Cheat Sheet qpy

The Queue Management System in Python - version 0.0, 2018

Basic usage: `qpy <command> [options]` Use `TAB` for completion and `? TAB TAB` to display a help

Command	Options	Explanation
`restart`		(re)starts the background environment for qpy
`sub`	`[-n <# cores>] [-m <memory, in GB>] <command>`	submits the command `<command>`, optionally giving the number of cores and memory
`status`		shows the status of nodes and users
`check`	`[<status>] [<dir>] [<job IDs>]`	lists ... ... all the jobs with the given status(es), or that was submitted from given directory(ies), or has ID given in `<job IDs>`. If no option is given, `clean` and `kill` have no effect, whereas `check` lists all jobs
`clean`	`[<status>] [<dir>] [<job IDs>]`	cleans ...
`kill`	`[<status>] [<dir>] [<job IDs>]`	kills ...
`config`	`[checkFMT <pattern>]`	shows the pattern used for check, or sets it to `<pattern>`
	`[colour <false|true>]`	turns the output of the `check` command coloured (`true`) or disable the colours (`false`)
	`[colourScheme <colour 1> ... <colour 5>]`	shows the colours used in the output of the `check` command, or sets them
`ctrlQueue`	`pause`	pauses the submission of jobs
	`continue`	continues the submission of jobs
	`jump <job IDs> <target>`	moves jobs with IDs in `<job IDs>` to `<target>`, that can be a ID, `begin`, or `end`
`tutorial`	`[<keyword>]`	shows the tutorial, at `<keyword>`
`finish`		finishes the background environment for qpy

Possible job statuses	
`queue`	Job in the queue, not running yet
`running`	Job being executed
`done`	Job has finished
`undone`	Job was removed from the queue before running
`kill`	Job was killed when running

Environment variables	
`QPY_JOB_ID`	job ID
`QPY_NODE`	node where job is running
`QPY_N_CORES`	number of requested cores
`QPY_MEM`	requested memory

Possible modifiers to be used in `<pattern>` for `config checkFMT`	
`%j`	job ID
`%s`	job status
`%c`	command used to submit the job
`%d`	working directory of the job
`%n`	node allocated for the job
`%N`	number of cores of the job
`%Q`	time when the job was submitted
`%S`	time when the job started to run
`%E`	time when the job has finished
`%R`	the time in queue, or the running time, or the total running time (depending on the status of the job)

Syntax for sets of job IDs
`<initial ID>-<final ID>,<ID1> <ID2>`
Example: `10-14,4,30 20`, means
10, 11, 12, 13, 14, 4, 30, and 20