

---

# Tiny-Thingy

Jul 02, 2018



---

## Contents

---

<b>1</b>	<b>Install</b>	<b>3</b>
<b>2</b>	<b>Examples</b>	<b>5</b>
2.1	Setup TinyDB database . . . . .	5
2.2	Insert and find thingies . . . . .	5
<b>3</b>	<b>Documentation</b>	<b>7</b>
3.1	API Reference . . . . .	7
<b>4</b>	<b>Indices and tables</b>	<b>9</b>
	<b>Python Module Index</b>	<b>11</b>



Tiny-Thingy is a wrapper around [TinyDB](#) that makes it more Pythonic.



# CHAPTER 1

---

## Install

---

```
$ pip install tiny-thingy
```





### 2.1 Setup TinyDB database

```
>>> from tiny_things import Thingy, use_database
>>> use_database("/some/json/file/serving/as/database.json")

>>> class Task(Thingy):
...     pass
```

The *Task* class will represent the *task* table and instances of it are documents in this table.

### 2.2 Insert and find thingies

```
>>> task = Task({"name": "work on tiny-thingy"}).save()
>>> Task.count()
1
>>> Task.find_one()
Task({'doc_id': 1, 'name': 'work on tiny-thingy'})
```



## 3.1 API Reference

```
class tiny_thingsy.Thingy (*args, **kwargs)
    Represents a JSON object in a table

classmethod count ()
    Count the number of objects in the table

classmethod create (obj)
    Directly insert an object in the table

delete ()
    Delete the object from the table

classmethod find (query=None, doc_ids=None)
    Return a list of objects matching the query

classmethod find_one (query=None, doc_id=None)
    Return a single object matching the query

classmethod inplace_update (fields, query=None, doc_ids=None)
    Directly update objects matching the query with given fields

classmethod remove (query=None, doc_ids=None)
    Remove the objects matching the query

save ()
    Save the object in the table

classmethod use_database (filename)
    Initialize a database for the class using the specified filename
```



## CHAPTER 4

---

### Indices and tables

---

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



**t**

`tiny_thingy`, [7](#)





## C

`count()` (`tiny_thingy.Thingy` class method), [7](#)  
`create()` (`tiny_thingy.Thingy` class method), [7](#)

## D

`delete()` (`tiny_thingy.Thingy` method), [7](#)

## F

`find()` (`tiny_thingy.Thingy` class method), [7](#)  
`find_one()` (`tiny_thingy.Thingy` class method), [7](#)

## I

`inplace_update()` (`tiny_thingy.Thingy` class method), [7](#)

## R

`remove()` (`tiny_thingy.Thingy` class method), [7](#)

## S

`save()` (`tiny_thingy.Thingy` method), [7](#)

## T

`Thingy` (class in `tiny_thingy`), [7](#)  
`tiny_thingy` (module), [7](#)

## U

`use_database()` (`tiny_thingy.Thingy` class method), [7](#)