
tcod-clock

Release 1.0.1

Kyle Benesch

Apr 01, 2024

CONTENTS:

| | | |
|----------|----------------------------|----------|
| 1 | About | 1 |
| 2 | API reference | 3 |
| 3 | Indices and tables | 5 |
| | Python Module Index | 7 |
| | Index | 9 |

ABOUT

Libtcod used to include a global framerate limiter which was eventually deprecated. This module was created as a replacement for that feature.

```
import time

import tcod.clock

FPS = 30

end_time = time.time() + 3 # Loop for 3 seconds.

clock = tcod.clock.Clock()
while time.time() < end_time:
    clock.sync(1 / FPS) # This loop will run at 30 FPS until interrupted.

# Timing information can be checked. Check the docs for more info.
print(f"{clock.last_fps}")
print(f"{clock.min_fps}")
print(f"{clock.max_fps}")
print(f"{clock.mean_fps}")
print(f"{clock.median_fps}")
```


API REFERENCE

Track and limit framerate of a program.

class `tcod.clock.Clock`

Bases: `object`

Measure framerate performance and sync to a given framerate.

Everything important is handled by `Clock.sync`. You can use the `fps` properties to track the performance of an application.

Time is sampled with `time.perf_counter`.

Example:

```
import tcod.clock

clock = tcod.clock.Clock()
while True:
    clock.sync(1 / 30)  # This loop will run at 30 FPS until interrupted.
```

property `last_fps`: `float`

The FPS of the most recent frame.

property `last_frame`: `float`

The length of the most recent frame.

last_time

Last time this Clock was synced.

property `max_fps`: `float`

The FPS of the fastest frame.

max_samples = 64

Number of framerate samples to log. This attribute be set in the class or instance.

property `mean_fps`: `float`

The FPS of the sampled frames overall.

property `median_fps`: `float`

The FPS of the median frame.

property min_fps: `float`

The FPS of the slowest frame.

sync(*desired_framerate=None*)

Sync to a given framerate and return the delta time.

Parameters

desired_framerate (`float` / `None`) – The desired framerate in seconds. If `None` is given then this function will track the time and framerate without ever waiting. Must be above zero when not `None`.

Returns

The delta time since the last call to `sync`, in seconds.

Return type

`float`

time_samples

A recent collection of delta-time samples.

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

t

`tcod.clock`, 3

INDEX

C

`Clock` (*class in `tcod.clock`*), 3

L

`last_fps` (*`tcod.clock.Clock` property*), 3

`last_frame` (*`tcod.clock.Clock` property*), 3

`last_time` (*`tcod.clock.Clock` attribute*), 3

M

`max_fps` (*`tcod.clock.Clock` property*), 3

`max_samples` (*`tcod.clock.Clock` attribute*), 3

`mean_fps` (*`tcod.clock.Clock` property*), 3

`median_fps` (*`tcod.clock.Clock` property*), 3

`min_fps` (*`tcod.clock.Clock` property*), 3

`module`
 `tcod.clock`, 3

S

`sync()` (*`tcod.clock.Clock` method*), 4

T

`tcod.clock`
 module, 3

`time_samples` (*`tcod.clock.Clock` attribute*), 4