# {{ project.name }}

This template is an example of how Jinja2 variables can be used in a Word template to dynamically drop-in information and generate tables. The following output is based on the Ghostwriter project: {{ client.name }} ({{ client.short\_name }}) {{ project.type }} (generated on {{ report\_date }})

The “raw” (JSON) report will show you the data accessible from your template. Some of this data is chopped-up into smaller bits to make it easy to reassemble in different ways. For example, the project’s start and end dates are accessible as pre-formatted dates based on your locale:

{{ project.start\_date }} – {{ project.end\_date }}

Perhaps you want to reference only pieces (e.g., day, month, and year) of these dates to present them in different ways. Here is one option for dynamically assembling a date range:

{% if project.start\_year == project.end\_year %}{% if project.start\_month == project.end\_month %}{{ project.start\_month }} {{ project.start\_day }}–{{ project.end\_day }}, {{ project.end\_year }}{% else %}{{ project.start\_day }} {{ project.start\_month }} to {{ project.end\_day }} {{ project.end\_month }} {{ project.end\_year }}{% endif %}{% else %}{{ project.start\_day }} {{ project.start\_month }} {{ project.end\_year }} to {{ project.end\_day }} {{ project.end\_month }} {{ project.end\_year }}{% endif %}

The above block may look confusing as one long line. This is necessary because Jinja2 removes all the statements when rendering the document. If there were returns at the end of each of the above if/else statements, the rendered date range would be surrounded by blank lines.

Adding new lines will help you debug and better understand more complex Jinja2 statements but remember to collapse everything back into one line before finalizing your template.

You can also pull in some pre-calculated values for various parts of your project and perform calculations:

Table 1 – Assessment Totals

|  |  |  |
| --- | --- | --- |
| Value | Total | Math |
| **Objectives** | {{ totals.objectives }} |  |
| **Completed Objectives** | {{ totals.objectives\_completed }} | {% if totals.objectives > 0 %}{{ totals.objectives\_completed / totals.objectives }}{% else %}100{% endif %}% |
| **Findings** | {{ totals.findings }} |  |
| **Team Members** | {{ totals.team }} |  |
| **Targeted Hosts** | {{ totals.targets }} |  |
| **Scope** | {{ totals.scope }} |  |

Be mindful of performing math like dividing a value that could be zero. In the above example, this template would fail rendering with an error if total.objectives was zero and the block did not have if totals.objectives > 0.

You will see an error like this:

Word document generation failed because the selected template has Jinja2 code that attempts to divide by zero

Tip: Before performing math, check if the number is greater than zero

## Assessment Points of Contact & Stakeholders

You can also create tables with loops:

Table 2 – {{ client.name }} Points of Contact

|  |  |  |
| --- | --- | --- |
| Name | Role | Email |
| {%tr for poc in client.contacts %} | | |
| {{ poc.name }} | {{ poc.job\_title }} | {{ poc.email }} |
| {%tr endfor %} | | |

Table 3 – {{ company.name }} Points of Contact

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Role | Email | Phone |
| {%tr for member in team %} | | | |
| {{ member.name }} | {{ member.role }} | {{ member.email }} | {{ member.phone }} |
| {%tr endfor %} | | | |

Table 4 – Domain Names Used for Assessment Activities

|  |  |
| --- | --- |
| Domain Name | Role |
| {%tr for domain in infrastructure.domains %} | |
| {{ domain.domain }} | {{ domain.activity }} |
| {%tr endfor %} | |

Table 5 – Servers Used for Assessment Activities

|  |  |  |
| --- | --- | --- |
| IP Address | Purpose | Role |
| {%tr for server in infrastructure.servers %} | | |
| {{ server.ip\_address }} | {{ server.activity }} | {{ server.role }} |
| {%tr endfor %} | | |
| {%tr for server in infrastructure.cloud %} | | |
| {{ server.ip\_address }} | {{ server.activity }} | {{ server.role }} |
| {%tr endfor %} | | |

Table 6 – Summary of Findings

|  |  |
| --- | --- |
| Finding | Severity |
| {%tr for finding in findings %} | |
| {{ finding.title }} | {% cellbg finding.severity\_color %}{{ finding.severity }} |
| {%tr endfor %} | |

There are numerous Jinja2 filters available within templates. Ghostwriter also has some custom filters (see the wiki). The above template looks like this with the filter\_severity filter:

|  |  |
| --- | --- |
| Finding | Severity |
| {%tr for finding in findings|filter\_severity([“Critical”, “High”]) %} | |
| {{ finding.title }} | {% cellbg finding.severity\_color %}{{ finding.severity }} |
| {%tr endfor %} | |

Here is that table again with the filter\_type filter:

|  |  |
| --- | --- |
| Finding | Severity |
| {%tr for finding in findings|filter\_type([“Network”, “Web”]) %} | |
| {{ finding.title }} | {% cellbg finding.severity\_color %}{{ finding.severity }} |
| {%tr endfor %} | |

Findings–and some other objects–include special versions of their content called *RichText* objects. These objects have the same name as the normal object with “\_rt” at the end. For example, a finding has a *Description* field that you access with finding.description. There is also a finding.description\_rt version that drops into a Word document fully formatted and styled.

Check the wiki for more and look at the following section.

{% for finding in findings %}

{{ finding.title }}

#### Severity – {{ finding.severity\_rt }}

**CVSS Score:** {{r finding.cvss\_score\_rt }}

**CVSS Vector:** {{r finding.cvss\_vector\_rt }}

#### Affected Entities

{{p finding.affected\_entities\_rt }}

#### Description

{{p finding.description\_rt }}

#### Impact

{{p finding.impact\_rt }}

#### Mitigation

{{p finding.recommendation\_rt }}

#### Replication Steps

{{p finding.replication\_steps\_rt }}

#### References

{{p finding.references\_rt }}

{% endfor %}