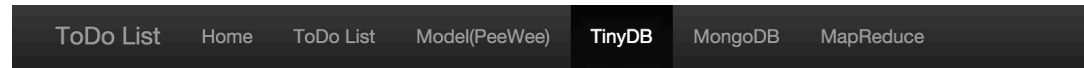# ADSD HW7 TinyDB ToDo

I use TinyDB CRUD both on local and pythonanywhere, to keep original peewee ORM working, I changed all TinyDB related request method.

Pythonanywhere:

Create:

ToDo List	Home	ToDo List	Model(PeeWee)	**TinyDB**	MongoDB	MapReduce

Model List View

The open items are as follows(Model List):

The open items are as follows:

Seq	Description	Marker	Delete
6	Install Xcode	0	🗑
15	Install Cocoapods	1	🗑

Enter a new item...

To be done: initial Podfile   Save new item...

Update:

ToDo List	Home	ToDo List	Model(PeeWee)	**TinyDB**	MongoDB	MapReduce
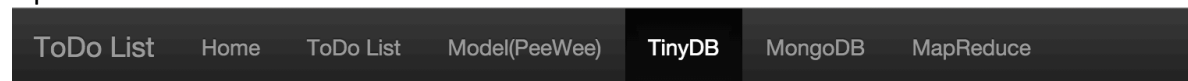
Edit the task with ID = 6

Install Xcode

open

save

Read:

Model List View

The open items are as follows(Model List):

The open items are as follows:

Seq	Description	Marker	Delete
6	Install Xcode	0	🗑
15	Install Cocoapods	1	🗑

Enter a new item...

To be done: `initial Podfile` [Save new item...]

Delete:

Delete the task with ID = 6

Task to delete:

Install Xcode

[Confirm deletion...]

Code:

```python
#tinyDB

@route('/tinydb')
@route('/tinydb/<statis:int>')
def tinydb(status=-1):
    items = tiny_get_items(status)
    result = []
    for item in items:
        result.append((item['id'],item['task'],item['status']))
    return template('tinydb', rows=result)

@get('/tinydb/new')
def get_new():
    return '''
        <p>Enter a new item...</p><br/>
        <form action="/tinydb/new" method="post">
            To be done: <input name="task" type="text" />
            <input value="Save" type="submit" />
        </form>      '''

@post('/tinydb/new')
def post_new():
    task = request.forms.get('task', '').strip()
    tiny_new_item(task,1)
    return tinydb()

@get('/tinydb/edit/<id:int>')
def get_edit(id):
    item = tiny_get_item(id)
    return template('tinydb_edit', id=id, task=item['task'], status=item['status'])

@post('/tinydb/edit/<id:int>')
def post_edit(id):
    task = request.forms.get('task', '').strip()
    status = request.GET.get('status','').strip()
    if status == 'open':
        status = 1
    else:
        status = 0
    item = tiny_get_item(id)
    item['task'] = task
    item['status'] = status
    tiny_save_item(item)
    return tinydb()

@get('/tinydb/delete/<id:int>')
def confirm_delete_item(id):
    item = tiny_get_item(id)
    return template('tinydb_delete_view', id=id, task=item['task'], status=item['status'])

@post('/tinydb/delete/<id:int>')
def delete_item(id):
    tiny_discard_item(id)
    return tinydb()
```