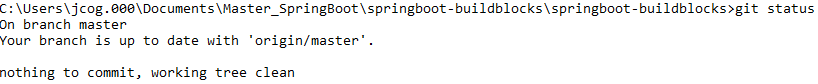
**Master SpringBoot**

**08-SpringBoot Filtering:**

**Step-00: Create new GIT branch for Filtering**

**git checkout –b 09-01-SpringBoot-Filtering-JsonIgnore**

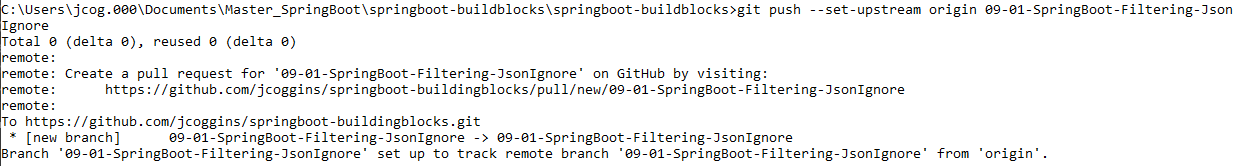
**git status**



**git checkout –b 09-01-SpringBoot-Filtering-JsonIgnore**



**git push –set-upstream origin 09-01-SpringBoot-Filtering-JsonIgnore**



**git branch –a**

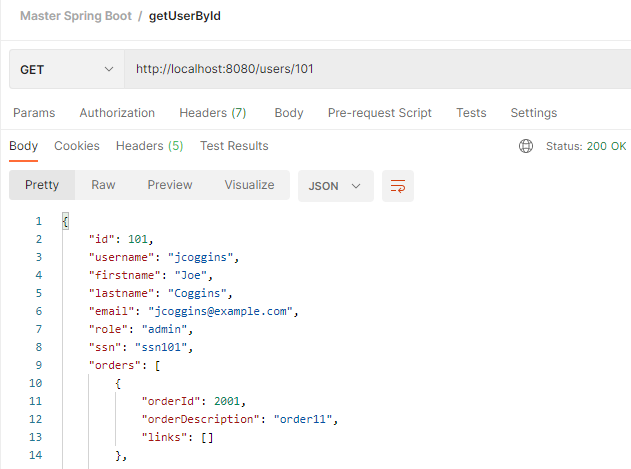


**Local branch and remote branch are the same**



**Test using POSTMAN, JSON response before changes to User.java entity:**

**Notice the ssn is displayed:**



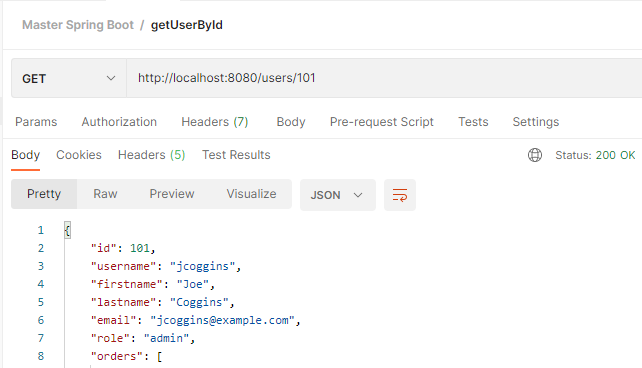
**Step-01: Static Filtering**

* **Entity Layer**
  + **@JsonIgnore – Apply to SSN field**
  + **Test**
  + **@JsonIgnoreProperties – Apply to role and email**
  + **Test**

**@JsonIgnore – Apply to SSN field**



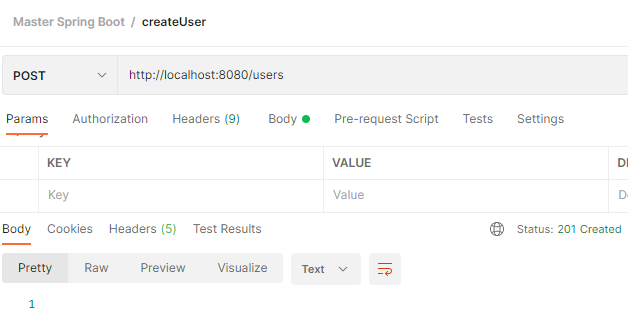
**Test using Postman: Notice ssn is filtered out**



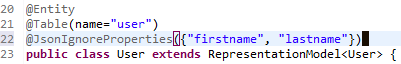
**createUser will fail if your input contains ssn, so nullable must be set to true**



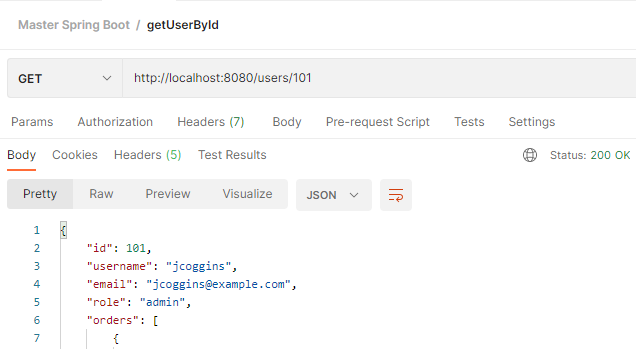
**createUser will succeed now: status 201 Created**



**Let us ignore properties at the class level:**



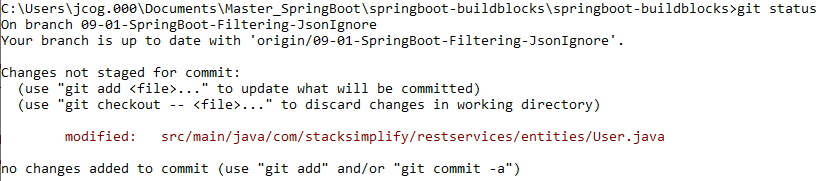
**Test using Postman: Notice ssn is filtered out and so are the first and last names**



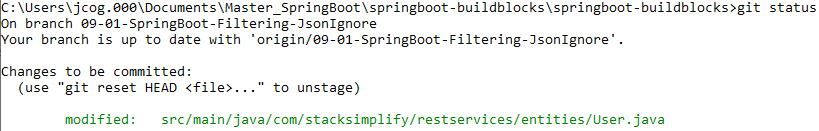
**Step-02: Commit Code**

* **Commit code**
* **Push branch to Remote repo**

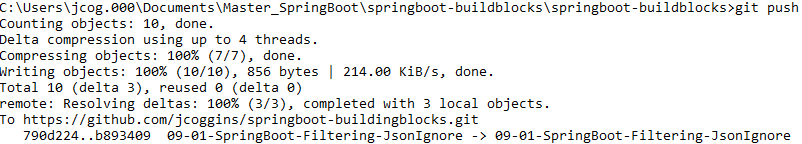
**git status git add . git status git commit –m “First Commit JsonIgnore” git push**











**git branch –vva**



**Step-00: Dynamic Filtering**

* **Step-01: Git branch**
* **git branch name: GIT branch 09-02-SpringBoot-Filtering-MappingJacksonValue**
* **Create new local branch**

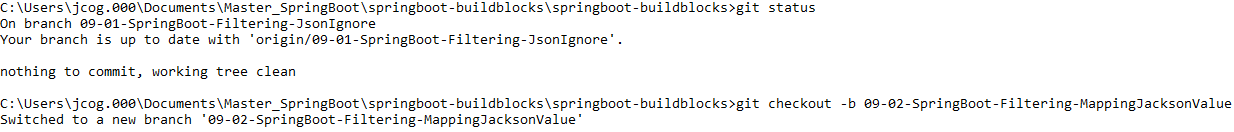
**Step-02: Dynamic Filtering using User MappingJacksonValue class**

* **Controller layer**
  + **Create a new UserMappingJacksonController**
  + **Copy getUserbyId methods**
  + **UserMappingJacksonController**
* **Implement filtering with statis hashset**
  + **Test it**
  + **Convert the same to a dynamic fitering by using @RequestParam**
  + **Test it**

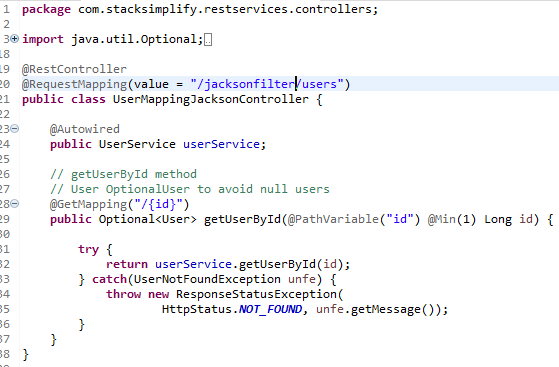
**Step-03: Commit Code**

* **Commit code**
* **Push branch to Remote repo**

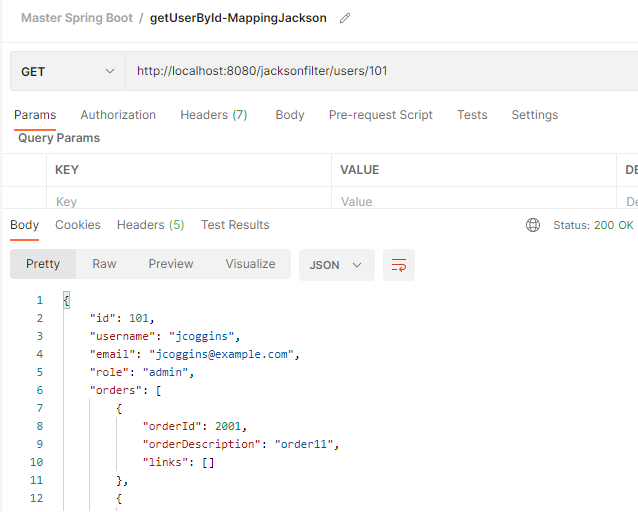
**git status git checkout …**



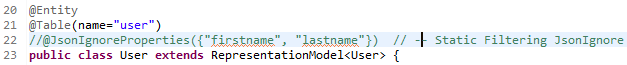
**Create the UserMappingJacksonController class:**



**Create the Postman test method:**



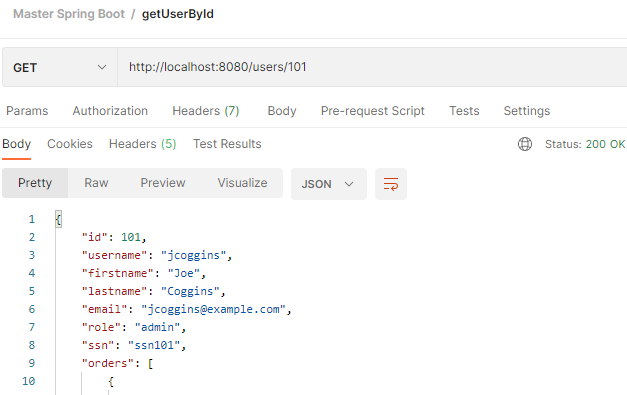
**Comment out @JsonIgnoreProperties annotation:**



**Comment out @JsonIgnore annotation**



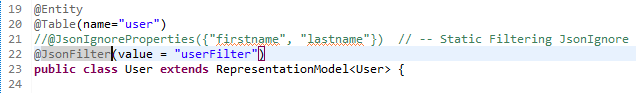
**Test using Postman: firstname, lastname and ssn are back**



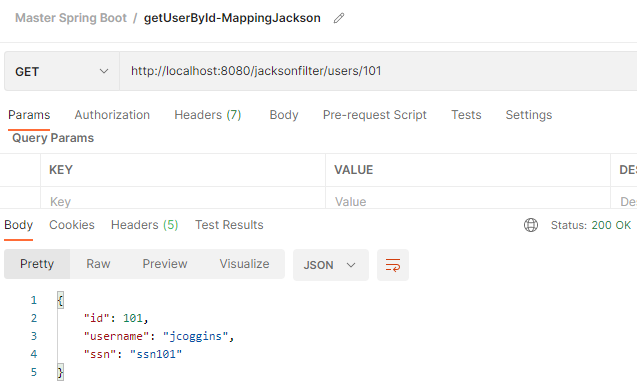
**In the UserMappingJacksonController:**



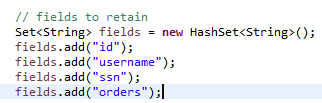
**Add annotation to User Entity:**



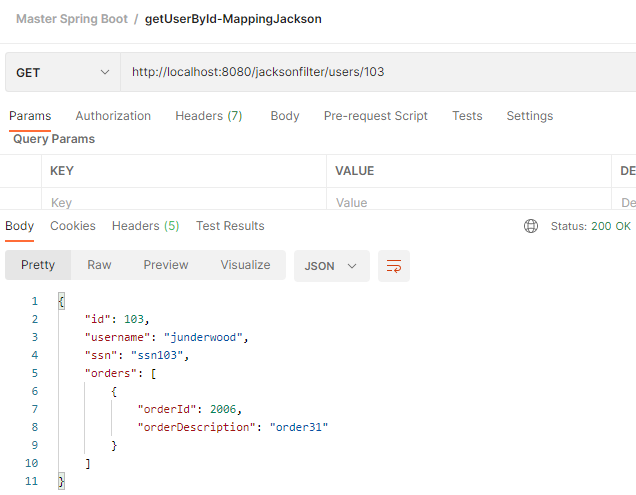
**Test using Postman:**



**Add orders**



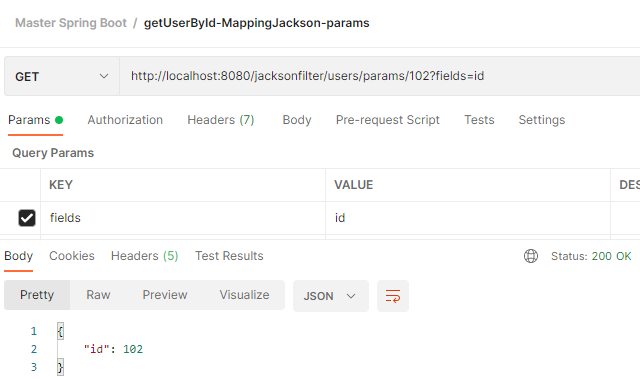
**Test using Postman**



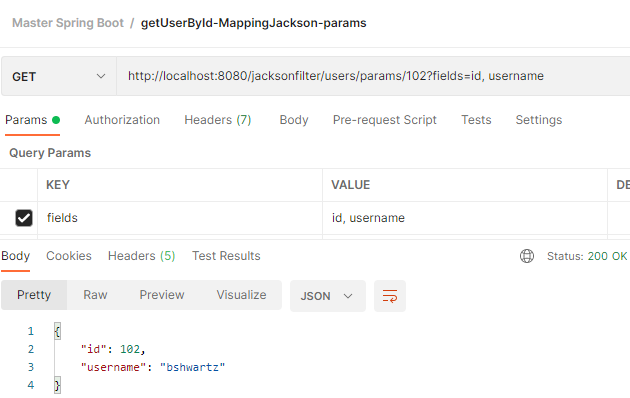
**Adding Params with key value pairs to filter is next (Dynamic Params)**



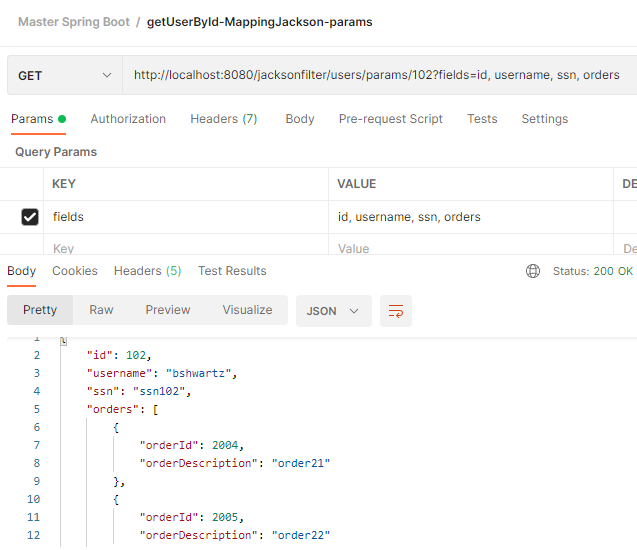
**Test using Postman:**



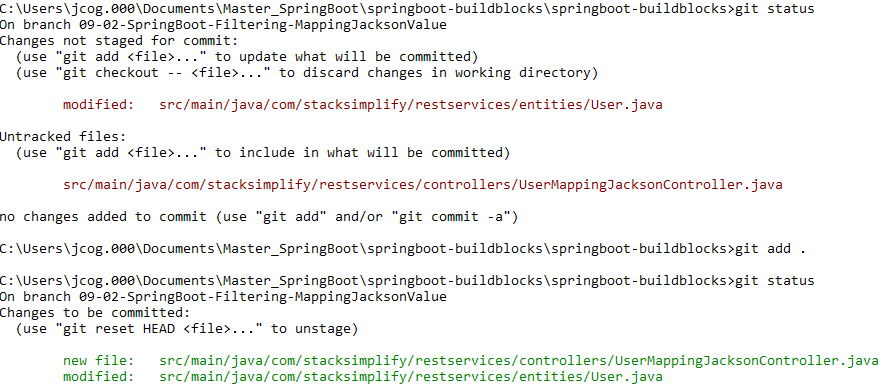
**Try id and username:**



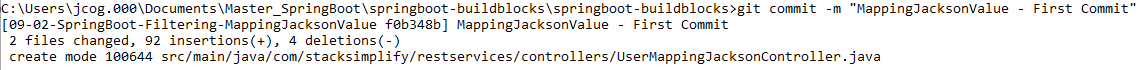
**Try with ssn and orders:**



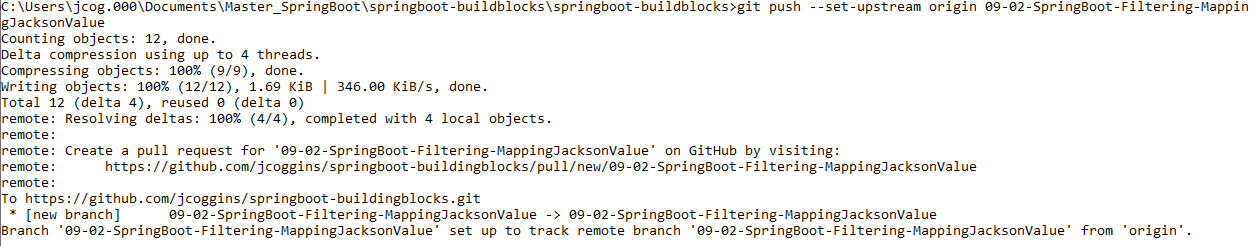
**git status git add . git status**



**git commit –m “MappingJacksonValue – First Commit”**



**git push –set-upsteam origin 09-02-SpringBoot-Filtering-MappingJacksonValue**



**git branch –vva**



**JsonView Filtering:**

**Step-00: Introduction**

**Step-01: New GIT branch (using IDE)**

* **git Branch name: 09-03-SpringBoot-Filtering-JsonView**
* **Create new local branch**

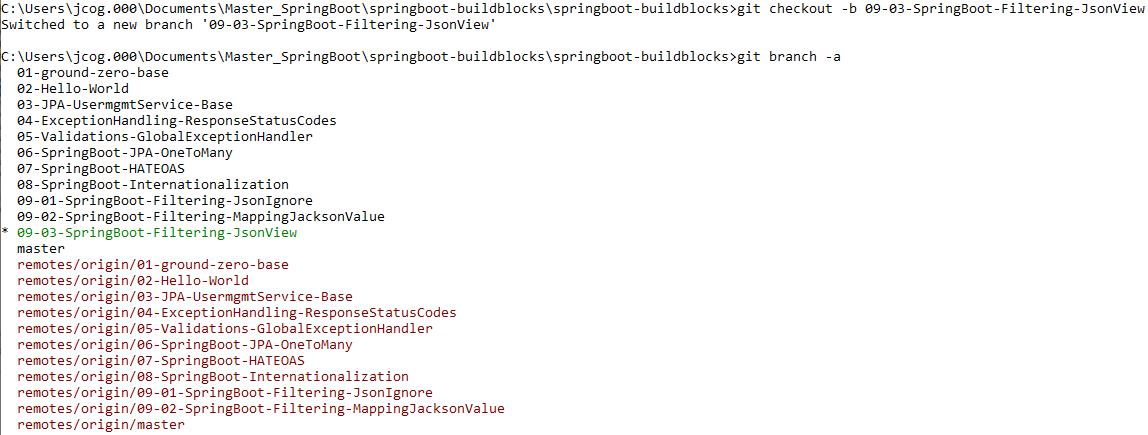
**Step-02: @JsonView**

* **Entity Layer**
  + **Create a class named Views**
    - **Create 2 static classes in View (External, Internal)**
  + **Annotate fields in User Entity with @JsonView**
  + **Decide which fields should be external and internal and annotate accordingly**
* **Controller Layer**
  + **Create new controller named “UserJsonViewController”**
  + **Copy getUserById method from UserController and create 2 methods**
    - **External getUserById**
      * **GET /jsonview/users/external/101**
      * **Annotate with @JsonView(Views.External.class)**
      * **Test it using Postman**
    - **Internal getUserById2**
      * **GET /jsonview/users/external/101**
      * **Annotate with @JsonView(Views.Internal.class)**
      * **Test it using Postman**

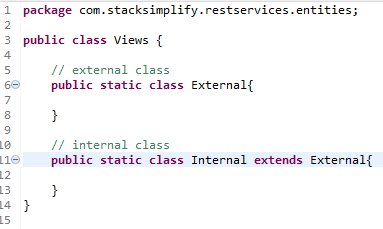
**Step-03: Commit Code (using IDE)**

* **Commit code**
* **Push branch to Remote repo**

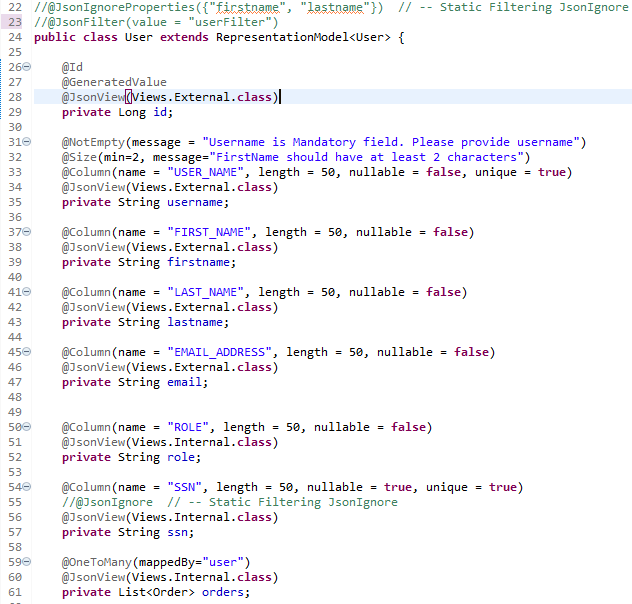
**git checkout … git branch –a**



**Create class Views:**



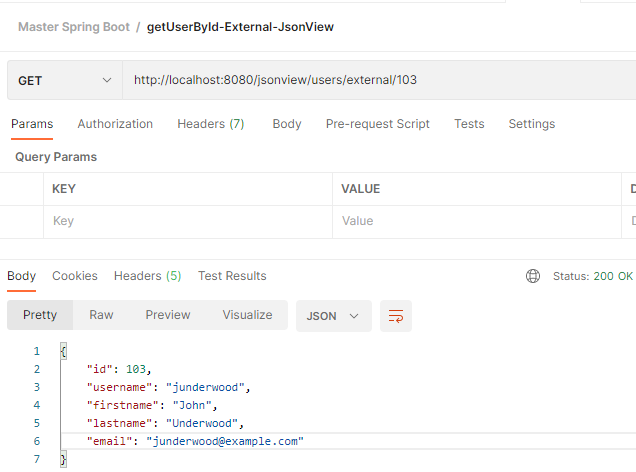
**Edit User entity class – annotate properties as External or Internal:**



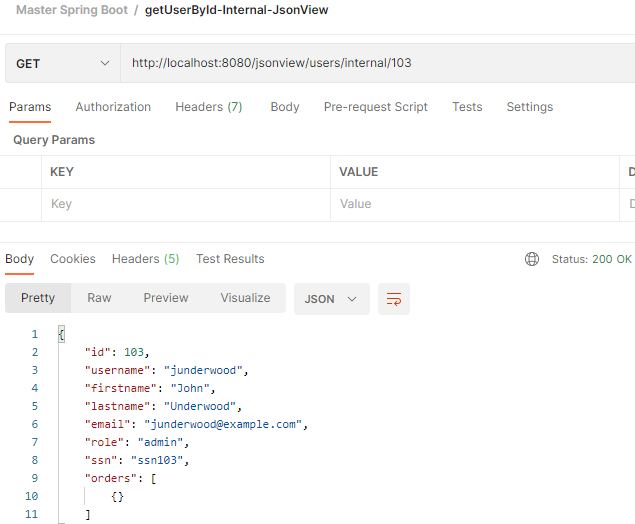
**Create UserJsonViewController:**



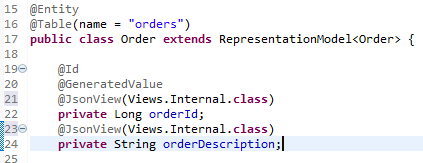
**Test using postman: External View**



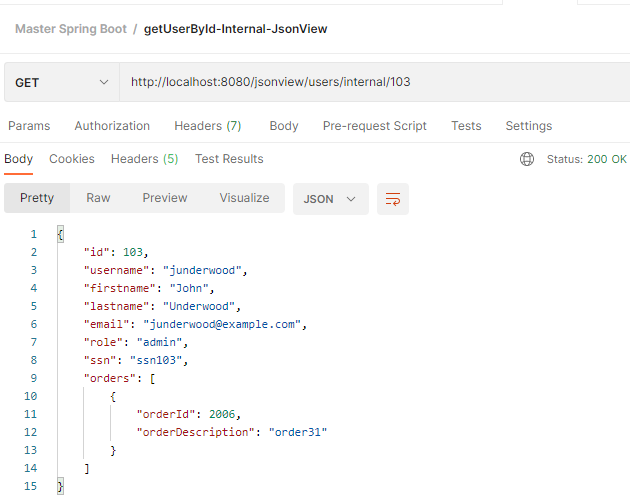
**Test using postman: Internal View**

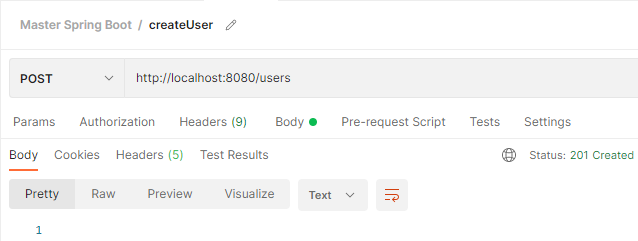


**We have to fix the view for orders:**

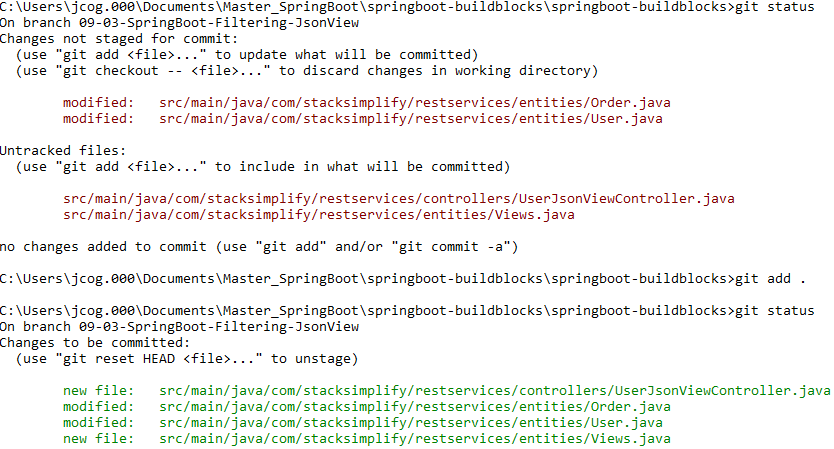


**Test with postman: Internal View**

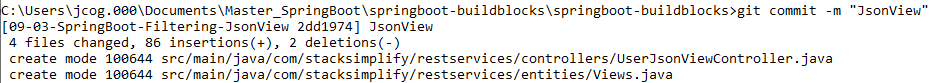
**You can use createUser successfully with JsonView!! Status 201 Created!**



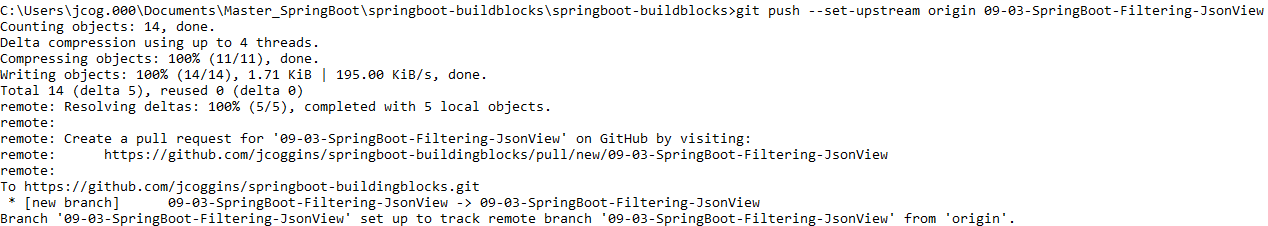
**git status git add . git status**



**git commit –m “JsonView”**



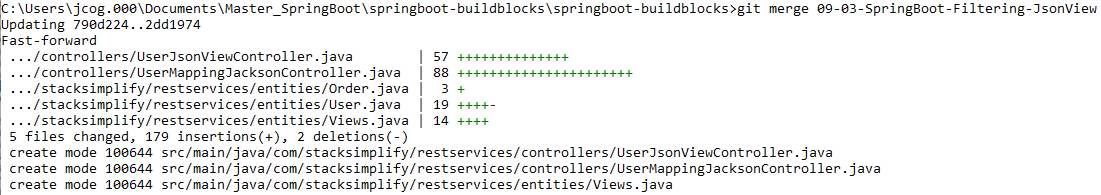
**git push –set-upsteam origin 09-03-SpringBoot-Filtering-JsonView**



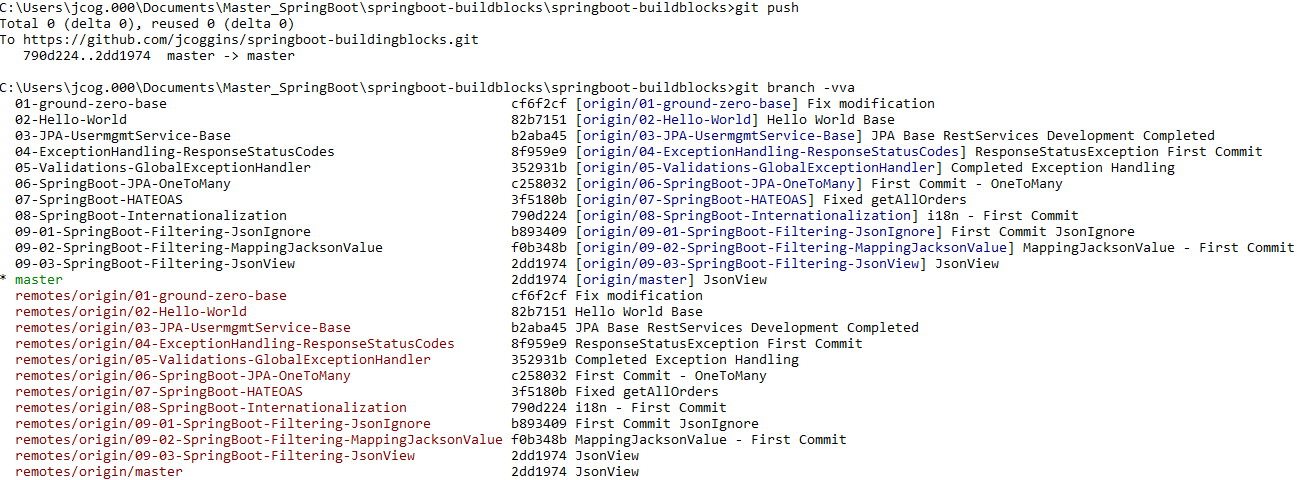
**git checkout master**



**git merge 09-03-SpringBoot-Filtering-JsonView**



**git push git branch –vva**



**GUI Git Branches**



**Done!!**