Rest Framework – API
Python
Assessment Test

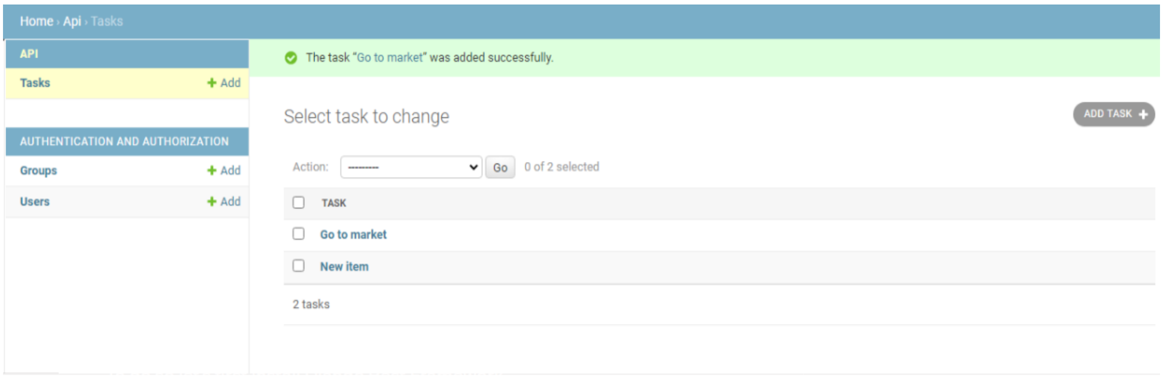
- **Write a project to demonstrate todo app api using of Django Rest framework**

- **Prepare demonstration of Python todo app api under software development principles and follow coding protocols**

- **The project was built in a MVT concept create an effective interactive interface.**

- **In this application Perform all operations under function based views**

- **Create a task model in the database for the todo application that the Django ORM will manage.**



The task "Go to market" was added successfully.

Select task to change

Action: --------- 0 of 2 selected

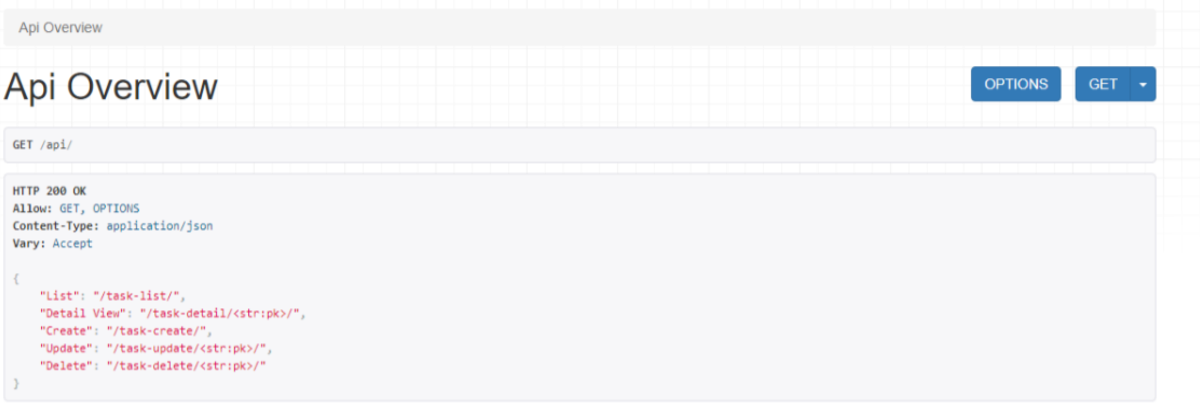
□ TASK

□ Go to market

□ New item

2 tasks

- **Make one api for which represent all APIs overview - need to make api which perform crud operations as follows :**



Api Overview

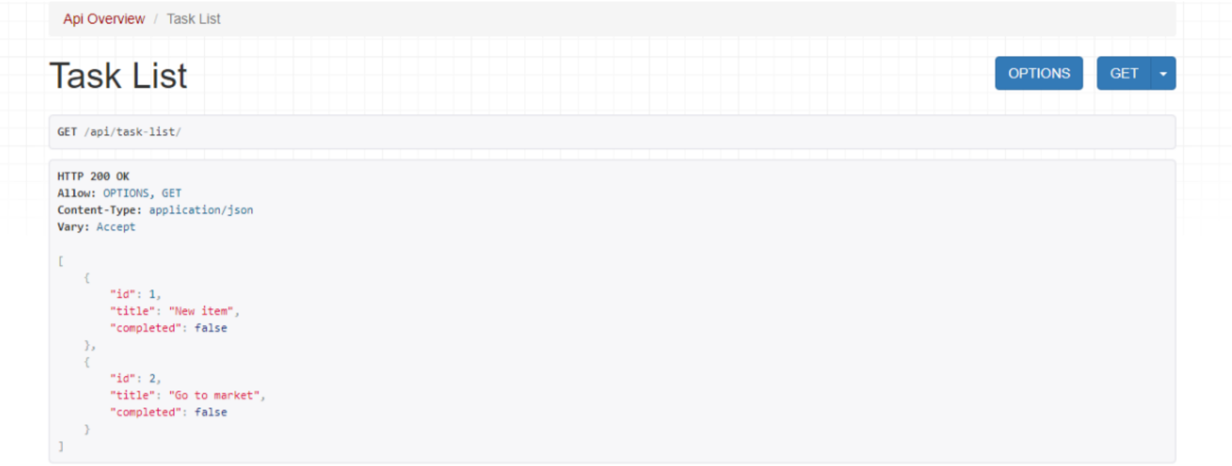
```
GET /api/

HTTP 200 OK
Allow: GET, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "List": "/task-list/",
    "Detail View": "/task-detail/<str:pk>/",
    "Create": "/task-create/",
    "Update": "/task-update/<str:pk>/",
    "Delete": "/task-delete/<str:pk>/"
}
```

- **Make sure all api data return in json format – create separate file of json serializes**

- **These are a few key options for a REST API request:**
- **GET — the most common option, returns some data from the API based on the endpoint you visit and any parameters you provide.**
- **POST — creates a new record that gets appended to the database.**
- **PUT — Looks for a record at the given URI you provided. If the exists, update the existing record. If not, create a new record.**
- **DELETE — Deletes the record at the given URI**
- **PATCH — Update the Individual field of the record**



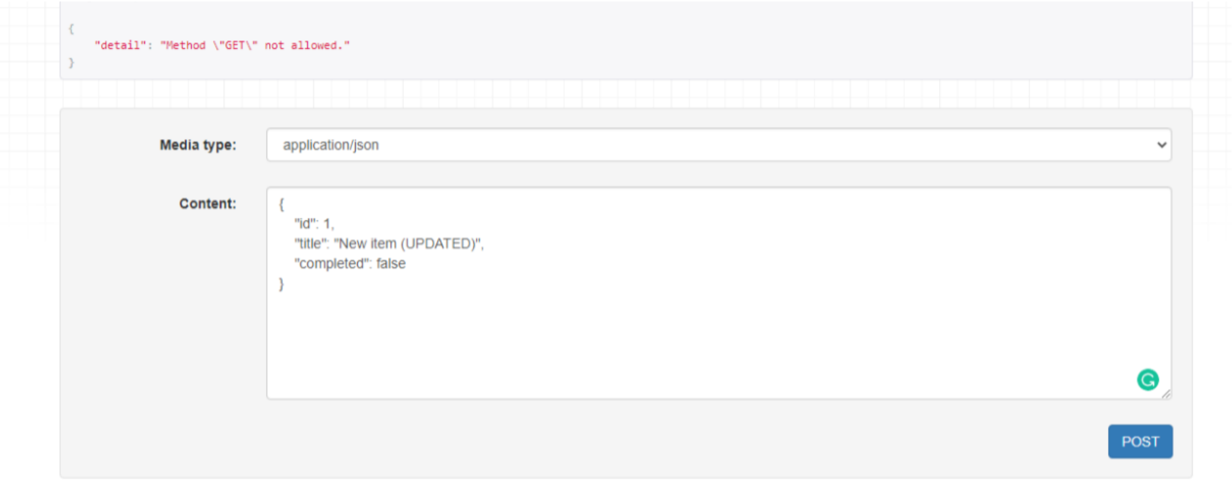
Task List

```
GET /api/task-list/
```

```
HTTP 200 OK
Allow: OPTIONS, GET
Content-Type: application/json
Vary: Accept

[
    {
        "id": 1,
        "title": "New item",
        "completed": false
    },
    {
        "id": 2,
        "title": "Go to market",
        "completed": false
    }
]
```

- **Create an Api which can accept data from user**



```
{
    "detail": "Method \"GET\" not allowed."
}
```

Media type: application/json

Content:

```
{
    "id": 1,
    "title": "New item (UPDATED)",
    "completed": false
}
```

- **Admin can able to delete existing records of this system**

- **Manage proper naming conversion – create appropriate objects name.**

- **create separate file for all business logics and make them reusable - use modules concepts for implements above logic**

- **Make sure code prevent from unexpected exception return to the previous screen and accept all details again.**

- **Make sure Database normalize manage in this project work**

- **Developer needs to test this product before launching it into the market**

- **After completion this project upload it on GitHub**
    - **Upload all features in develop branch after completion all features merge it with main branch**