

## TASK-5 CODE

### QUESTION :

- 1- Create a path based servlet
- 2- when the user hits the path with parameter "session", then read that and create the random 6 letter word and return it .

### Description :

This servlet is designed to generate and return random strings based on specific request parameters. It uses the Apache Commons Lang library to generate random alphabetic, alphanumeric, and numeric strings. Depending on the request parameters, it either returns a JSON object, a random alphanumeric string, or a numeric string. If the parameters do not match the expected values, it returns a 400 Bad Request status with an error message.

### ANSWER:

#### SERVLET CODE FOR DISPALYING RANDOM 6 LETTER WORD

```
package com.adobe.aem.guides.demo.core.servlets;

import java.io.IOException;

import javax.json.Json;
import javax.json.JsonObjectBuilder;
import javax.servlet.Servlet;
import javax.servlet.ServletException;

import org.apache.commons.lang.RandomStringUtils;
import org.apache.sling.api.SlingHttpServletRequest;
import org.apache.sling.api.SlingHttpServletResponse;
import org.apache.sling.api.servlets.SlingAllMethodsServlet;
import org.osgi.service.component.annotations.Component;
```

```

@Component(service = Servlet.class,immediate = true,
property={
    "sling.servlet.paths=/bin/tas"
})

public class ServTask_1 extends SlingAllMethodsServlet{

    @Override

    protected void doGet(SlingHttpServletRequest req, SlingHttpServletResponse resp)
        throws ServletException, IOException {

        String sessionpar =req.getParameter("session");
        String numberpar =req.getParameter("random");

        resp.setContentType("application/json");

        String randomword =RandomStringUtils.randomAlphabetic(6);
        String alphword =RandomStringUtils.randomAlphanumeric(6);
        String numword =RandomStringUtils.randomNumeric(6);
        JsonObjectBuilder json=Json.createObjectBuilder();

        if("session".equals(sessionpar)&&"alpha".equals(numberpar)){
            json.add(randomword.toUpperCase(), numword);
            resp.getWriter().write(json.build().toString());

        }else if("session".equals(sessionpar)&&"alphaNum".equals(numberpar)){
            resp.getWriter().write(alphword.toUpperCase());
        }else if("session".equals(sessionpar)&&"numeric".equals(numberpar)){
            resp.getWriter().write(numword.toUpperCase());
        }else{

            // resp.sendError(SlingHttpServletResponse.SC_BAD_REQUEST."parameter'session'is
not found");

```

```
        resp.setStatus(SlingHttpServletResponse.SC_BAD_REQUEST);  
        resp.getWriter().write("parameter'session'is not found");  
    }  
}  
}
```

## CHECKING SERVLET BY USING PATH &PARAMETER

