## Managing applications states

With PM2 you can easily start/restart/reload/stop/list applications in background.

#### Start

To start an application:

```
pm2 start http.js
[PM2] Starting /var/examples/http.js in fork mode (1 instance)
                                version
                                                     pid
                                                                uptime
                                                                                                                               watching
 id
                 namespace
                                          mode
                                                                                status
       name
                                                                         5
                                                                                             cpu
                                                                                                        mem
                                                                                                                    user
                                          fork
 0
       http
                 default
                                N/A
                                                     48553
                                                                         0
                                                                                 online
                                                                                                        17.7mb
                                                                                                                    unitech
                                                                0s
                                                                                             0%
                                                                                                                               disabled
```

You can also start any kind of application like bash commands, script, binaries:

\$ pm2 start "npm run start" \$ pm2 start "is -la" \$ pm2 start app.py

#### Start and display log stream

To start an app and check logs stream use the --attach option:

\$ pm2 start api.js --attach

When quitting via Ctrl-C, the app will still run in background.

#### Passing arguments

All option passed after -- will be passed as argument to the app:

\$ pm2 start api.js -- arg1 arg2

#### Configuration File

When managing multiple application at the same time or having to specify multiple options, you can use a configuration file. Example with this ecosystem.config.js file:

module exports = {
 apps : [{
 init worker\*,
 script : "./worker.js",
 args : "limit"
 },{
 name : "rotate worker\*,
 script : "./worker.js",
 args : "rotate"
}]

Then to start both apps:

\$ pm2 start ecosystem.config.js

Read more about configuration file

# Restart

To restart an application:

To restart all applications:

To restart multiple apps at once:

\$ pm2 restart app1 app3 app4

### Updating environment variables and options

To update environment variables or PM2 options, specify the --update-env CLI option:

\$ NODE\_ENV=production pm2 restart web-interface --update-env

#### Stop

To stop a specified application:

\$ pm2 stop api
\$ pm2 stop [process\_id]

To stop them all:

To stop multiple apps at once:

\$ pm2 stop app1 app3 app4

Note: this will not delete the application from PM2 application list. See next section to delete an application.

#### Delete

To stop and delete an application:

To delete them all:

## Listing Applications

To list all running applications:

\$ pm2 list
# Or
\$ pm2 [list|ls|l|status]

> pm2 ls

id	name	namespace	version	mode	pid	uptime	J	status	сри	mem	user	watching
2	autoexit	default	N/A	fork	0	0	0	stopped	0%	0b	unitech	disabled
0	http	default	N/A	fork	38914	26m	0	online	0%	42.9mb	unitech	disabled
3	watch ls	default	N/A	fork	41366	7m	0	online	0%	2.9mb	unitech	disabled

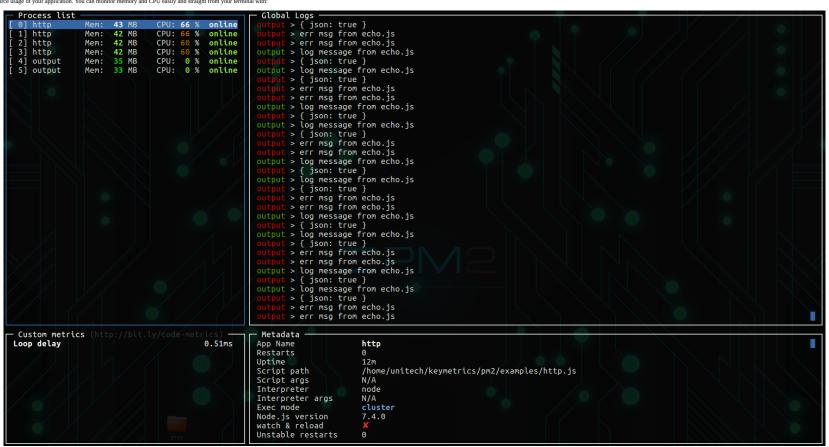
To specify which order you want the application to be listed:

\$ pm2 list --sort name:desc

\$ pm2 list --sort [name|id|pid|memory|cpu|status|uptime][:asc|desc]

#### **Terminal Dashboard**

PM2 gives you a simple way to monitor the resource usage of your application. You can monitor memory and CPU easily and straight from your terminal with:



Showing application metadata

To display metadata about an application:

```
> pm2 show 0
Describing process with id 0 - name http
   status
                                   http
default
N/A
  name
namespace
version
   restarts
  uptime
   script path
                                    /var/examples/http.js
N/A
  script path
script args
error log path
out log path
pid path
interpreter
                                     /home/unitech/.pm2/logs/http-error.log
/home/unitech/.pm2/logs/http-out.log
/home/unitech/.pm2/pids/http-0.pid
                                    node
N/A
   interpreter args
script id
   exec cwd
                                     /var/examples
  exec mode
node.js version
                                     fork_mode
16.2.0
  node env
watch & reload
unstable restarts
created at
                                     development
                                     2021-06-26T10:49:23.764Z
```

## Actions available

km:heapdump km:cpu:profiling:start km:cpu:profiling:stop km:heap:sampling:start km:heap:sampling:stop

Trigger via: pm2 trigger http <action name>

### Code metrics value

Heap Size 10.14 MiB Heap Usage 64.48 % Used Heap Size 6.54 MiB

## Reset restart count

To reset the restart counter:

Contribute to this page