**CronJob:**

 The Word “CRON” is derived from Greek which means “Time” or Schedule, and the word “JOB” means performing some task. So now the word “CronJob”=Time+Job, which means performing some job in particular time.

The Concept of cronjob consists of types which interact with each other to complete the task. They are

1.    CronJob Type

2.    Job Type

3.    Trigger Type

**CronJob** Type: This type holds the business logic which is to be performed at particular times and intervals

**Job Type**: This type consists of logic to be executed which is defined by JobPerformable.

**Trigger Type**:  This type is used for scheduling when to run the job. Which uses cron expression[ \* \* \* \* ? \*]which indicates [minutes, hours, day of month, month, day of week, year] to run the job.

If we want make notifications periodically then we go for CronJob.

**Example**: If we want to send notification every day at 5pm then we can make a setup and perform CronJob.

**How to create a cronjob:**

1.    Create  the job

2.    Write the java class by extending the Abstractjobperformable or implementing the jobperformable interface. Override the perform method.

3.    Configure the java class in to the extension-spring.xml  file.

4.    Create the CronJob impex file

Register your CronJob into CronJob type.

Register our CronJob into Trigger type with schedule.

5.    Update your system so that server automatically runs the CronJob when the time reached.

**How to Start a CronJob:**

 There are different ways to start a cronjob which are given below :

1.      Manually start the CronJob using HMC

Goto hmc -> system -> cronjobs -> select CronJob -> click on "StartCronJobNow".

2.      Automatically running the CronJob Through Impex file

3.      Using the ant command

 ant runcronjob  -d  cronjob=“CronJobName “

4.      Using the javacode using the CronJob services we can run the cronjob.

**How to stop a cronjob:**

We can stop the cronjob by following ways:

1.      Using the abort method in the java code.it is done automatically after performing the CronJob.

2.      Manually from hmc we can stop the CronJob.

Below given step by step to create hybris cron job

**1. create a class TestCronJob as below**  
  
package aaa.work.core.cronjob;  
import de.hybris.platform.cronjob.enums.CronJobResult;  
import de.hybris.platform.cronjob.enums.CronJobStatus;  
import de.hybris.platform.cronjob.model.CronJobModel;  
import de.hybris.platform.servicelayer.cronjob.AbstractJobPerformable;  
import de.hybris.platform.servicelayer.cronjob.PerformResult;  
  
public class TestCronJob extends AbstractJobPerformable  
{  
  private String exportDir;  
                public String getExportDir() {  
                                return exportDir;  
                }  
                public void setExportDir(final String exportDir) {  
                                this.exportDir = exportDir;  
                }  
 @Override  
                public PerformResult perform(final CronJobModel arg0) {  
                                System.out.println("TestCronJob Job Started");  
 System.out.println("TestCronJob Job Finished");  
                                return new PerformResult(CronJobResult.SUCCESS, CronJobStatus.FINISHED);  
                }  
  
}

2. After create TestCronJob.java register this cron job in xml as below  
  
<beanid="testCronJob" class="aaa.work.core.cronjob.TestCronJob" scope="tenant"  
                                  parent="abstractJobPerformable">  
                                  <property name="exportDir" ref="exportDataBaseDirectory"/>  
                </bean>  
  
3. Now Run UPDATE Hybris  
  
4. After UPDATE your created Cron job TestCronJob will show in list of cronjob in Hmc where you can schedule .