Spring Boot Ajax Call

<https://grokonez.com/java-integration/integrate-jquery-ajax-post-get-spring-boot-web-service>

Swagger Documentation :

This ApplicationConfig file to be added in Controller

package com.javasampleapproach.jqueryajax.controller;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

import springfox.documentation.builders.PathSelectors;

import springfox.documentation.builders.RequestHandlerSelectors;

import springfox.documentation.spi.DocumentationType;

import springfox.documentation.spring.web.plugins.Docket;

import springfox.documentation.swagger2.annotations.EnableSwagger2;

@Configuration

@EnableSwagger2

public class ApplicationConfig {

@Bean

public Docket api() {

return new Docket(DocumentationType.SWAGGER\_2)

.select()

.apis(RequestHandlerSelectors.basePackage("com.javasampleapproach.jqueryajax.controller"))

.paths(PathSelectors.any())

.build();

}

}

Pom.xml

<dependency>

<groupId>io.springfox</groupId>

<artifactId>springfox-swagger2</artifactId>

<version>2.7.0</version>

</dependency>

<dependency>

<groupId>io.springfox</groupId>

<artifactId>springfox-swagger-ui</artifactId>

<version>2.7.0</version>

</dependency>

https://editor.swagger.io/#

http://localhost:8080/v2/api-docs

<http://localhost:8080/swagger-ui.html#/>

Logger

Dependency

<dependency>

<groupId>log4j</groupId>

<artifactId>log4j</artifactId>

<version>1.2.17</version>

</dependency>

# Root logger option

log4j.rootLogger=DEBUG, stdout, file

log4j.logger.infoLogger=DEBUG, infoLogger

log4j.additivity.infoLogger = false

# Redirect log messages to console

log4j.appender.stdout=org.apache.log4j.ConsoleAppender

log4j.appender.stdout.Target=System.out

log4j.appender.stdout.layout=org.apache.log4j.PatternLayout

log4j.appender.stdout.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c**{1}**:%L - %m%n

# Redirect log messages to a log file, support file rolling.

log4j.appender.file=org.apache.log4j.RollingFileAppender

log4j.appender.file.File=E:\\LOG\\ConvertorLogger.log

log4j.appender.file.MaxFileSize=5MB

log4j.appender.file.MaxBackupIndex=10

log4j.appender.file.layout=org.apache.log4j.PatternLayout

log4j.appender.file.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c**{1}**:%L - %m%n

log4j.rootLogger = DEBUG, DB

log4j.appender.DB=org.apache.log4j.jdbc.JDBCAppender

# Set JDBC URL

log4j.appender.DB.URL=jdbc:postgresql://localhost:5432/postgres

# Set Database Driver

log4j.appender.DB.driver=org.postgresql.Driver

# Set database user name and password

log4j.appender.DB.user=postgres

log4j.appender.DB.password=root

# Set the SQL statement to be executed.

log4j.appender.DB.layout.ConversionPattern=INSERT INTO LOGS VALUES('%x',now(),'%C','%p','%m')

# Define the layout for SQL appender

log4j.appender.DB.layout=org.apache.log4j.EnhancedPatternLayout

CREATE TABLE LOGS (USER\_ID VARCHAR(20) NOT NULL,DATED DATE NOT NULL, LOGGER VARCHAR(50) NOT NULL,LEVEL VARCHAR(10) NOT NULL,MESSAGE VARCHAR(1000) NOT NULL);

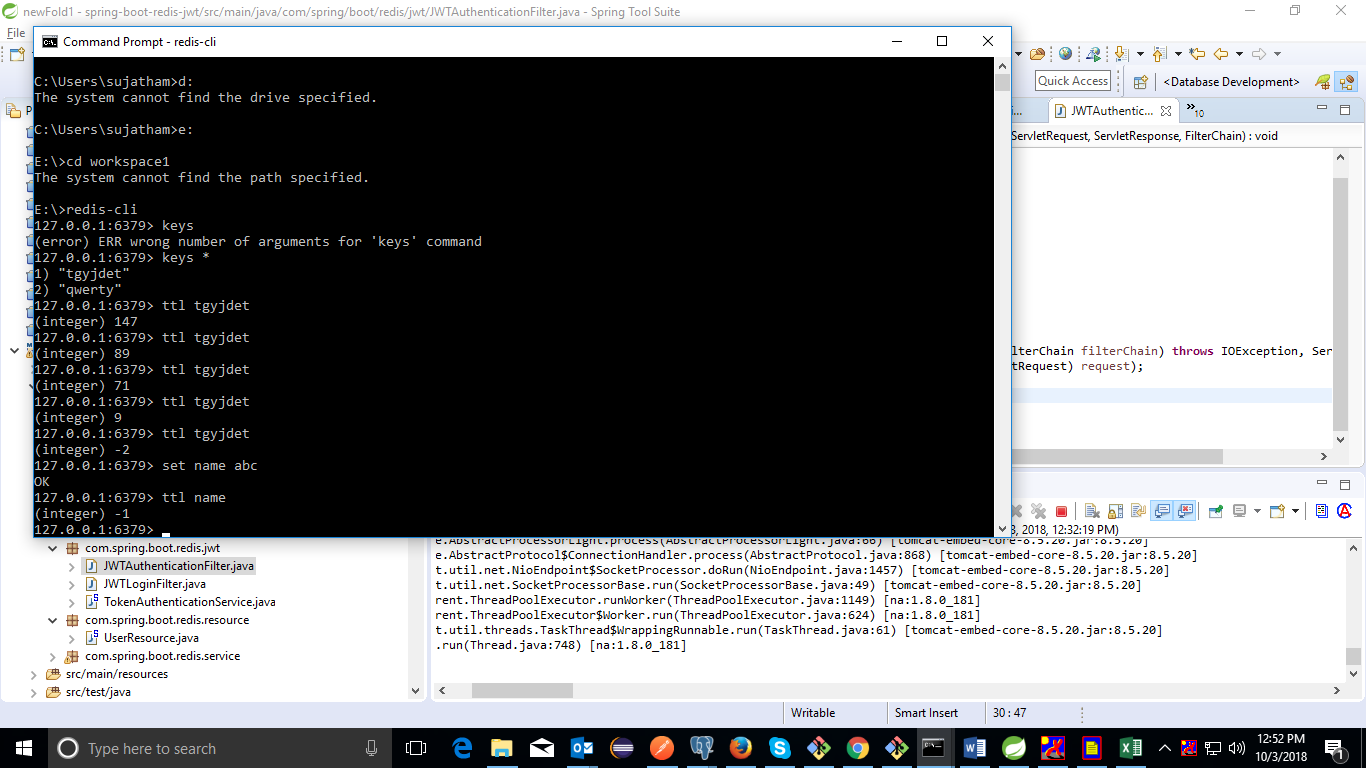
Redis

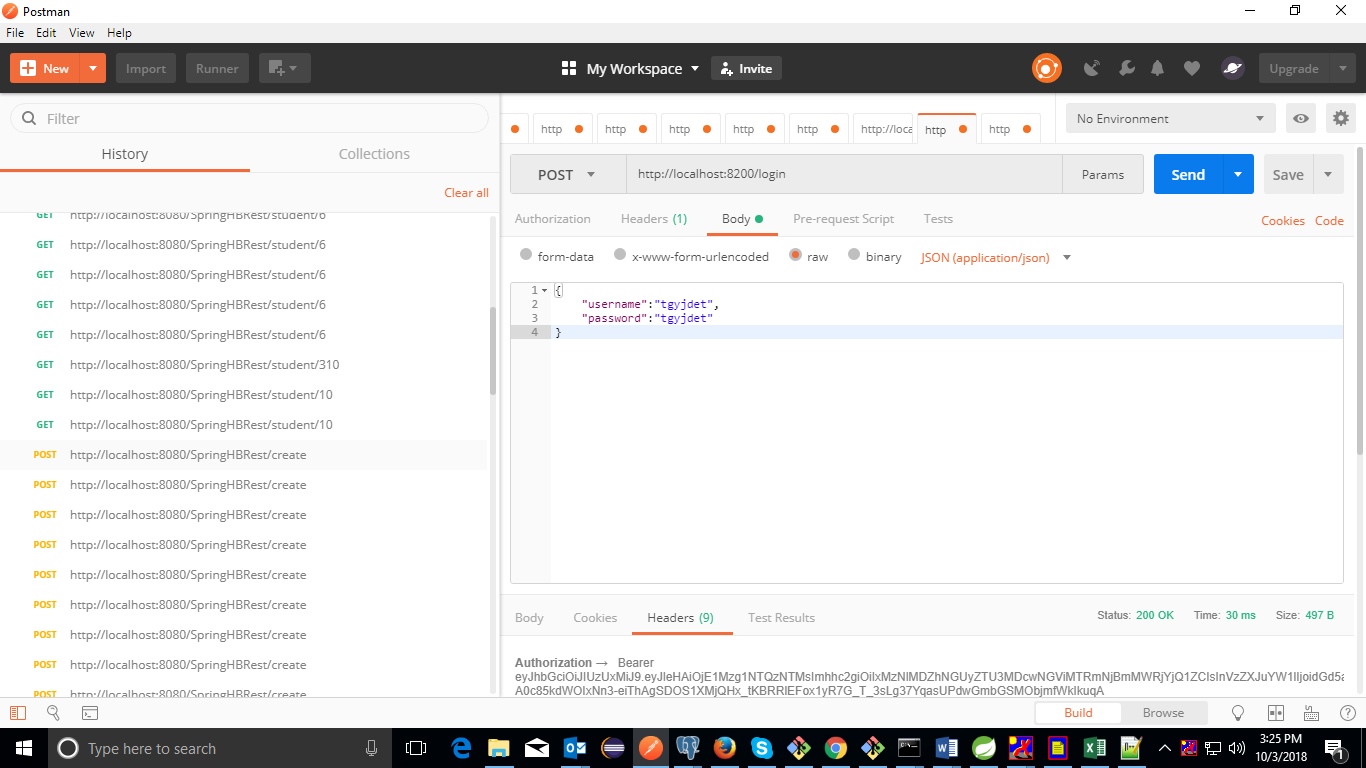
=========

<https://github.com/privatejava/spring-boot-redis-j>

Redis -you tube For Installation

<https://www.youtube.com/watch?v=ncFhlv-gBXQ>





Refer : Sample code

<https://blog.ngopal.com.np/2017/10/10/spring-boot-with-jwt-authentication-using-redis/>

Read About Lombok

<https://howtodoinjava.com/automation/lombok-eclipse-installation-examples/>

C:\Users\sujatham\.m2\repository\org\projectlombok\lombok\1.16.18\lombok.jar

Double click the jar it ll install Lombok tool

Client and Server Connect Spring With JSP Client :

-------------------------------------------------------------------------------------------

UserPortel-Client to be run in Eclipse

user-portel [boot] -server -run in sts

E:\Newfolder3(Default Port no of Wild fly is 8080 so 9990 to be ignored )

http://localhost:8080/UserPortal/add