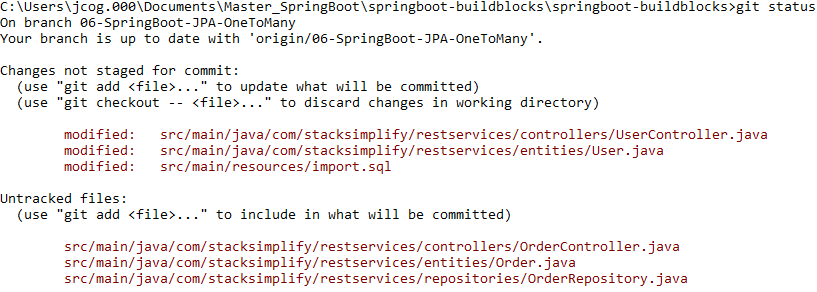


**getAllOrders for that user is missing orderId 2001:**

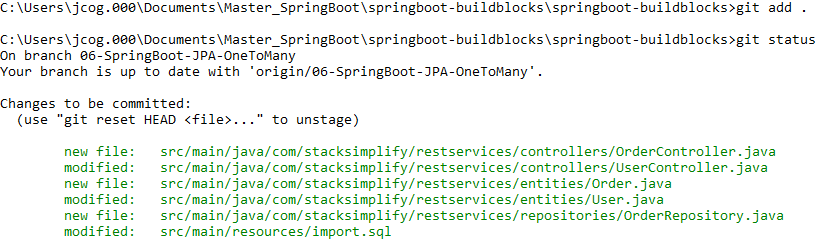


**Step-07: GIT commit code, push to remote, merge to master and push to remote**

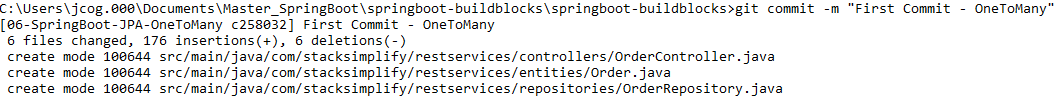
* **git status**



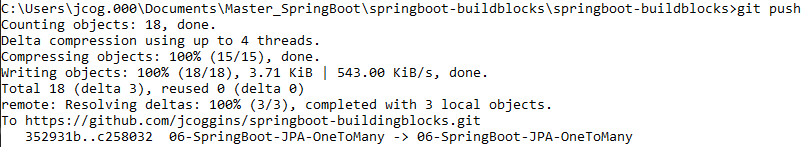
* **git add . git status**



* **git commit –m ”First Commit - OneToMany”**



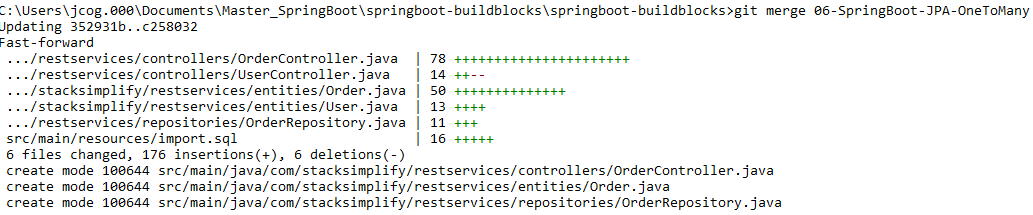
* **git push**



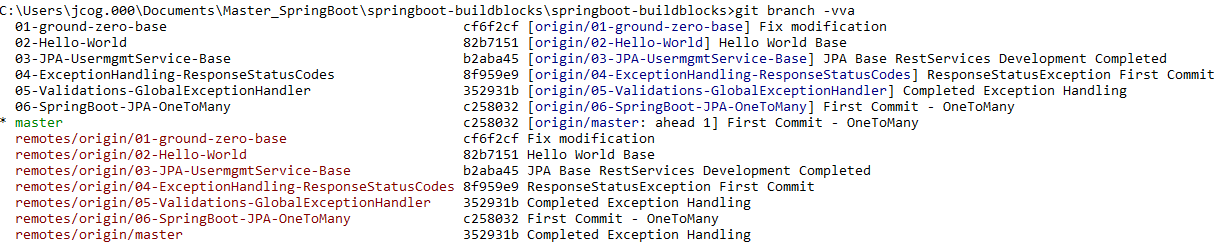
* **git checkout master**



* **git merge 06-SpringBoot-JPA-ONeToMany**



* **git branch –vva**



**To fix the ahead 1) do git push while on master**

**git push git branch -vva**

