

# Add three days skipping weekends automatically to a Date Picker

On this page

[Configuration](#) | [Related use cases](#)

## Use case

Add three days to a Date picker from the date that you prefer. For demonstration purposes, we are going to use the current date.



## Configuration

1

Add the post function [Update fields](#) to the transition of your choice.

2

### Target issue\*

Select **current issue**.

3

### Fields

1

#### Field\*

Choose a date field, e.g., **Date picker**, **Planned end date**, **Planned start date**, **Due date**...etc.  
In this example, we will consider **Due date** as our date field.

2

#### Value\*

Select **Set field value manually (parser expression)** and enter the following expression:

```
%{dayOfTheWeek({system.currentDateTime},RUN_AS_LOCAL) in [3,4,5] ?
dateTimeToString(addDays({issue.dueDate}, 5, RUN_AS_LOCAL), "YYYY-MM-DD",
RUN_AS_LANG) : dateTimeToString(addDays({issue.dueDate}, 3, RUN_AS_LOCAL),
"YYYY-MM-DD", RUN_AS_LANG)}
```

With minor modifications, you will be able to add up to 5 days to any date skipping weekends as long as you increase or reduce the count in [3,4,5] for every unit added or deducted respectively from the number of days that you would like to skip and also increase or reduce the values of the number 5 by an unit.

For instance, adding 4 days instead of 3 would suppose adding a unit to 3 and 5 (4 and 6 after the change) and adding another number to the number list (it would become [2,3,4,5]).



## Related use cases

Use case	JWT feature	Workflow function	Parser functions	Label
Add a sub-task's summary and key to the description of its parent		Update fields		
Add three days skipping weekends automatically to a Date Picker		Update fields	<code>dateTimeToString()</code> <code>dayOfTheWeek()</code> <code>addDays()</code>	
Add watchers from another field		Update fields Transition issue	<code>union()</code> <code>toStringList()</code>	
Assign an issue to the project lead, if the issue is unassigned on creation		Update fields		
Assign an issue to the user who last commented on it		Update fields		
Assign important issues to the project lead		Update fields		
Copy field values from epic to issues under it after creation		Update fields Transition issue	<code>epic()</code> <code>issuesUnderEpic()</code> <code>filterByStatus()</code>	
Copy labels of a sub-task to the parent issue upon closing		Update fields		
Keep parent's priority in sync		Update fields		
Obtain the difference between two dates		Update fields		
Set a date field to a future date		Update fields	<code>dateTimeToString()</code>	
Set a date field to the current date		Update fields	<code>dateTimeToString()</code>	
Set Fix version to Affects version when resolving an issue		Update fields		
Set the priority to Highest if the 'Infrastructure' component is selected		Update fields		

If you still have questions, feel free to refer to our [support](#) team.