

# Exporting a Test Set

Document Generator allows you to get following data from the Xray Test Set:

- [Test Set Status](#)
- [Iterating over Tests](#)

## Test Set Status

Export all Test Set Status with name and percentage from the Test Set

```
${Test Set Status}
```

You can print the status of the Test Set by using the following notation:

% per Status	Number of Tests per Status
\${Test Set Status.NameOfStatus}	\${Test Set Status.NameOfStatus.Count}

See the real example below:

```
#Test Set Status Percentage + total of testes per status
Todo: ${Test Set Status.TODO} % (${Test Set Status.TODO.Count})
Fail: ${Test Set Status.FAIL} % (${Test Set Status.FAIL.Count})
Pass: ${Test Set Status.PASS} % (${Test Set Status.PASS.Count})
Executing: ${Test Set Status.EXECUTING} % (${Test Set Status.EXECUTING.Count})
Aborted: ${Test Set Status.ABORTED} % (${Test Set Status.ABORTED.Count})
```

You can also use the translated name of Test Set Status to get the same information.

Language	Translated Name
ES	Estado del Grupo de Tests
DE	Testsetstatus
FR	État de la Suite de Tests

See the example:

### Expand to see the examples on sample code

```
#Test Set Status Percentage + total of testes per status
Todo: ${Estado del Grupo de Tests.TODO} % (${Estado del Grupo de Tests.TODO.Count})
Fail: ${Estado del Grupo de Tests.FAIL} % (${Estado del Grupo de Tests.FAIL.Count})
Pass: ${Estado del Grupo de Tests.PASS} % (${Estado del Grupo de Tests.PASS.Count})
Executing: ${Estado del Grupo de Tests.EXECUTING} % (${Estado del Grupo de Tests.EXECUTING.Count})
Aborted: ${Estado del Grupo de Tests.ABORTED} % (${Estado del Grupo de Tests.ABORTED.Count})
```

## Iterating over Tests

As it is not known in advance how many Tests exist in a Test Set, there is a notation that allows iterating a section over all the Tests of a Test Set. The notation is:

#### Expand to see the examples on sample code

```
#{for tests}
    Content and Test Mappings.      Example: ${Tests[n].Key}
#{end}

or

#{for j=TestsCount}
    Content and Test Mappings.      Example: ${Tests[j].Summary}
#{end}
```

**n** is the index of Test, starting from 0. The field **TestsCount** was created in order to give the total of Tests.

**Note:** Test Sets are Jira Issues + Xray data, so, you can print native Jira fields or even native Iterations. You can have a combination of both if you need.