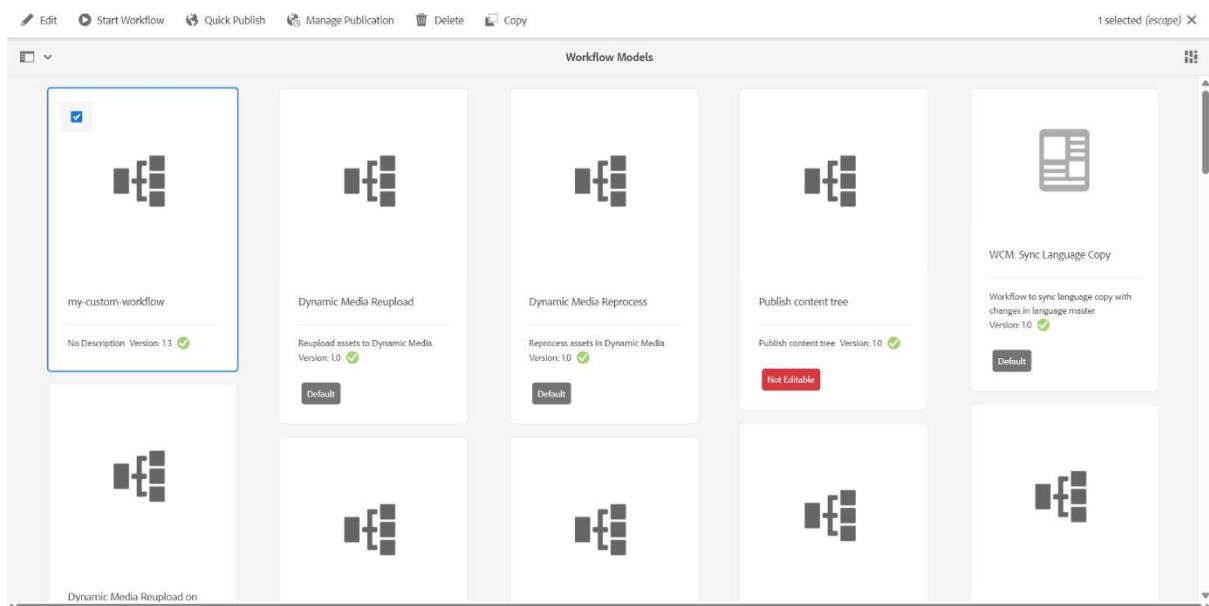


## AEM – Task 6

### Step 1: Create a Custom Workflow ("my custom workflow")

1. Go to AEM → Tools → Workflow → Models.
2. Click Create → Create Model, name it "my custom workflow".
3. Open the model editor and:
  - Drag the "Process Step" component onto the workflow.
  - Double-click it and set:
    - Process: Select "myCustomWorkflowProcess" (to be created in Step 2).
4. Save and activate the workflow



### Step 2: Create a Custom Workflow Process to Print Page Title in Logs

1. Implement a Java class extending WorkflowProcess.
2. Print the page title in logs using workflowSession.getMetaDataMap()
3. Deploy the bundle and configure the process step in myCustomWorkflow.

#### 4. Apply the workflow to a page and observe logs in AEM.

##### Java code

- Java code for WorkflowProcess.
- 

```
package com.myTraining.core.workflows;

import com.adobe.granite.workflow.WorkflowSession;
import com.adobe.granite.workflow.exec.WorkItem;
import com.adobe.granite.workflow.exec.WorkflowProcess;
import com.adobe.granite.workflow.metadata.MetadataMap;
import com.day.cq.wcm.api.Page;
import com.day.cq.wcm.api.PageManager;
import org.apache.sling.api.resource.Resource;
import org.apache.sling.api.resource.ResourceResolver;
import org.osgi.service.component.annotations.Component;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

@Component(
    service = WorkflowProcess.class,
    property = {
        "process.label=Custom Workflow Process"
    }
)

public class CustomWorkflowProcess implements WorkflowProcess {

    private static final Logger LOGGER =
        LoggerFactory.getLogger(CustomWorkflowProcess.class);

    @Override
```

```

public void execute(WorkItem workItem, WorkflowSession workflowSession, MetaDataMap
metaDataMap) {

    LOGGER.info("[Custom Workflow] Executing workflow process...");

    if (workItem == null || workItem.getWorkflowData() == null) {

        LOGGER.error("[Custom Workflow] WorkItem or WorkflowData is NULL!");

        return;

    }

    String payloadPath = workItem.getWorkflowData().getPayload().toString();

    LOGGER.info("[Custom Workflow] Payload Path: {}", payloadPath);

    // Get the Resource Resolver

    ResourceResolver resolver = workflowSession.adaptTo(ResourceResolver.class);

    if (resolver != null) {

        Resource resource = resolver.getResource(payloadPath);

        if (resource != null) {

            PageManager pageManager = resolver.adaptTo(PageManager.class);

            Page page = pageManager.getContainingPage(resource);

            if (page != null) {

                LOGGER.info("[Custom Workflow] Page Title: {}", page.getTitle());

                LOGGER.info("[Custom Workflow] Page Path: {}", page.getPath());

            } else {

                LOGGER.warn("[Custom Workflow] No Page found for the given resource.");

            }

        } else {

            LOGGER.warn("[Custom Workflow] No Resource found at the given path.");

        }

    } else {

```

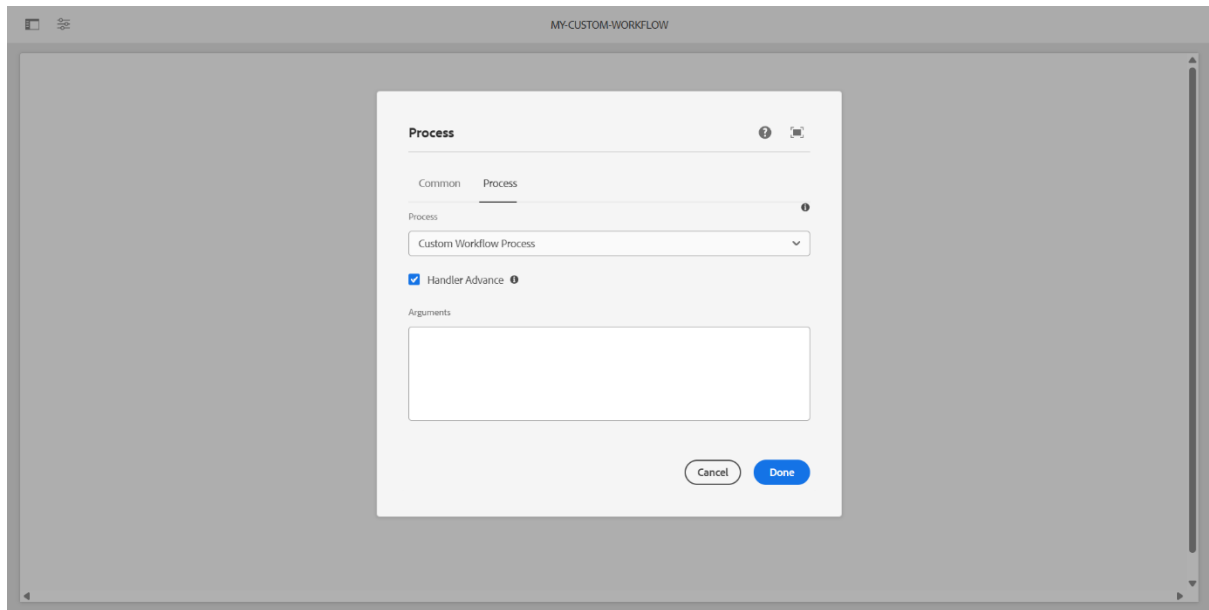
```

        LOGGER.error("[Custom Workflow] Resource Resolver is NULL.");
    }
}

```

## Screenshot:

- AEM Workflow Model UI with the custom process step configured.



```

02.04.2025 16:57:40.072 *INFO* [[0:0:0:0:0:0:1] [1743593259999] GET /libs/wcm/core/content/components.1743593259807.json HTTP/1.1] com.day.cq.wcm.core.impl.components.ComponentServlet provided components.
02.04.2025 16:57:45.669 *INFO* [[0:0:0:0:0:0:1] [1743593265631] POST /var/workflow/instances HTTP/1.1] com.adobe.granite.workflow.core.advance.DynamicParticipantNodeHandler Activate
com.adobe.granite.workflow.core.advance.DynamicParticipantNodeHandler@7f534f56
02.04.2025 16:57:45.702 *INFO* [sling-oak-observation-9] org.apache.sling.event.impl.jobs.queue.JobQueueImpl.Granite Workflow Queue Starting job queue Granite Workflow Queue
02.04.2025 16:57:45.704 *INFO* [FelixLogListener] Events.Service.org.apache.sling.event Service [QueueMBean for queue Granite Workflow Queue,6549, [org.apache.sling.event.jobs.jmx.StatisticsMBean]] ServiceEvent
REGISTERED
02.04.2025 16:57:45.713 *INFO* [JobHandler: /var/workflow/instances/server0/2025-04-02_1/my-custom-workflow_1:/content/myTraining/us/en/sample-page] com.adobe.granite.workflow.core.job.JobHandler Start of
Workflow Execution: Title=my-custom-workflow, Process=com.myTraining.core.workflow.CustomWorkflowProcess, Workfront-object-type=null
02.04.2025 16:57:45.713 *INFO* [JobHandler: /var/workflow/instances/server0/2025-04-02_1/my-custom-workflow_1:/content/myTraining/us/en/sample-page] com.myTraining.core.workflow.CustomWorkflowProcess [Custom
Workflow] Executing workflow process...
02.04.2025 16:57:45.713 *INFO* [JobHandler: /var/workflow/instances/server0/2025-04-02_1/my-custom-workflow_1:/content/myTraining/us/en/sample-page] com.myTraining.core.workflow.CustomWorkflowProcess [Custom
Workflow] Payload Path: /content/myTraining/us/en/sample-page
02.04.2025 16:57:45.713 *INFO* [JobHandler: /var/workflow/instances/server0/2025-04-02_1/my-custom-workflow_1:/content/myTraining/us/en/sample-page] com.myTraining.core.workflow.CustomWorkflowProcess [Custom
Workflow] Page Title: Sample Page
02.04.2025 16:57:45.713 *INFO* [JobHandler: /var/workflow/instances/server0/2025-04-02_1/my-custom-workflow_1:/content/myTraining/us/en/sample-page] com.myTraining.core.workflow.CustomWorkflowProcess [Custom
Workflow] Page Path: /content/myTraining/us/en/sample-page
02.04.2025 16:57:45.713 *INFO* [JobHandler: /var/workflow/instances/server0/2025-04-02_1/my-custom-workflow_1:/content/myTraining/us/en/sample-page] com.adobe.granite.workflow.core.job.JobHandler End of
Workflow Execution - Title: my-custom-workflow, Process: com.myTraining.core.workflow.CustomWorkflowProcess
02.04.2025 16:57:45.716 *INFO* [[0:0:0:0:0:0:1] [1743593265712] GET /content/myTraining/us/en/sample-page.html HTTP/1.1] com.day.cq.wcm.core.impl.designer.SearchPathLimiter Search path limiter configured with
searchPathLimiterFeatureToggleOn: true and searchPathThreshold: true.

```

## Step 3: Create an Event Handler to Print Resource Path in Logs

1. Create a Java class in the **core module**:

### Java code

```
package com.myTraining.core.listeners;
```

```
import org.apache.sling.api.resource.observation.ResourceChange;
```

```

import org.apache.sling.api.resource.observation.ResourceChangeListener;
import org.osgi.service.component.annotations.Component;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import java.util.List;

```

```

@Component(
    service = ResourceChangeListener.class,
    property = {
        ResourceChangeListener.PATHS + "=/content/myTraining/us",
        ResourceChangeListener.CHANGES + "=ADDED",
        ResourceChangeListener.CHANGES + "=CHANGED",
        ResourceChangeListener.CHANGES + "=REMOVED"
    }
)

public class ResourceEventHandler implements ResourceChangeListener {

    private static final Logger LOGGER =
        LoggerFactory.getLogger(ResourceEventHandler.class);

    @Override
    public void onChange(List<ResourceChange> changes) {
        for (ResourceChange change : changes) {
            LOGGER.info("[Resource Event] Type: {} | Path: {}", change.getType(),
                change.getPath());
        }
    }
}

```

## 2. Deploy the code and publish a page to see logs like:

Page Event Triggered for Path: /content/us/en/news/news-1

```

com.day.cq.wcm.core.impl.designer.SearchPathLimiter Search path limiter configured with searchPathLimiterFeatureToggleOn: true and searchPathThreshold: true.
02.04.2025 16:59:58.257 *INFO* [[0:0:0:0:0:0:1] [1743593338254] GET /mnt/override/apps/core/wcm/components/title/v3/title/_cq_dialog.html/content/myTraining/us/en/sample-page/jcr:content/root/container/title HTTP/1.1] com.day.cq.wcm.core.impl.designer.SearchPathLimiter Search path limiter configured with searchPathLimiterFeatureToggleOn: true and searchPathThreshold: true.
02.04.2025 16:59:01.883 *INFO* [sling-oak-observation-17] com.adobe.cq.ups.user.util.UserCreationListener UPS Feature is disabled.
02.04.2025 16:59:07.844 *INFO* [[0:0:0:0:0:0:1] [1743593347832] POST /content/myTraining/us/en/sample-page/jcr:content/root/container/title HTTP/1.1] com.day.cq.wcm.foundation.impl.ColumnCtlPostProcessor logging out of localServiceSession in process()
02.04.2025 16:59:07.863 *INFO* [sling-oak-observation-11] com.myTraining.core.listeners.ResourceEventHandler [Resource Event] Type: CHANGED | Path: /content/myTraining/us/en/sample-page/jcr:content
02.04.2025 16:59:07.863 *INFO* [sling-oak-observation-11] com.myTraining.core.listeners.ResourceEventHandler [Resource Event] Type: CHANGED | Path: /content/myTraining/us/en/sample-page/jcr:content/root/container/title
02.04.2025 16:59:07.881 *WARN* [sling-threadpool-045d439-2e34-4edb-a320-2df3ee9bbb06-(com.day.cq.wcm.workflow.impl.wcmWorkflowServiceImpl)-4] com.day.cq.wcm.workflow.impl.WcmWorkflowServiceImpl No resource found at the path : /content/myTraining/us/en/sample-page
02.04.2025 16:59:07.884 *INFO* [[0:0:0:0:0:0:1] [1743593347873] GET /content/myTraining/us/en/sample-page/jcr:content/root/container/title.html HTTP/1.1] com.day.cq.wcm.core.impl.designer.SearchPathLimiter Search path limiter configured with searchPathLimiterFeatureToggleOn: true and searchPathThreshold: true.
02.04.2025 16:59:07.887 *INFO* [EventAdminAsyncThread #8] org.apache.sling.commons.threads.impl.DefaultThreadPool Initializing thread pool [ThreadPool-083c35fe-6726-467b-ad50-d632379f8bb8 (com.day.cq.wcm.designerimporter.impl.CanvasPageDeleteHandler)] ...
02.04.2025 16:59:07.887 *INFO* [EventAdminAsyncThread #8] org.apache.sling.commons.threads.impl.DefaultThreadPool min-pool-size (5) < max-pool-size (10) for pool "ThreadPool-083c35fe-6726-467b-ad50-d632379f8bb8 (com.day.cq.wcm.designerimporter.impl.CanvasPageDeleteHandler)" which has unbounded queue (queue size -1). Set to 10
02.04.2025 16:59:07.887 *INFO* [EventAdminAsyncThread #8] org.apache.sling.commons.threads.impl.ThreadLocalCleaner Validating reflective access is permitted...

```

## Step 4: Create a Sling Job to Print "Hello World" in Logs

### 1. Create a Java class in the **core** module:

### Java code

#### HelloWorldJob.java

```
package com.myTraining.core.schedulers;
```

```
import org.apache.sling.event.jobs.Job;
```

```
import org.apache.sling.event.jobs.consumer.JobConsumer;
```

```
import org.osgi.service.component.annotations.Component;
```

```
import org.slf4j.Logger;
```

```
import org.slf4j.LoggerFactory;
```

```
@Component(
```

```
    service = JobConsumer.class,
```

```
    property = {
```

```
        JobConsumer.PROPERTY_TOPICS + "=myTraining/job/helloWorld"
```

```
    }
```

```
)
```

```
public class HelloWorldJob implements JobConsumer {
```

```
    private static final Logger LOGGER = LoggerFactory.getLogger(HelloWorldJob.class);
```

```
@Override
```

```
    public JobResult process(Job job) {
```

```

        LOGGER.info("[Sling Job] Executing Hello World Job...");
        LOGGER.info("[Sling Job] Hello World!");
        return JobResult.OK;
    }
}

```

2. Run the job using **Sling JobManager**:

## Java Code

### JobTrigger.java

```

package com.myTraining.core.schedulers;

import org.apache.sling.event.jobs.JobManager;
import org.osgi.service.component.annotations.Activate;
import org.osgi.service.component.annotations.Component;
import org.osgi.service.component.annotations.Reference;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import java.util.HashMap;
import java.util.Map;

@Component(service = JobTrigger.class, immediate = true)
public class JobTrigger {

    private static final Logger LOGGER = LoggerFactory.getLogger(JobTrigger.class);

    @Reference
    private JobManager jobManager;

    @Activate

```

```
protected void activate() {

    LOGGER.info("Activating JobTrigger - Automatically Triggering Sling Job...");

    triggerJob();

}

private void triggerJob() {

    if (jobManager != null) {

        Map<String, Object> jobProps = new HashMap<>();

        jobProps.put("message", "Triggered Hello World Job!");

        jobManager.addJob("myTraining/job/helloWorld", jobProps);

        LOGGER.info("Sling Job Triggered Successfully!");

    } else {

        LOGGER.error("JobManager is null. Cannot trigger the job!");

    }

}
```

### 3. Deploy and check logs for "Sling Job Executed: Hello World" message.

## Screenshot

```
02.04.2025 11:16:50.050 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6515, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:16:50.157 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6516, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:16:50.336 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6517, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:16:51.871 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6518, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:16:51.979 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6519, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:16:52.729 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6520, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:16:52.781 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6521, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:16:53.245 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6522, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:16:54.493 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6523, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:16:58.298 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6524, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:17:00.173 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6525, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:17:00.826 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6526, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:17:00.829 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6527, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:17:10.433 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6528, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:17:12.285 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6529, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:17:12.288 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6530, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:17:13.160 *INFO* [FelixLogListener] Events.Service.com.adobe.granite.repository Service [6531, [org.apache.jackrabbit.oak.api.jmx.SessionMBean]] ServiceEvent REGISTERED
02.04.2025 11:20:00.027 *INFO* [sling-default-1-YellowWorldSchedulerJob] com.myTraining.core.schedulers.YellowWorldScheduler [Scheduler] Yellow World!
02.04.2025 11:23:47.977 *INFO* [[0:0:0:0:0:0:0:1] [1743573227970] GET /libs/granite/core/content/login.html HTTP/1.1] org.apache.sling.i18n.impl.JcrResourceBundle Finding all dictionaries for 'en_IN' (basename:
<none>) ...
02.04.2025 11:23:47.981 *INFO* [[0:0:0:0:0:0:0:1] [1743573227970] GET /libs/granite/core/content/login.html HTTP/1.1] org.apache.sling.i18n.impl.JcrResourceBundle Finished loading 0 entries for 'en_IN'
(basename: <none>) in 4ms
02.04.2025 11:23:47.983 *INFO* [FelixLogListener] Events.Service.org.apache.sling.i18n Service [6532, [java.util.ResourceBundle]] ServiceEvent REGISTERED
02.04.2025 11:23:47.984 *INFO* [[0:0:0:0:0:0:0:1] [1743573227970] GET /libs/granite/core/content/login.html HTTP/1.1] org.apache.sling.i18n.impl.JcrResourceBundleProvider Currently loaded dictionaries across
```

## Step 5: Create a Scheduler to Print "Yellow World" Every 5 Minutes Using Cron Expression

1. Create a new Java class:

## Java code



```

package com.myTraining.core.schedulers;

import org.apache.sling.commons.scheduler.Scheduler;
import org.osgi.service.component.annotations.Activate;
import org.osgi.service.component.annotations.Component;
import org.osgi.service.component.annotations.Modified;
import org.osgi.service.metatype.annotations.AttributeDefinition;
import org.osgi.service.metatype.annotations.ObjectClassDefinition;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

import java.util.concurrent.atomic.AtomicBoolean;

@Component(service = Runnable.class, immediate = true, configurationPid =
"com.myproject.core.schedulers.YellowWorldScheduler")
public class YellowWorldScheduler implements Runnable {

    private static final Logger LOG = LoggerFactory.getLogger(YellowWorldScheduler.class);

    @ObjectClassDefinition(name = "Yellow World Scheduler Configuration", description =
"Scheduler to log 'Yellow World' every 5 minutes")
    public @interface Config {
        @AttributeDefinition(name = "Cron Expression", description = "Cron expression for
scheduling")
        String scheduler_expression() default "0 */5 * * * ?";
    }

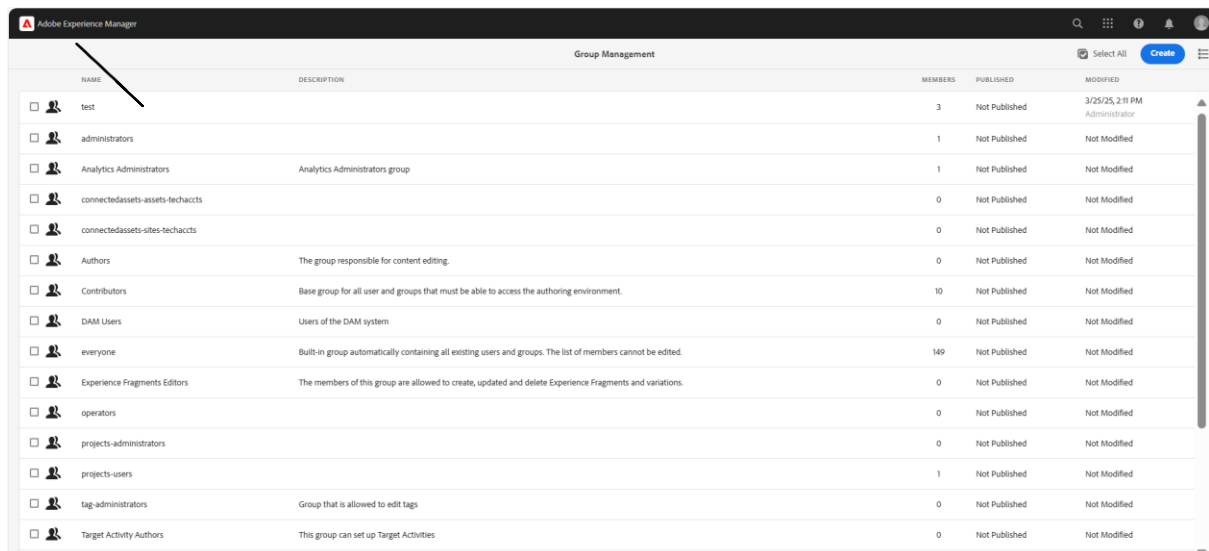
    private final AtomicBoolean running = new AtomicBoolean(false);

    @Activate
    @Modified
    protected void activate(final Config config) {
        LOG.info("YellowWorldScheduler activated with cron expression: {}",
config.scheduler_expression());
    }

    @Override
    public void run() {
        if (running.compareAndSet(false, true)) {
            try {
                LOG.info("Yellow World");
            } finally {
                running.set(false);
            }
        }
    }
}

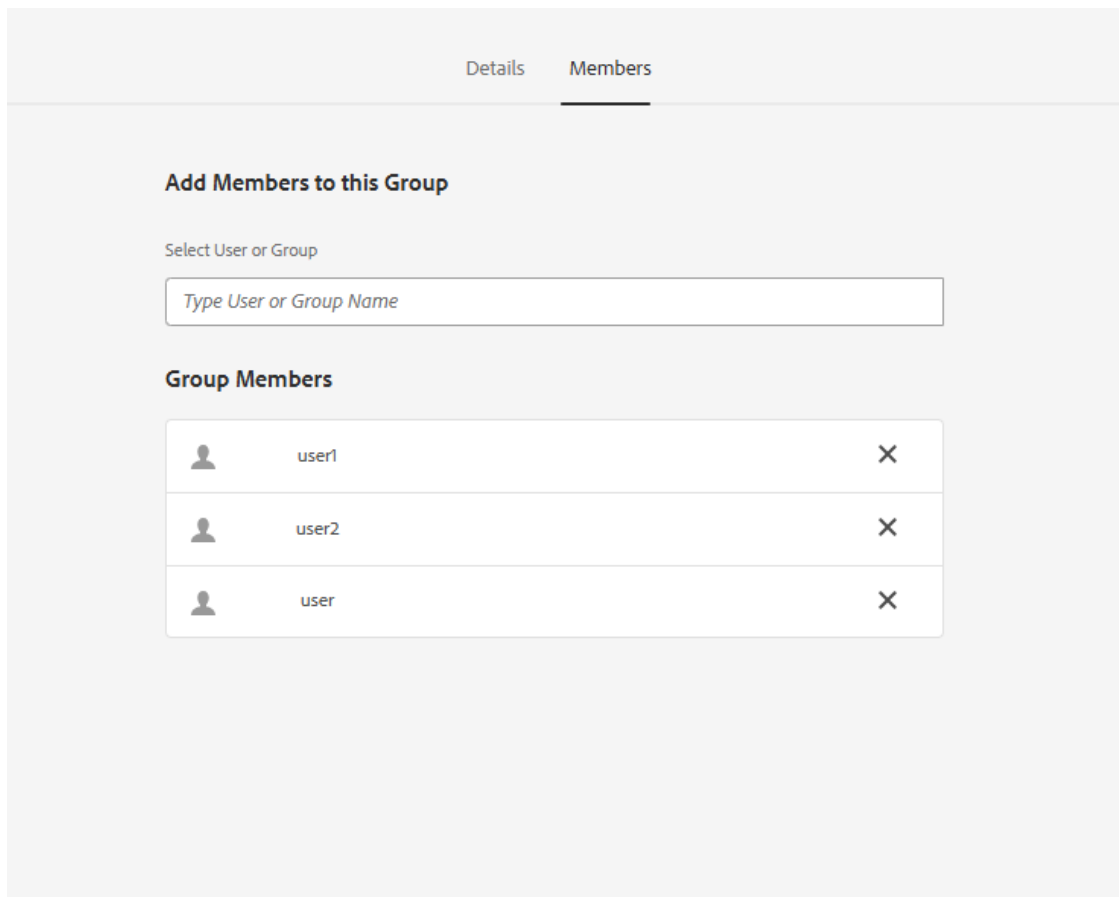
```

2. Deploy and check logs for "Scheduled Job Executed: Yellow World" every 5 minutes.



Group Management					Select All	Create
NAME	DESCRIPTION	MEMBERS	PUBLISHED	MODIFIED		
<input type="checkbox"/> test		3	Not Published	3/25/25, 2:11 PM		
<input type="checkbox"/> administrators		1	Not Published	Not Modified		
<input type="checkbox"/> Analytics Administrators	Analytics Administrators group	1	Not Published	Not Modified		
<input type="checkbox"/> connectedassets-assets-techacctcs		0	Not Published	Not Modified		
<input type="checkbox"/> connectedassets-sites-techacctcs		0	Not Published	Not Modified		
<input type="checkbox"/> Authors	The group responsible for content editing.	0	Not Published	Not Modified		
<input type="checkbox"/> Contributors	Base group for all user and groups that must be able to access the authoring environment.	10	Not Published	Not Modified		
<input type="checkbox"/> DAM Users	Users of the DAM system	0	Not Published	Not Modified		
<input type="checkbox"/> everyone	Built-in group automatically containing all existing users and groups. The list of members cannot be edited.	149	Not Published	Not Modified		
<input type="checkbox"/> Experience Fragments Editors	The members of this group are allowed to create, update and delete Experience Fragments and variations.	0	Not Published	Not Modified		
<input type="checkbox"/> operators		0	Not Published	Not Modified		
<input type="checkbox"/> projects-administrators		0	Not Published	Not Modified		
<input type="checkbox"/> projects-users		1	Not Published	Not Modified		
<input type="checkbox"/> tag-administrators	Group that is allowed to edit tags	0	Not Published	Not Modified		
<input type="checkbox"/> Target Activity Authors	This group can set up Target Activities	0	Not Published	Not Modified		

- User ui



DetailsMembers

Add Members to this Group

Select User or Group

Type User or Group Name

Group Members

user1

user2

user

- Permissions UI showing read-only access and replication privileges.

