

_qubeuiPanelJobsTreeStructureAddCustomLevelHow To

It is possible to write your own Level Functions using Python, but first some definitions so we're on the same page.

Level Function

Level Functions define the job tree structure. As a typical tree structure, a Level Function may have a parent and/or children. Each Level Function is given jobs that match its parent and in turn hand down jobs that match itself to its children. If there is no parent, that is the Level Function is the root of the structure, then it processes all jobs. If there is no child Level Function then all jobs that match the Level Function become its children.

The function of a Level Function sorts jobs into different groups. The function runs on a per job basis (so be aware of performance issues) it is passed a single job at a time and returns the name of the group(s) it belongs to. By returning a list or tuple of strings a single job can appear in several groups. Group names are not set or stored anywhere, they are generated by the Level Functions themselves.

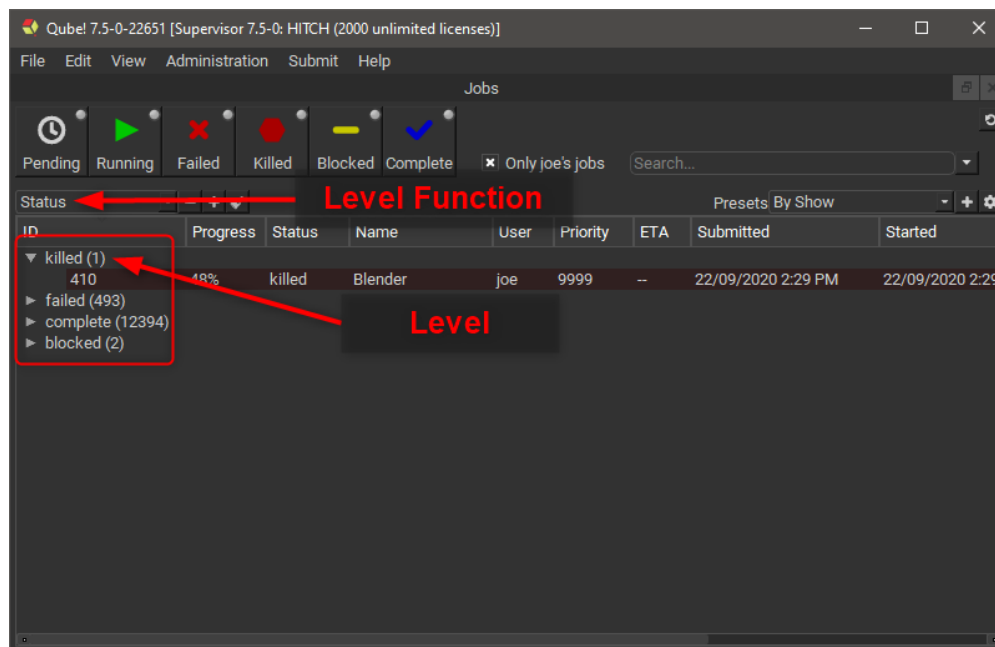
There are many built-in Level Functions, but you can also make your own custom levels using Python.

Level

A Level is what the Qube! UI calls the 'groups' referred to in the above Level Function definition.

Level name

A Level name is the text that is displayed in the tree structure at the Levels (branches of the tree). Level names are generated by the Level Function.

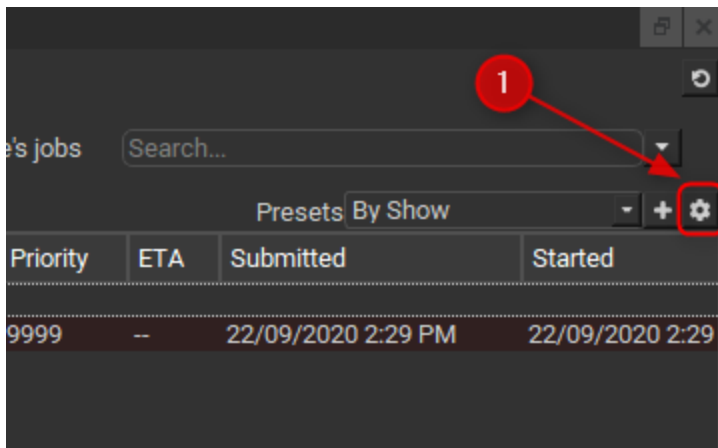


The Level Functions used within a particular structure act like ANDs, that is, for a job to be a leaf under a series of Level Functions it has to match all of them.

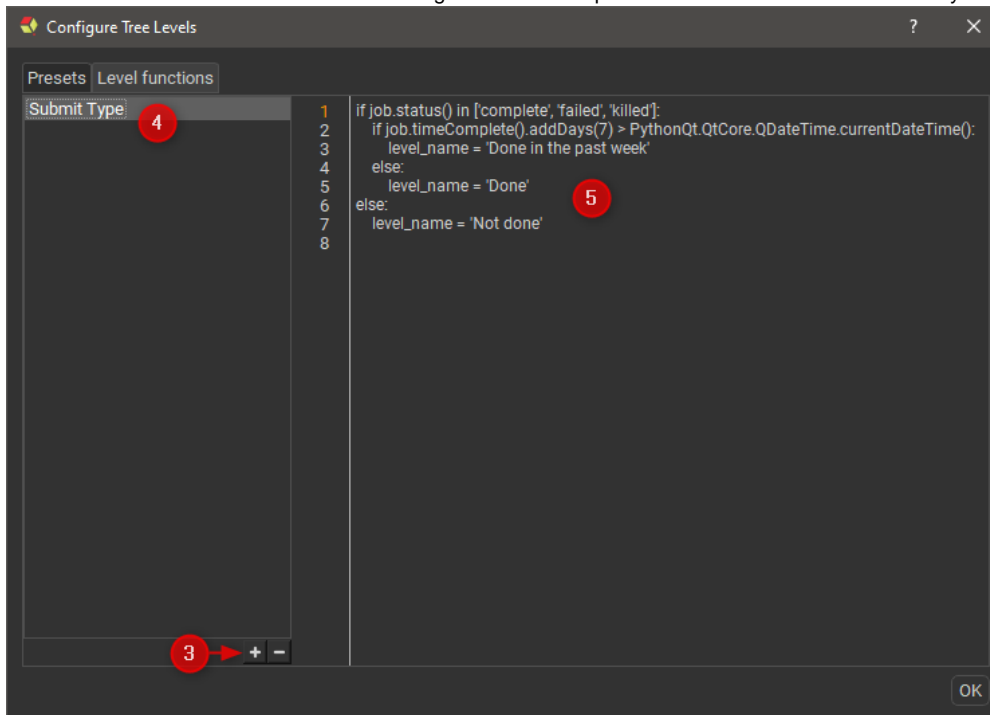
Creating a custom Level Function

As mentioned above, you can create your own custom Level Functions using Python.

1. Click on the gear icon to the right of the Presets selector.



2. In the dialog that appears select the "Level functions" tab.
3. Add a new Level Function using the plus button at the bottom of the Level Function list.
4. Double click on the new level function and give it a name, for this example call it "Submit Type".
5. You will notice that in the code editor on the right a basic example of a Level Function is automatically filled.



6. The Qube! UI pre-defines a variable **job** which you use to set another pre-defined variable **level_name**, you set it to either a string or list/tuple of strings.
7. Copy/paste the following text into the code editor.

```

if 'submitType' in job.packages().keys():                # If the job has a
submitType value in it's package
    level_name = job.packages()['submitType'].value()    # then set the level_name
to the value of the submitType
else:
    level_name = 'Unknown'                               # otherwise just set the
level_name to Unknown
  
```

8. Close the dialog and add a Level to the job list.
9. Using the Level Function selector choose your new level.
10. Enjoy the fruits of your labor.

Qubel 7.5-0-22651 [Supervisor 7.5-0: HITCH (2000 unlimited licenses)]

File Edit View Administration Submit Help

Jobs

Pending Running Failed Killed Blocked Complete

Only joe's jobs Search...

SubmitType Presets By Show

ID	Progress	Status	Name	User	Priority	ETA	Submitted	Started
▶ 3dsmax_batch (5)								
▼ blender (9)								
7	0%	complete	05_025_A_0028	joe	9999	--	14/07/2020 11:03 AM	14/07/2020 11:03 AM
26	100%	complete	Batch test	joe	9999	--	23/07/2020 2:04 PM	23/07/2020 2:04 PM
401	100%	complete	Blender	joe	9999	--	21/09/2020 4:19 PM	21/09/2020 5:19 PM
402	100%	complete	Blender	joe	9999	--	21/09/2020 6:47 PM	21/09/2020 6:47 PM
408	100%	complete	Blender	joe	9999	--	22/09/2020 2:21 PM	22/09/2020 2:21 PM
409	100%	complete	Blender	joe	9999	--	22/09/2020 2:22 PM	22/09/2020 2:22 PM
410	48%	killed	Blender	joe	9999	--	22/09/2020 2:29 PM	22/09/2020 2:29 PM
411	100%	complete	Blender	joe	9999	--	22/09/2020 2:36 PM	22/09/2020 2:36 PM
412	100%	complete	Blender	joe	9999	--	22/09/2020 2:42 PM	22/09/2020 2:42 PM
▶ cmdline (52)								
▶ cmdrange (10)								
▶ maya_batch_SW (3)								
▶ maya_jobtype (7)								
▶ test_set (4)								
▶ Unknown (12800)								