4 Documentation

Case 1

```
Method: POST
Endpoint: http://127.0.0.1:5000/api/directors/
Auth credentials: No authentication case in this chapter!
Example 1:
Request:
{
     "first_name": "james",
     "last_name": "cameron"
}
Response:
  "code": "success",
  "director": {
     "created": "2021-05-16 07:34:08",
    "first_name": "james",
    "id": 1.0,
    "last_name": "cameron",
    "movies": []
  }
}
Example 2:
Request:
  "first_name": "Christopher",
  "last_name": "Nolan"
}
```

```
Response:
{
  "code": "success",
  "director": {
     "created": "2021-05-16 07:38:57",
     "first_name": "Christopher",
     "id": 2.0,
     "last_name": "Nolan",
     "movies": []
  }
}
Case 2:
Method: GET
Endpoint: http://127.0.0.1:5000/api/directors/
Response:
  "code": "success",
  "directors": [
        "first_name": "james",
        "id": 1.0,
        "last_name": "cameron"
     },
        "first_name": "Christopher",
        "id": 2.0,
        "last_name": "Nolan"
     }
  ]
}
Case 3:
Method: GET
Endpoint: <a href="http://127.0.0.1:5000/api/directors/1">http://127.0.0.1:5000/api/directors/1</a>
  "code": "success",
```

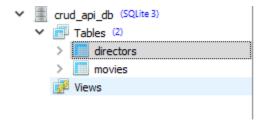
```
"director": {
    "created": "2021-05-16 07:34:08",
    "first_name": "james",
    "id": 1.0,
    "last_name": "cameron",
    "movies": []
  }
```

Next cases:

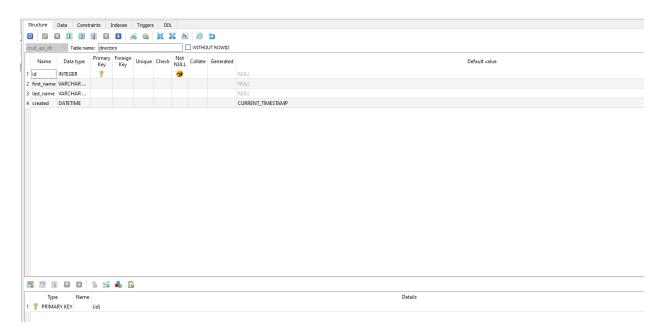
In this chapter only POST method and get method is used. In the upcoming chapters other methods like put, delete and patch methods are created.

In the upcoming chapters movies table will be filled. Also authentication will be added!

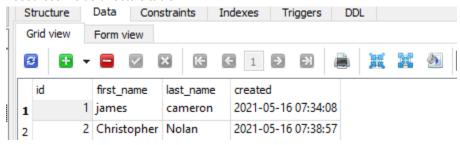
Appendix:



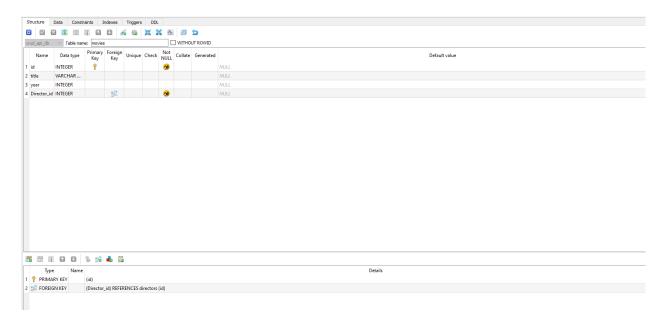
Directors table structure



Resources inside directors table



movies table structure:



movies table resources:

