

Task-10 CODE

QUESTION

1. Create a scheduler with a cron expression that should execute after the specified time.
2. Continuation: If a path is provided, all the child pages under that path should be published automatically according to the cron expression.

ANSWER:

OSGI CONFIGURATION

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

```
import org.osgi.service.metatype.annotations.AttributeDefinition;
```

```
import org.osgi.service.metatype.annotations.AttributeType;
```

```
import org.osgi.service.metatype.annotations.ObjectClassDefinition;
```

```
@ObjectClassDefinition(name = "my scheduler",description = "scheduler is ccreated")
```

```
public @interface schedulerpractisemrng {
```

```
    @AttributeDefinition(  
        name="service name",  
        type = AttributeType.STRING,  
        description = "enter the service name"  
    )
```

```
public String getservice_name() default "practise";
```

```
    @AttributeDefinition
```

```
(  
        name="can run concurrently",  
        type=AttributeType.BOOLEAN,  
        description="can run concurrently"  
    )
```

```
public boolean canrunconcurrently() default false;
```

```
@AttributeDefinition
```

```
(  
        name="Enabled scheduler",  
        type=AttributeType.BOOLEAN,  
        description="Enable the scheduler"
```

```

    )
    public boolean Enabledscheduler() default true;

    @AttributeDefinition(
        name="Expression",
        type = AttributeType.STRING,
        description = "enter the Expression"
    )

    public String getExpressi(); }

```

SCHEDULERS CODE FOR CORN EXPRESSION

```

package com.adobe.aem.guides.demo.core.schedulers;

import java.util.HashMap;
import java.util.Map;

import javax.jcr.Session;

import org.apache.sling.api.resource.LoginException;
import org.apache.sling.api.resource.ResourceResolver;
import org.apache.sling.api.resource.ResourceResolverFactory;
import org.apache.sling.commons.scheduler.ScheduleOptions;
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.Deactivate;
import org.osgi.service.component.annotations.Modified;
import org.osgi.service.component.annotations.Reference;
import org.osgi.service.metatype.annotations.Designate;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

import com.day.cq.replication.ReplicationActionType;
import com.day.cq.replication.Replicator;
import com.day.cq.wcm.api.Page;
import com.day.cq.wcm.api.PageManager;

@Designate(ocd = schedulerpractisemrng.class)
@Component(service = Runnable.class, immediate = true)
public class schedulermrng implements Runnable {
    private static final Logger log = LoggerFactory.getLogger(schedulermrng.class);

    @Reference

```

```

private Scheduler scheduler;

@Reference
private ResourceResolverFactory resourceResolverFactory;

@Reference
private Replicator replicator;

private static final String SERVICE_USER = "hemanth";

private String cronExpression;

@Modified
public void modify(schedulerpractisemrng sch) {
    this.cronExpression = sch.getExpressi();
    addscheduler(sch); // Re-schedule with the new cron expression
}

@Activate
public void activation(schedulerpractisemrng sch) {
    this.cronExpression = sch.getExpressi();
    // addscheduler(sch);
}

public void addscheduler(schedulerpractisemrng sch) {
    log.info("Scheduler is created");
    if (sch.Enabledscheduler()) {
        ScheduleOptions scheduleOptions = scheduler.EXPR(cronExpression);
        scheduleOptions.canRunConcurrently(sch.canrunconcurrently());
        scheduleOptions.name(sch.getservice_name());
        scheduler.schedule(this, scheduleOptions);
    }
}

@Deactivate
public void dectivate(schedulerpractisemrng sch) {
    removescheduler(sch);
}

public void removescheduler(schedulerpractisemrng sch) {
    log.info("Job is unscheduled");
    scheduler.unschedule(sch.getservice_name());
}

@Override
public void run() {

```

```

    log.info("Scheduler is running");
    log.info("My cron expression: " + cronExpression);
    publishPages("/content/Demo/us/en");
}

```

PUBLISH PAGES AUTOMATICALLY BY CORN EXP

```

private void publishPages(String path) {
    try (ResourceResolver resourceResolver = getServiceResourceResolver()) {
        PageManager pageManager = resourceResolver.adaptTo(PageManager.class);
        if (pageManager != null) {
            Page page = pageManager.getPage(path);
            if (page != null) {
                page.listChildren().forEachRemaining(childPage -> {
                    try {
                        log.info("Publishing page: {}", childPage.getPath());
                        Session session = resourceResolver.adaptTo(Session.class);
                        replicator.replicate(session, ReplicationActionType.ACTIVATE,
childPage.getPath());
                    } catch (Exception e) {
                        log.error("Error while publishing page: {}", childPage.getPath(), e);
                    }
                });
            } else {
                log.warn("No page found at path: {}", path);
            }
        }
    } catch (LoginException e) {
        log.error("Error obtaining service resource resolver", e);
    }
}

private ResourceResolver getServiceResourceResolver() throws LoginException {
    Map<String, Object> param = new HashMap<>();
    param.put(ResourceResolverFactory.SUBSERVICE, SERVICE_USER);
    return resourceResolverFactory.getServiceResourceResolver(param);
}
}

```

OSGI CONFIGURATION

my scheduler

scheduler is created

service name

practise

enter the service name (getservice.name)

can run concurrently

☐

can run concurrently (canrunconcurrently)

Enabled scheduler

☒

Enable the scheduler (Enabledscheduler)

Expression

0 0/2 * 1/1 * ? *

enter the Expression (getExpressi)

Configuration Information

Persistent Identity (PID)

com.adobe.aem.guides.demo.core.schedulers.schedulermrnng

Configuration Binding

Unbound or new configuration

Cancel

Reset

Delete

Unbind

Save

LOG MESSAGE FOR PAGE PUBLISHED

19.07.2024 11:14:59.275 *INFO*	[CM EventDispatcher (Fire ConfigurationEvent: pid=com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng)]	Schedulers.schedulerlmrngng Scheduler is created
com.adobe.aem.guides.demo.core.19.07.2024 11:16:00.007 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Scheduler is running
19.07.2024 11:16:00.007 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng My cron expression: 0 0/2 * 1/1 * ?
19.07.2024 11:16:00.022 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page:
/content/Demo/us/en/AboutUs19.07.2024 11:16:00.207 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page:
/content/Demo/us/en/Privacy19.07.2024 11:16:00.352 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page:
/content/Demo/us/en/Terms19.07.2024 11:16:00.542 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page:
/content/Demo/us/en/contactus19.07.2024 11:16:00.732 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page: /content/Demo/us/en/
19.07.2024 11:16:01.033 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page:
/content/Demo/us/en/article19.07.2024 11:16:01.240 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page:
/content/Demo/us/en/capstone19.07.2024 11:16:01.509 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page: /content/Demo/us/en/
19.07.2024 11:16:01.714 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page: /content/Demo/us/en/
19.07.2024 11:16:02.081 *INFO*	[slng-default-1-practise]	com.adobe.aem.guides.demo.core.schedulers.schedulerlmrngng Publishing page: