**Cron Job - Project creation.**

**This is implementing using two ways.**

1.Using @Scheduler annotation.

2.Using Quartz scheduler (In the previous example, we executed scheduled tasks, but at the same time, we could not dynamically set the start time of the job, or pass parameters to it. To solve these problems, you can directly use Quartz.)



Quartz dependency is not necessary to implement Scheduler. This has been added for future Quartz project implementation.

Using @Scheduler

Need to add @EnableScheduling to main class.

@SpringBootApplication  
@EnableScheduling  
public class JobScheduleApplication {  
 public static void main(String[] args) {  
 SpringApplication.*run*(JobScheduleApplication.class, args);  
 }  
}

This is a simple scheduler class.

@Component  
public class SchedulerTest {  
  
 @Scheduled(cron = "0/5 \* \* \* \* ?")  
 public void everyFiveSeconds() {  
 System.*out*.println("Scheduler Executed.EveryFiveSeconds Scheduler: " + new Date());  
 }  
}

**Cron Job Time pattern**

Now understand the cron pattern(\* \* \* \* \* \*).

Guiding Tutorial:

<https://productresources.collibra.com/docs/collibra/latest/Content/Cron/co_spring-cron.htm>

<second> <minute> <hour> <day\_of\_month> <month> <day\_of\_week>

, - \* / , - \* / , - \* / , - \* ? / L W , - \* / , - \* ? / L #

0-59 0-59 0-23 1-31 1-12 JAN-DEC 0-7(where 0 is MON)MON-SUN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Position | Field | Allowed Values | Allowed Special characters | Examples |
| 1 | Second | 0-59 | , - \* / | 01) 10 = At 10th second.  02) \*/10 = every 10 seconds. |
| 2 | Minutes | 0-59 | , - \* / | 01) 30 = at the 30th minute.  02) \*/15 = every 15 minutes.  03) 5/10 = every 10 minutes starting at the 5th minute after the hour |
| 3 | Hours | 0-23 | , - \* / | 01) 10 = at 10 o'clock.  02) 8-10 = at 8,9 and 10 AM.  03) 6,18 = at 6 AM and at 6 PM. |
| 4 | Days of Month | 1-31 | , - \* ? / L W | 01) 3 = on the 3rd day of the month.  02) 1-4 = every first four days of the month.  03) 1,15 = the first day of the month and the 15th day of the month. |
| 5 | Month | 1-12  JAN-DEC | , - \* / | 01) 12 = in December.  02) 1-3 = every first three months of the year.  03) JUL,AUG = every July and August.  The names of the months are not case-sensitive. |
| 6 | Days of Week | 0-7(where 0 is MON)  MON-SUN | , - \* ? / L # | 01) TUE = every Tuesday.  02) 1-5 = every weekday, Monday to Friday.  03) MON,WED,FRI = every Monday, Wednesday and Friday.  04) L = only in combination with a digit or short day notation indicating the last day of the month. For example, 7L or SUNL indicates the last Sunday of the month, 3L or WEDL indicates the last Wednesday of the month.  05) 5#3 = on the 3rd Friday of the month.  The names of the days are not case-sensitive. |

**Special characters**

|  |  |
| --- | --- |
| \* | Used to select all **values** within a field.  Ex:\* in the minute field corresponds with every minute. |
| ? | Used to specify something in one of the two fields in which the character is allowed, but not the other, (used for days of the week.  <second> <minute> <hour> <day\_of\_month> <month> <day\_of\_week>  , - \* / , - \* / , - \* / , - \* ? / L W , - \* / , - \* ? / L #  \* \* \* 10 \* ?  Ex:If you want your trigger to fire on a particular day of the month, for example the 10th, but don't care what day of the week that happens to be, you could put "10" in the day-of-month field, and "?" in the day of the week field. |
| - | Used to specify ranges.  <second> <minute> <hour> <day\_of\_month> <month> <day\_of\_week>  , - \* / , - \* / , - \* / , - \* ? / L W , - \* / , - \* ? / L #  \* \* 10-12 \* \* \*  10-12 in the hour field means “the hours 10, 11 and 12”. |
| , | Used to specify additional values.  <second> <minute> <hour> <day\_of\_month> <month> <day\_of\_week>  , - \* / , - \* / , - \* / , - \* ? / L W , - \* / , - \* ? / L #  \* \* \* \* \* MON,WED,FRI  Example MON,WED,FRI in the day-of-week field means “the days Monday, Wednesday, and Friday”. |
| / | Used to specify increments.  <second> <minute> <hour> <day\_of\_month> <month> <day\_of\_week>  , - \* / , - \* / , - \* / , - \* ? / L W , - \* / , - \* ? / L #  0/15 \* \* \* \* \*  Ex:0/15 in the seconds field means “the seconds 0, 15, 30, and 45”. And 5/15 in the seconds field means “the seconds 5, 20, 35, and 50". You can also leave out the number before /, which is equivalent to having 0 before / . 1/3 in the day-of-month field means “fire every 3 days starting on the first day of the month”. |
| L | <second> <minute> <hour> <day\_of\_month> <month> <day\_of\_week>  , - \* / , - \* / , - \* / , - \* ? / L W , - \* / , - \* ? / L #  \* \* \* L \* \*  The value L in the day-of-month field means “the last day of the month” - day 31 for January, day 28 for February on non-leap years. You can also specify an offset from the last day of the month, such as “L-3” which would mean the third-to-last day of the calendar month.  You cannot use L in the day-of-week field by itself, you always have to prefix it by a digit (0-7) or the short day notation (MON-SUN). This combination means “the last xxx day of the month” - for example “6L” or "SATL" means “the last Saturday of the month”. |
| W | Used to specify the weekday (Monday-Friday) nearest the given day.  <second> <minute> <hour> <day\_of\_month> <month> <day\_of\_week>  , - \* / , - \* / , - \* / , - \* ? / L W , - \* / , - \* ? / L #  \* \* \* 15W \* \*  Ex:15W in the value for the day-of-month field, means the nearest weekday to the 15th of the month:  If the 15th is a Saturday, the trigger will fire on Friday the 14th.  If the 15th is a Sunday, the trigger will fire on Monday the 16th.  If the 15th is a Tuesday, then it will fire on Tuesday the 15th.  However if you specify 1W as the value for day-of-month, and the 1st is a Saturday, the trigger will fire on Monday the 3rd, as it will not ‘jump’ over the boundary of a month’s days. The ‘W’ character can only be specified when the value in the day-of-month field specifies a single day, not a range or list of days.  The 'L' and 'W' characters can also be combined in the day-of-month field to yield 'LW', which translates to \*"last weekday of the month"\*. |
| # | Used to specify “the nth” XXX day of the month.  Used to specify the weekday (Monday-Friday) nearest the given day.  <second> <minute> <hour> <day\_of\_month> <month> <day\_of\_week>  , - \* / , - \* / , - \* / , - \* ? / L W , - \* / , - \* ? / L #  \* \* \* \* \* 6#3  Ex:6#3 in the day-of-week field means “the third Saturday of the month” (day 6 = Friday and “#3” = the 3rd one in the month). Other examples: 2#1 is the first Tuesday of the month and 4#5 is the fifth Thursday of the month. Note that if you specify #5 and there is not 5 of the given day-of-week in the month, then no firing will occur that month. |

Example:

0 0 \* \* \* \* = the top of every hour of every day.

\*/10 \* \* \* \* \* = every ten seconds.

0 0 8-10 \* \* \* = 8, 9 and 10 o'clock of every day.

0 0 6,19 \* \* \* = 6:00 AM and 7:00 PM every day.

0 0/30 8-10 \* \* \* = 8:00, 8:30, 9:00, 9:30, 10:00 and 10:30 every day.

0 0 9-17 \* \* MON-FRI= on the hour nine-to-five weekdays.

0 0 0 25 12 ?= every Christmas Day at midnight, no matter what weekday it is.

Quartz

01. Create a project by applying bellow dependencies.

compile group: 'com.mchange', name: 'c3p0', version: '0.9.5.5'

<dependency>

<groupId>com.mchange</groupId>

<artifactId>c3p0</artifactId>

<version>0.9.5.5</version></dependency>

<https://mvnrepository.com/artifact/com.mchange/c3p0>

02. Add bellow dependencies in aplication.property file(The MySQL configuration defined in this file is not for Quartz. Quartz uses a separate configuration file that we will create later.)

03.Create Quartz property file(quartz.properties).

Quartz uses a properties file called quartz.properties(src/main/resources/quartz.properties). This file contains the most basic configuration of Quartz.

04.Create Quartz configuration class.