Sample Groovy Scripts to Create/Update Asset workflow post function

- Get Asset's Attribute value
- Get Asset's Multiple Value Attribute values by name as List
- Get Asset's Multiple Value Attribute values by ID as List
- Get Asset's Attribute values by asset object in a loop
- Get "Text Field" JIRA custom field value
- Get "Select List (single choice)" or "Radio Buttons" JIRA custom field value
- Get "Select List (multiple choices)" or "Checkboxes" JIRA custom field values
- Get old and new status name of transitioned issue
- Get resolution of transitioned issue
- Get Created/Updated Date of an issue
- Get Reporter/Assignee user name of an issue
- Get Current datetime

Please note that you can directly use `issue` or `originalIssue` of the transition, or fetch an issue by key or id with IssueManager (like `anotherIssue` in examples).

return statements are optional but it makes script more readable.

Get Asset's Attribute value

```groovy
return aipUtils.getAttributeValueAsStringByName(asset, 'Quantity');
```

Result: 40

Get Asset's Multiple Value Attribute values by name as List

```groovy
import inventoryplugin.entity.JipInventoryItem;
import org.apache.commons.lang3.StringUtils;

String getMultipleValues() {  
def result = '';  
def cityList = aipUtils.getMultiAttributeValueAsListByName(asset, 'Cities ListBox Multiple');  
for (String city : cityList) {  
    result += city + ', '  
}  
return result;
}

return getMultipleValues();
```

Result: New York, London,

Get Asset's Multiple Value Attribute values by ID as List

```groovy
import inventoryplugin.entity.JipInventoryItem;
import org.apache.commons.lang3.StringUtils;

String getMultipleValues() {  
def result = '';  
def cityList = aipUtils.getMultiAttributeValueAsListById(asset, 63);  
for (String city : cityList) {  
    result += city + ', '  
}  
return result;
}

return getMultipleValues();
```
Get Asset's Attribute values by asset object in a loop

Script:

```java
import inventoryplugin.entity.JipInventory
import inventoryplugin.entity.JipInventoryItem

def result = ""
result = result + "\n" + "Asset name: " + asset.getName()
result = result + "\n" + "Form name: " + asset.getForm().getFormName()
result = result + "\n" + "Attachments details: " + asset.getAttachments()
result = result + "\n" + "Creator: " + asset.getCreator()
result = result + "\n" + "Create time: " + asset.getCreated()

// attributes of asset
for(JipInventoryItem inventoryItem: asset.getInventoryItems()){
    // For the demo, we append all values to the Text area, so we skip TextArea attribute type values.
    if(inventoryItem.getFormAttribute().getAttribute().getAttributeType() != 'TextArea'){
        result = result + "\n" + inventoryItem.getFormAttribute().getAttribute().getAttributeName() + " : " + inventoryItem.getValue()
    }
}

return issue.description + "\n" + result
```

Result

*Creating Quick Filters* You can add your own Quick Filters in the board configuration (select *Board > Configure*)

Asset name: My second asset 001
Form name: All type of attributes form
Creator: admin
Create time: 2018-12-11 13:30:05.124

Attribute name: field value

----------------------------------
CheckboxList Field: bal
City: izm
Cities: ist, ada
InventoryList (All Assets List): 3
InventoryListByForm (Assets List By Form(s)): 10
City as DropDownList Field: ada
Purchase Date: 02.01.19
Assign DateTime: 02.01.19 15:16
Device IP: 10.0.0.2
Device IPv6: 2001:0db8:85a3:0000:0000:8a2e:0370:7334
Some Free Text: As a user, I can find important items on the board by using the customisable "Quick Filters" above >> Try clicking the "Only My Issues" Quick Filter above
A great URL example: http://www.snapbytes.com/
Assigned User: admin(admin)
Get "Text Field" JIRA custom field value

Script

```java
import com.atlassian.jira.issue.Issue
import com.atlassian.jira.component.ComponentAccessor

def customFieldManager = ComponentAccessor.getCustomFieldManager()
def cField = customFieldManager.getCustomFieldObject("customfield_11000")
def cFieldValue = issue.getCustomFieldValue(cField).getValue()

return cFieldValue
```

Result: Los Angeles

Get "Select List (single choice)" or "Radio Buttons" JIRA custom field value

Script

```java
import com.atlassian.jira.issue.Issue
import com.atlassian.jira.component.ComponentAccessor

def customFieldManager = ComponentAccessor.getCustomFieldManager()
def cField = customFieldManager.getCustomFieldObject("customfield_11000")
def cFieldValue = issue.getCustomFieldValue(cField).getValue()

return cFieldValue
```

Result: Los Angeles

Get "Select List (multiple choices)" or "Checkboxes" JIRA custom field values

Script

```java
import com.atlassian.jira.issue.Issue
import com.atlassian.jira.component.ComponentAccessor

def customFieldManager = ComponentAccessor.getCustomFieldManager()
def cField = customFieldManager.getCustomFieldObject("customfield_11001")
def cFieldValue = issue.getCustomFieldValue(cField)

def cityNameList = new ArrayList<String>()
for(String cityName: cFieldValue){
    cityNameList.add(cityName)
}
return String.join(", ", cityNameList)
```

Result: Los Angeles, San Francisco, stanbul

You can directly use issue or originalIssue of the transition, or get another issue with IssueManager (like example below).
import com.atlassian.jira.issue.Issue
import com.atlassian.jira.component.ComponentAccessor

def anotherIssue = ComponentAccessor.getIssueManager().getIssueObject("AP-5");

def issueManager = ComponentAccessor.getIssueManager()
def customFieldManager = ComponentAccessor.getCustomFieldManager()
def cfield = customFieldManager.getCustomFieldObject("customfield_11001")
def cFieldValue = anotherIssue.getCustomFieldValue(cfield)

def cityNameList = new ArrayList<String>()
for(String cityName: cFieldValue){
    cityNameList.add(cityName)
}
return String.join(",", cityList)

Result: Los Angeles, San Francisco

Get old and new status name of transitioned issue

return originalIssue.getStatus().getSimpleStatus().getName() + " -> " + issue.getStatus().getSimpleStatus().getName()

Result: To Do -> In Progress

Get resolution of transitioned issue

Please note that getResolution may be null for unresolved issues.

return issue.getResolution() != null ? issue.getResolution().getName(): "Not resolved"

Result: Done

Get resolution of an other issue

Please note that getResolution may be null for unresolved issues.

Result: Done

Get Created/Updated Date of an issue

If you want to assign a date or dateTime asset attribute, value must be in ISO time format, i.e: 2018-11-06T10:24:37.513

- For date-time fields, "second" and rest of the value will be purged automatically. Result example: 2018-11-06T10:24
- For date fields, "minute" and rest of the value will be purged automatically. Result example: 2018-11-06
Get Reporter/Assignee user name of an issue

If you want to assign a date or dateTime asset attribute, value must be in ISO time format, i.e: 2018-11-06T10:24:37.513

- For date-time fields, "second" and rest of the value will be purged automatically. Result example: 2018-11-06T10:24
- For date fields, "minute" and rest of the value will be purged automatically. Result example: 2018-11-06

Get Current datetime

If you want to assign a date or dateTime asset attribute, value must be in ISO time format, i.e: 2018-11-06T10:24:37.513

- For date-time fields, "second" and rest of the value will be purged automatically. Result example: 2018-11-06T10:24
- For date fields, "minute" and rest of the value will be purged automatically. Result example: 2018-11-06

Result: 2018-11-06T10:24:37.513

Result: marla

Result: 2019-01-07T17:03:43.767