Connectivity Dilemma :\_\_\_\_\_\_\_\_

1. Get Request

* React js snippet to send get request through axion kindly understand corresponding to angular get request.

var objToSend={

key:"value",

key1:"value1",

key2:"value2",

key3:[{name:"anurag"},{name:"virag"}]

};

//objToSend = JSON.stringify(objToSend);

async function getDataAxios(){

const response = await axios.get("http://localhost:8082/Settlement\_Systems/sendjson",{params:{

data: objToSend

}});

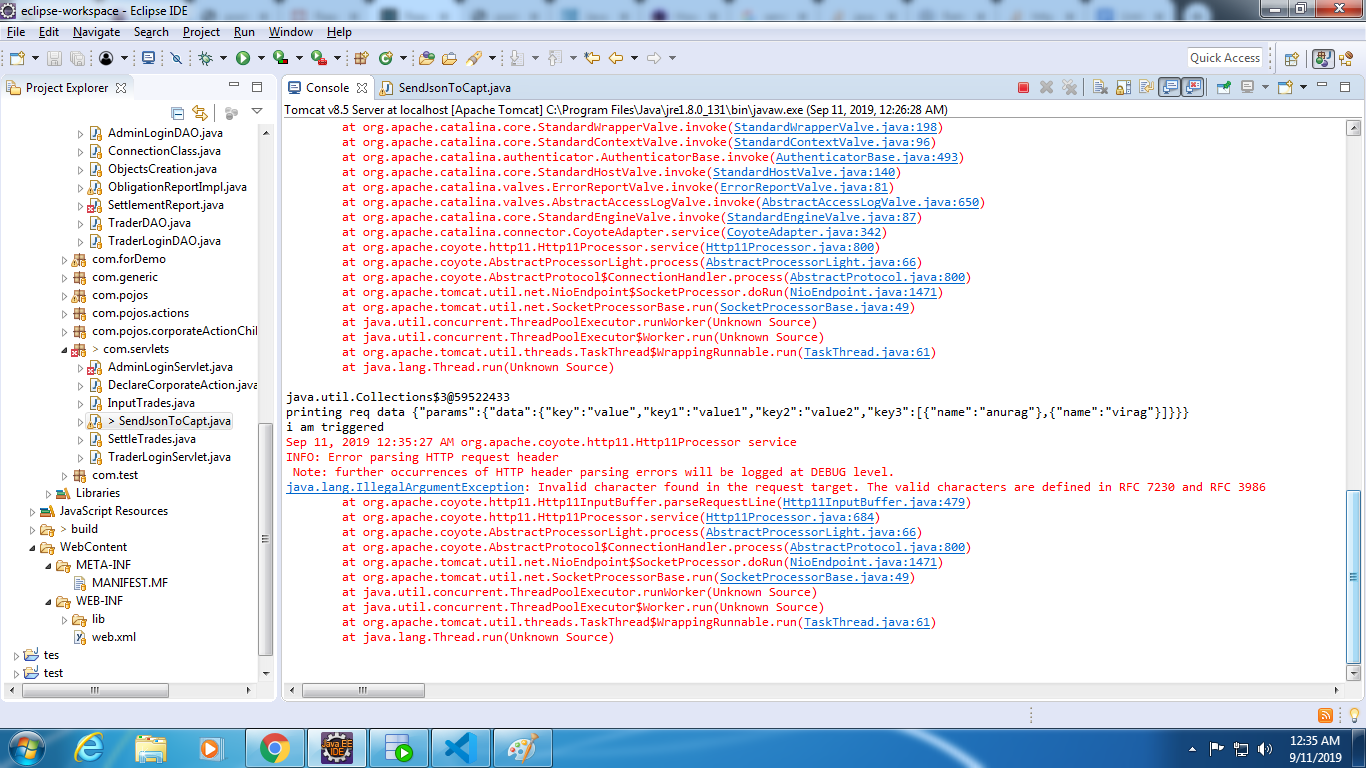
console.log(response);

}

This didn’t work with illegal argument exception in servlet. Although the same argument runs on postman-echo.com/get…….

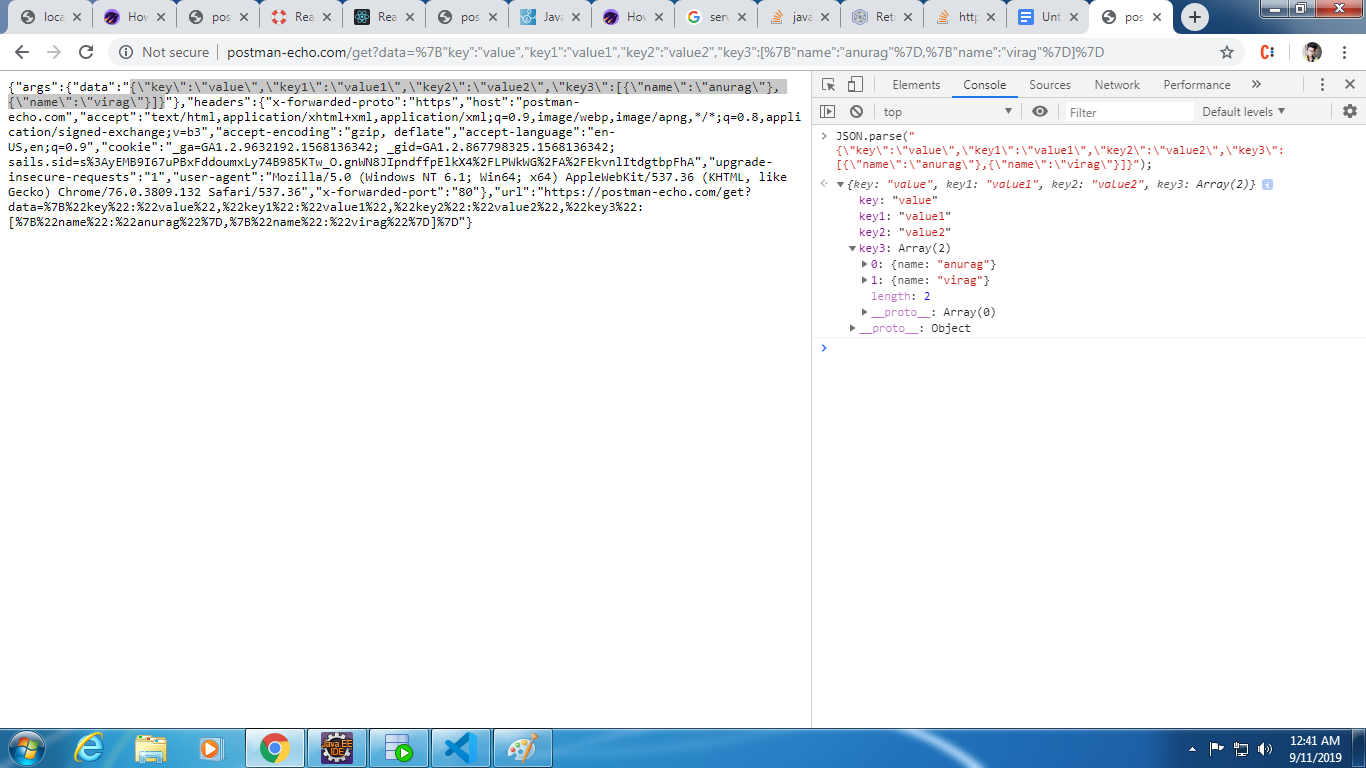
**Result 1**

ILLEGAL ARGUMENT EXCEPTION



**Result 2**

Request Successful



Exactly same happens when the javascript object is converted to string with JSON.stringify();

1. Post Request

* React js snippet to send get request through axion kindly understand corresponding to angular get request.

var objToSend={

key:"value",

key1:"value1",

key2:"value2",

key3:[{name:"anurag"},{name:"virag"}]

};

//objToSend = JSON.stringify(objToSend);

async function getDataAxios(){

const response = await axios.post("http://localhost:8082/Settlement\_Systems/sendjson",{params:{

data: objToSend

}});

console.log(response);

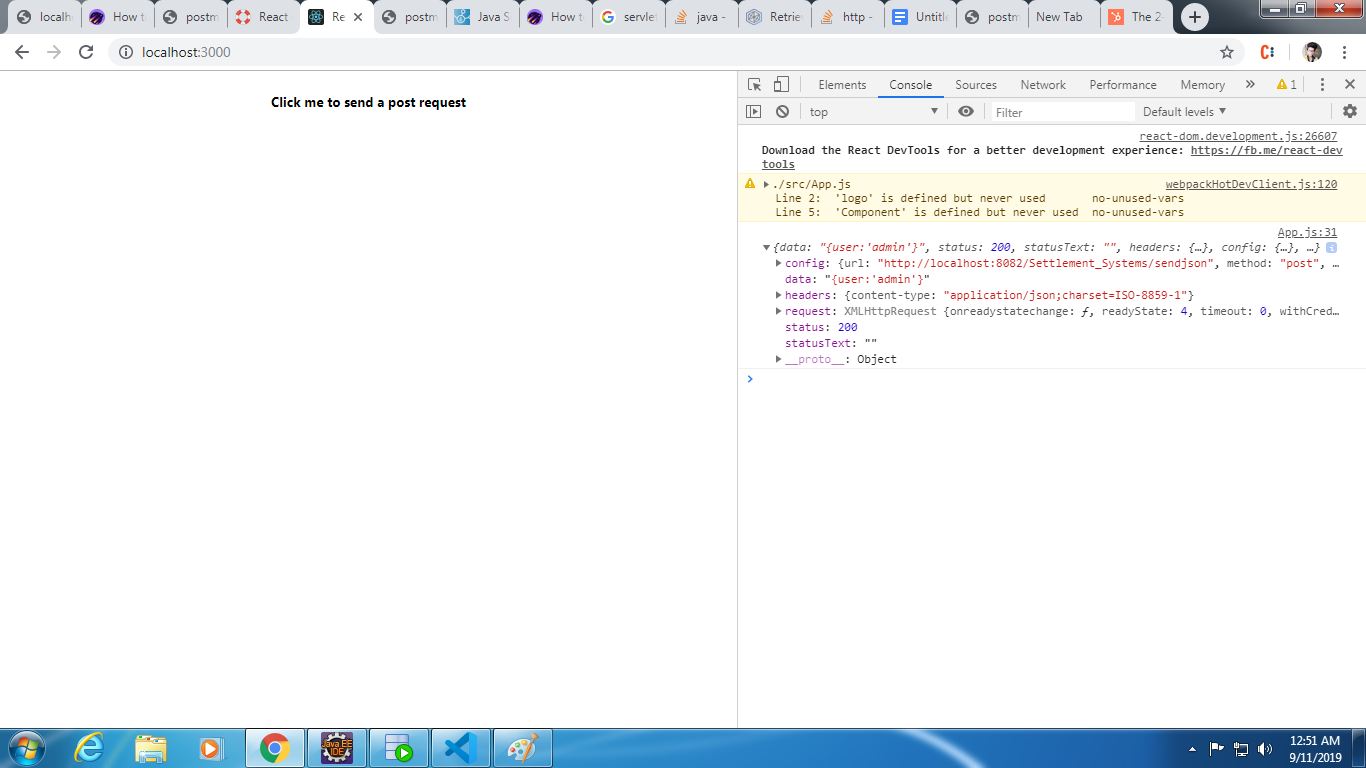
}

This runs very well on our servlet and the object sent can be seen to be printed.

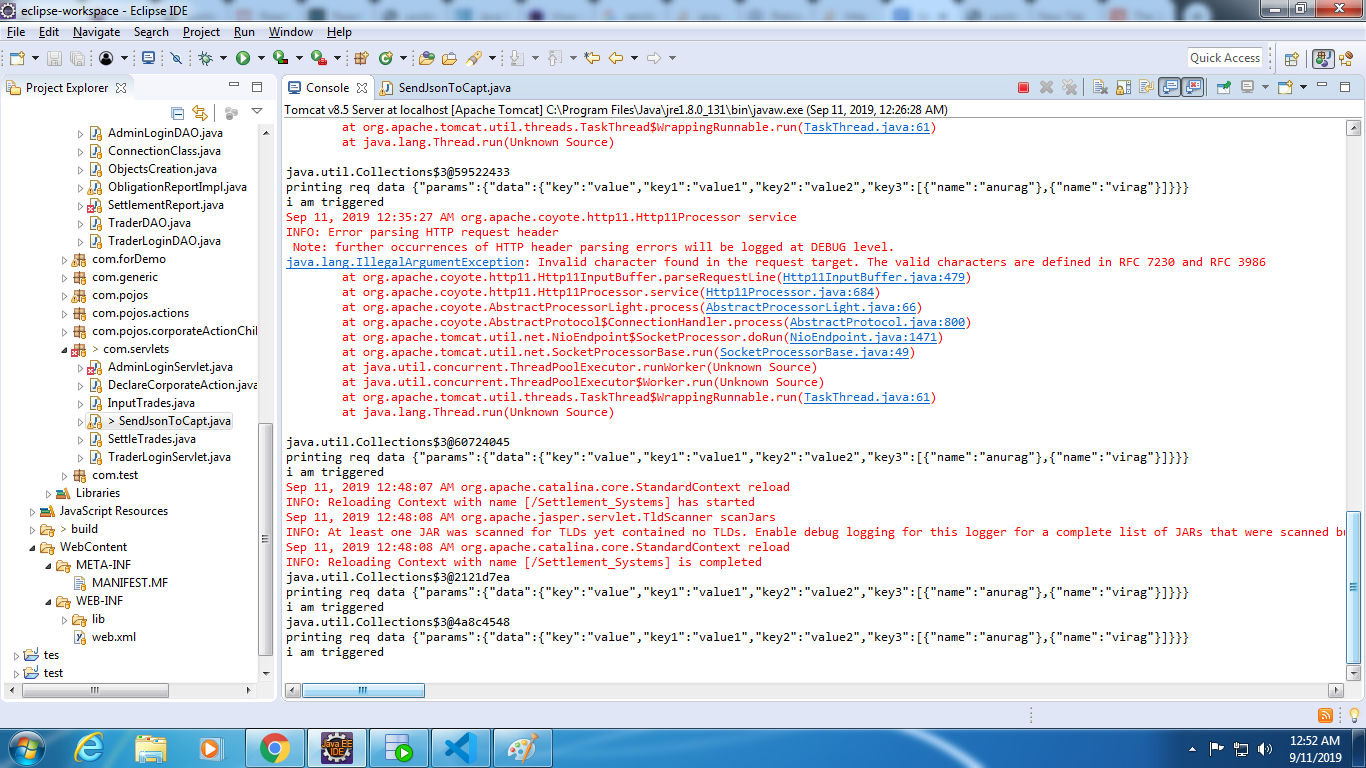
**Result**

Successful

Client side



Servlet Side



Servlet file just for reference:

package com.servlets;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.Enumeration;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

@WebServlet(name = "SendJsonToCapt", urlPatterns = "/sendjson")

public class SendJsonToCapt extends HttpServlet {

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

String username = req.getParameter("username");

String password = req.getParameter("password");

String data = req.getParameter("data");

System.out.println(data);

PrintWriter out = resp.getWriter();

out = resp.getWriter();

resp.setContentType("application/json");

resp.setCharacterEncoding("UTF-8");

if(username.equals("admin") && password.equals("admin")) {

out.print("{user:'admin'}");

out.flush();

}

else {

out.print("{user:'false'}");

out.flush();

}

}

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

// TODO Auto-generated method stub

//super.doPost(req, resp);

PrintWriter out = resp.getWriter();

out = resp.getWriter();

resp.setContentType("application/json");

resp.setCharacterEncoding("UTF-8");

Enumeration<String> params = req.getParameterNames();

System.out.println(params.toString());

while(params.hasMoreElements()){

String paramName = params.nextElement();

System.out.println("Parameter Name - "+paramName+", Value - "+req.getParameter(paramName));

}

StringBuilder buffer = new StringBuilder();

BufferedReader reader = req.getReader();

String line;

while ((line = reader.readLine()) != null) {

buffer.append(line);

}

String data = buffer.toString();

System.out.println("printing req data "+ data);

System.out.println("i am triggered");

out.print("{user:'admin'}");

out.flush();

}

}

**Inference :**

* Simple object(key value pair only) can be sent through get request **in our servlet** complex objects can’ts be sent through get request not even in stringified format, as server throws illegalargumentexception.
* Complex objects can very well be sent through post request **in our servlet**  even if not stringified, stringified objects go at its best.