## Schedule issues for re-submission

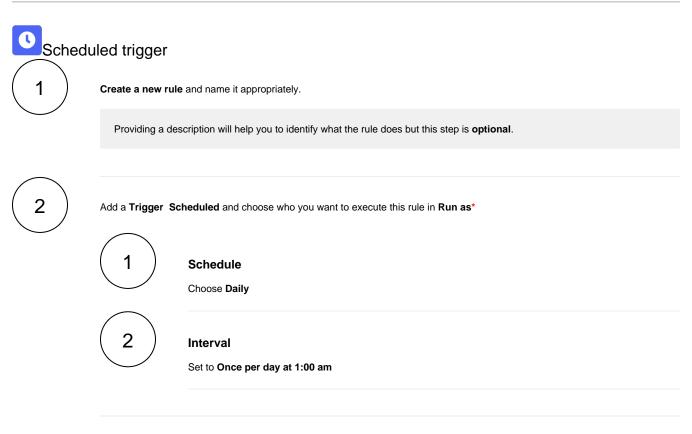
## On this page

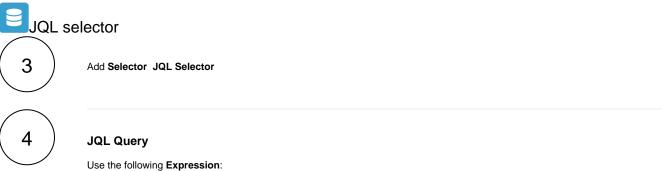
Scheduled trigger | JQL selector | Transition issue action | Screencast | Import the example | Related use cases

## Use case

When keeping track of issues to be done, it is helpful to **add a reminder to them**. Here we are going to use a **custom date-time field** called "Reminder". Every morning at 01:00 am, the following rule **che cks for issues that have reached the re-submission date** and will **re-transition them** to

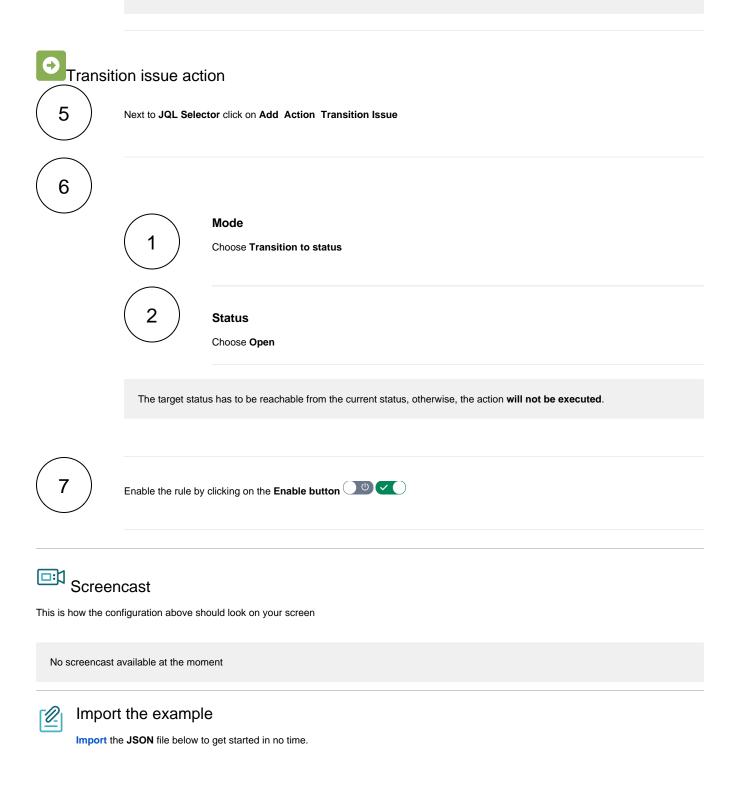
OPEN





nnnnn = startOfDay()

nnnnn has to be the name of the custom date field, like "Reminder" or "duedate".



After importing the JSON file, make sure to **check** the **configuration** of the rule. Non-existing configuration elements (issue types, fields, values etc.) will be highlighted.

```
{
"name": "Schedule issue for re-submission",
 "description": "",
"creator": "admin",
 "status": false,
 "triggerData": "",
 "triggerType": "SCHEDULED",
 "configuration": {
     "refs": [
         "issue",
         "system",
         "trigger.issue",
         "trigger.parent"
     ],
     "runAs": "admin",
     "scheduledTriggerCron.dailyWeeklyMonthly": "daily",
     "scheduledTriggerCron.daysOfMonthOpt": "dayOfMonth",
     "scheduledTriggerCron.monthDay": "1",
     "scheduledTriggerCron.week": "1",
     "scheduledTriggerCron.day": "1",
     "scheduledTriggerCron.interval": "0",
     "scheduledTriggerCron.runOnceHours": "1",
     "scheduledTriggerCron.runOnceMins": "0",
     "scheduledTriggerCron.runOnceMeridian": "am",
     "scheduledTriggerCron.runFromHours": "1",
     "scheduledTriggerCron.runFromMeridian": "am",
     "scheduledTriggerCron.runToHours": "1",
     "scheduledTriggerCron.runToMeridian": "am",
     "triggerType": ""
},
 "children": [
    {
         "sequence": 0,
         "type": "JQL_SELECTOR",
         "ruleEntityType": "SELECTOR",
         "configuration": {
             "refs": [
                 "system"
             ],
             "jql": "Reminder = startOfDay()",
             "jqlParsingMode": "jql",
             "actingUser": "field_00020"
         },
         "children": [
             {
                 "sequence": 0,
                 "type": "TRANSITION_ISSUE",
                 "ruleEntityType": "ACTION",
                 "configuration": {
                     "refs": [
                         "issue",
                         "issues",
                         "selector.issue",
                         "selector.parent",
                         "system"
                     ],
                     "option": "status",
                     "status": "10610",
                     "actingUser": "field_00020"
                 },
                 "children": null,
                 "hasChildren": false
             }
         ],
         "hasChildren": true
     }
],
 "hasChildren": true
```

}

## Related use cases

Title	Automated action	JWT feature	Label
Transition affected issues when releasing a version (manually)	Transition issue action	<b>\$</b>	
Close epic when stories are done	Transition issue action	<b>\$</b>	
Automatically close parent when all sub-tasks are done	Transition issue action	<b>\$</b>	STAFF PICK
Resolve issues when all options of a checklist are checked	Transition issue action	<b>\$</b>	
Schedule issues for re-submission	Transition issue action	\$	
Schedule the escalation of overdue issues	Transition Issue	\$	
Keep the status of parents and sub-tasks in sync	Transition issue action	\$	STAFF PICK
Transition affected issues when releasing a version (automatically)	Transition issue action	\$	STAFF PICK
Transition issues when assigned to a developer	Transition issue action	\$	
Automatically close sub-tasks when parent is completed	Transition issue action	\$	STAFF PICK
Re-open issue when a new comment is added	Transition issue action	<del>\</del>	
Close stories when epic is done	Transition issue action		

If you still have questions, feel free to refer to our support team.