Topic: 9. CRUD Operations using Mixins

Speaker: Personal | Notebook: API Development using Django Framework

---

Mixins are a way to allow the reusability of methods. In object-oriented programming, mixin is a class with methods from other classes.

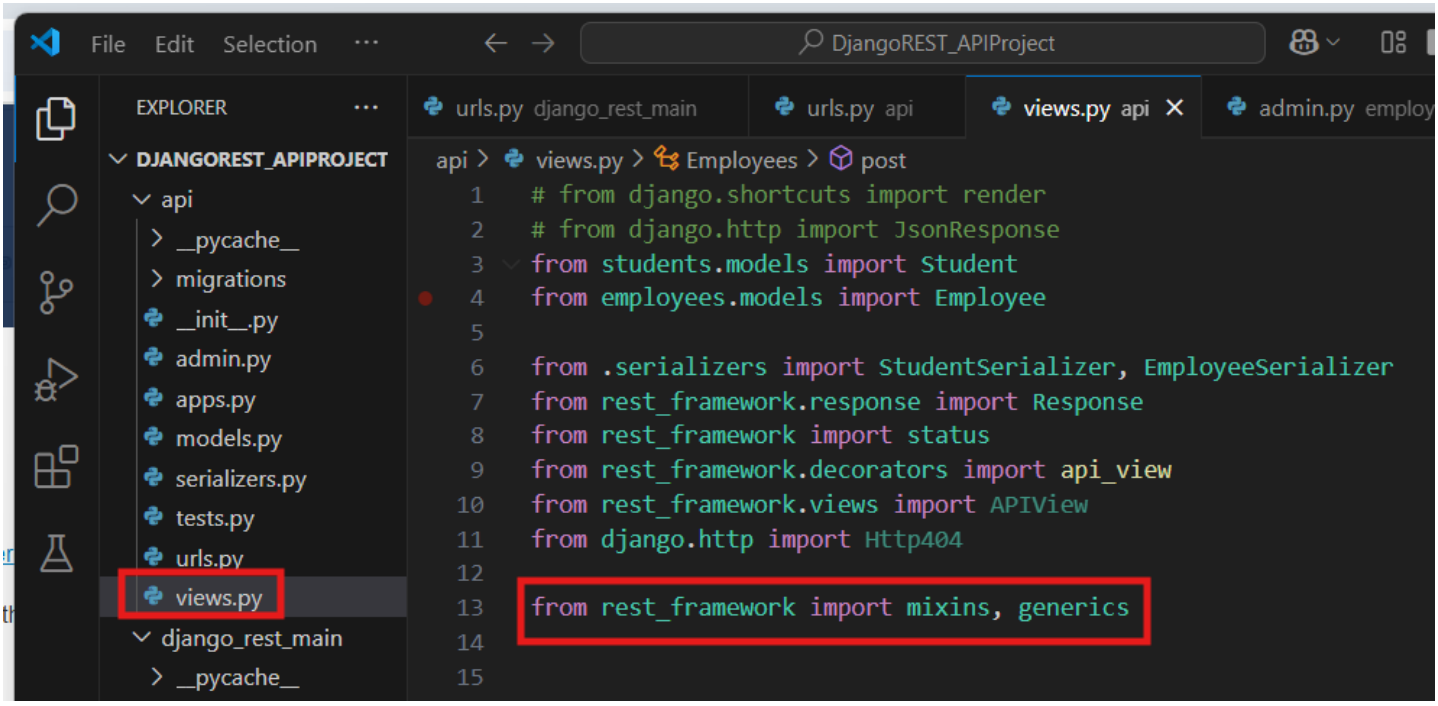
With mixins, we can reuse certain CRUD operations. These are the following:

```
LISTMODELMIXIN - method used is list()

CREATMODELMIXIN - method used is create()

RETRIEVEMODELMIXIN - - method used is retrieve()

UPDATEMODELMIXIN - method used is update()

DESTROYMODELMIXIN - method used is destroy()
```

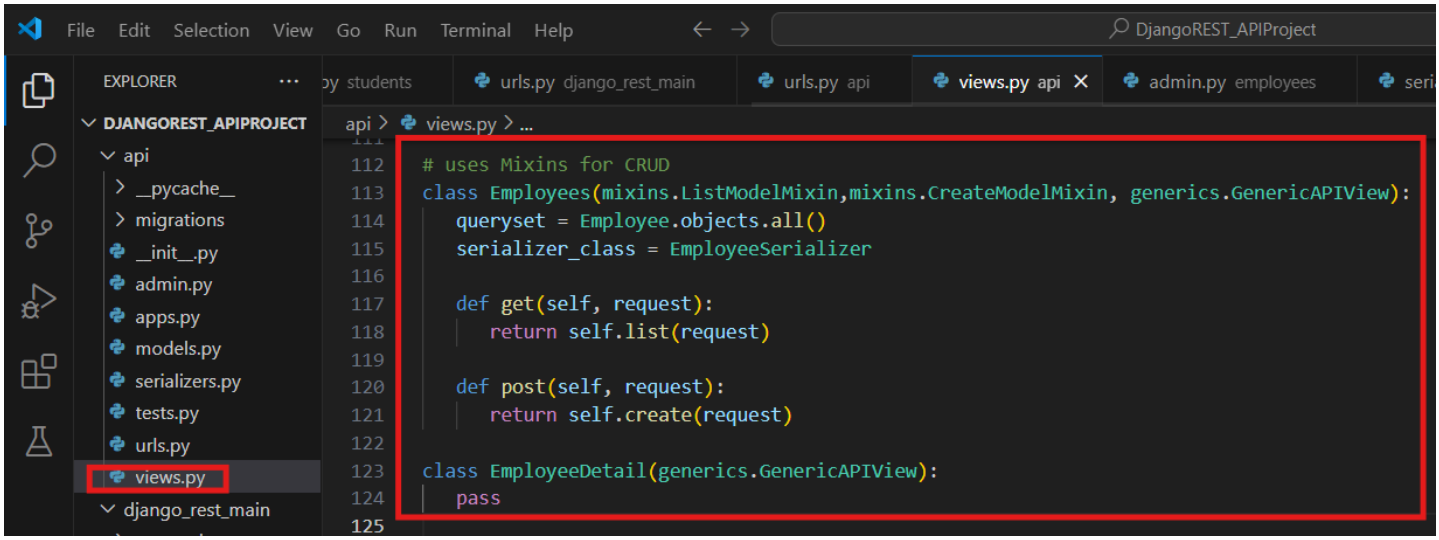
These are used with GenericAPIView. See the documentation here.

1. To view all the records of Employees using Mixins, we update the API\VIEWS.PY and comment out the classes EMPLOYEE AND EMPLOYEE DETAIL.

2. Import the needed library.

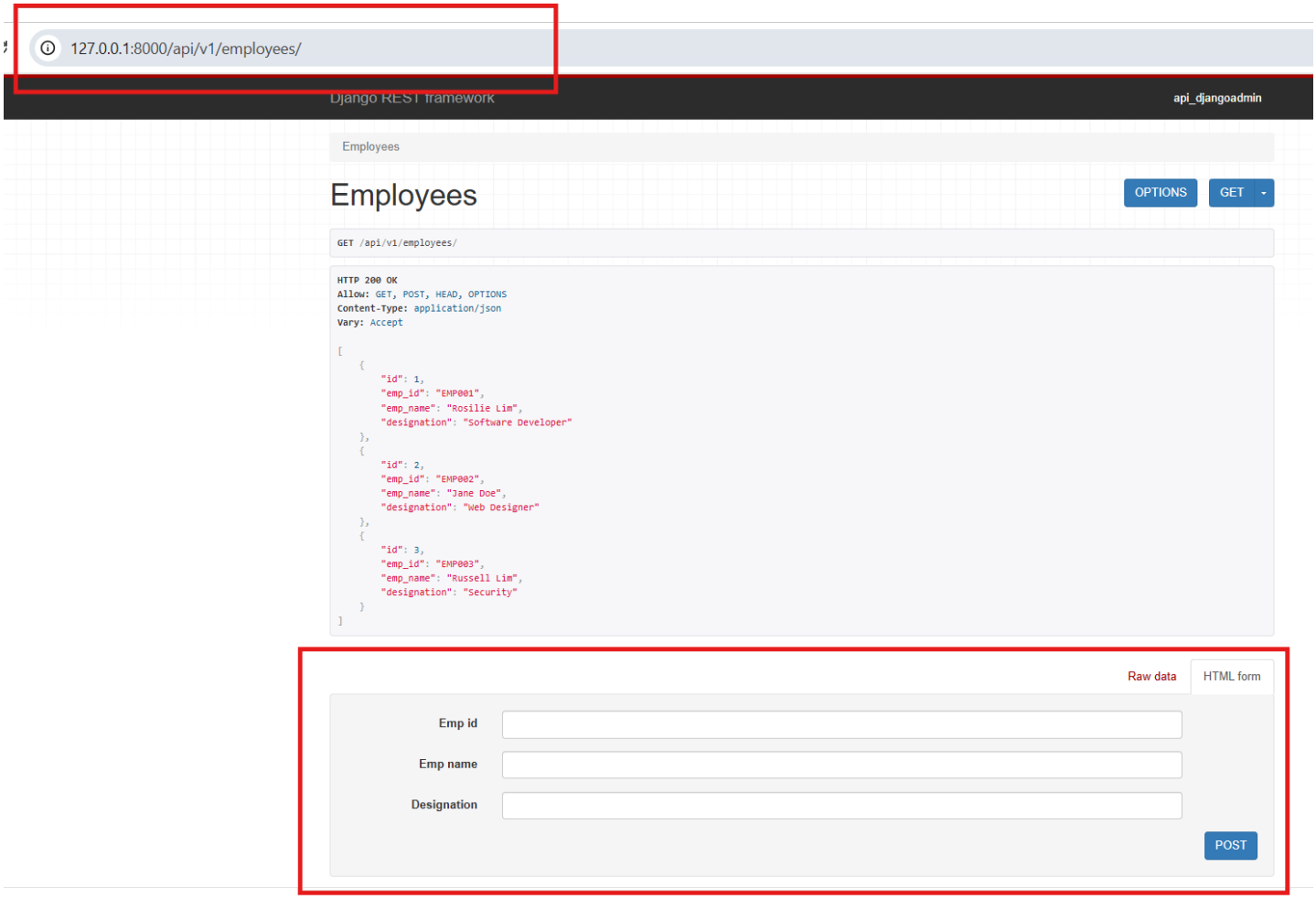


3. Create the new classes and test the URL.

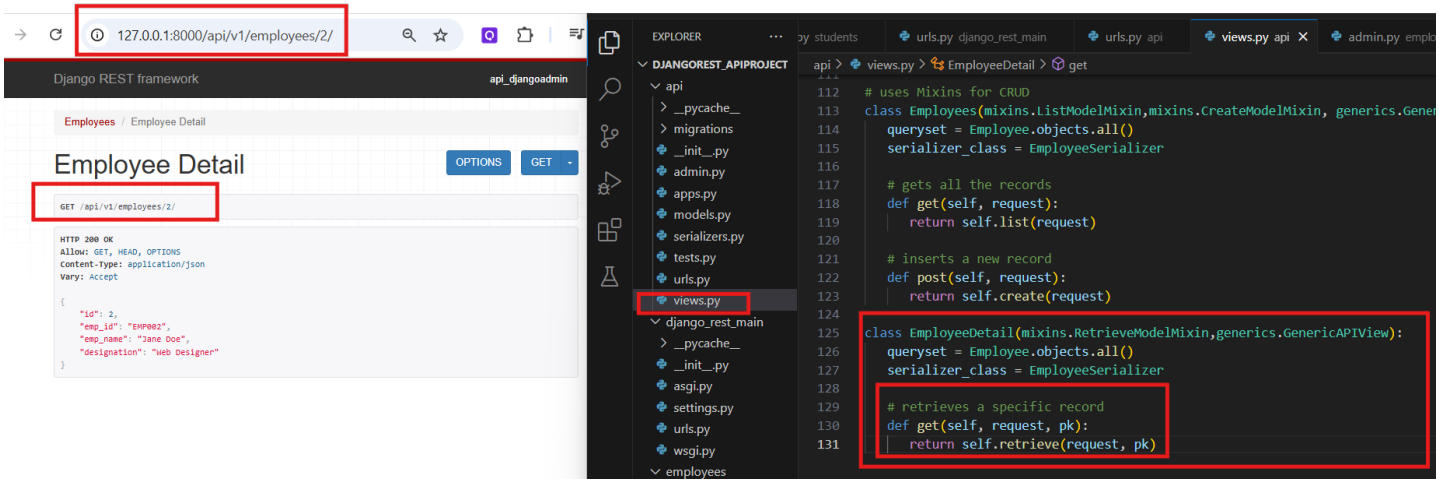


4. Testing the URL: `http://127.0.0.1:8000/api/v1/employees/`

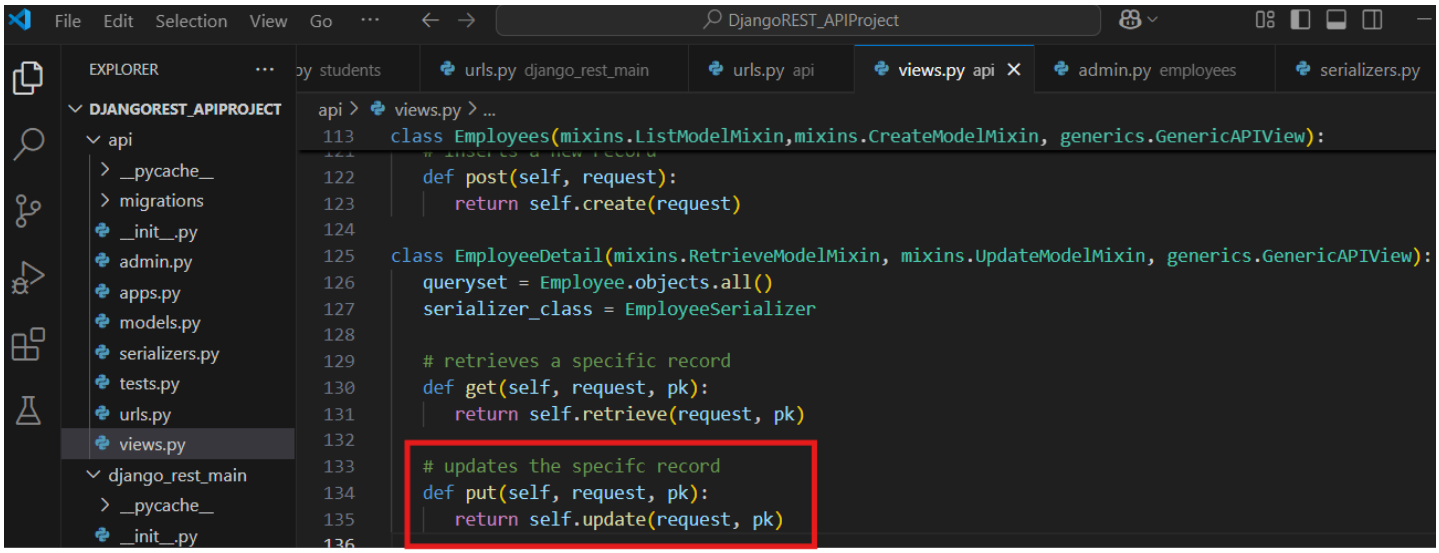
This will result to having a form where we can input employee details and when submitted, this will be added to our Employee model.



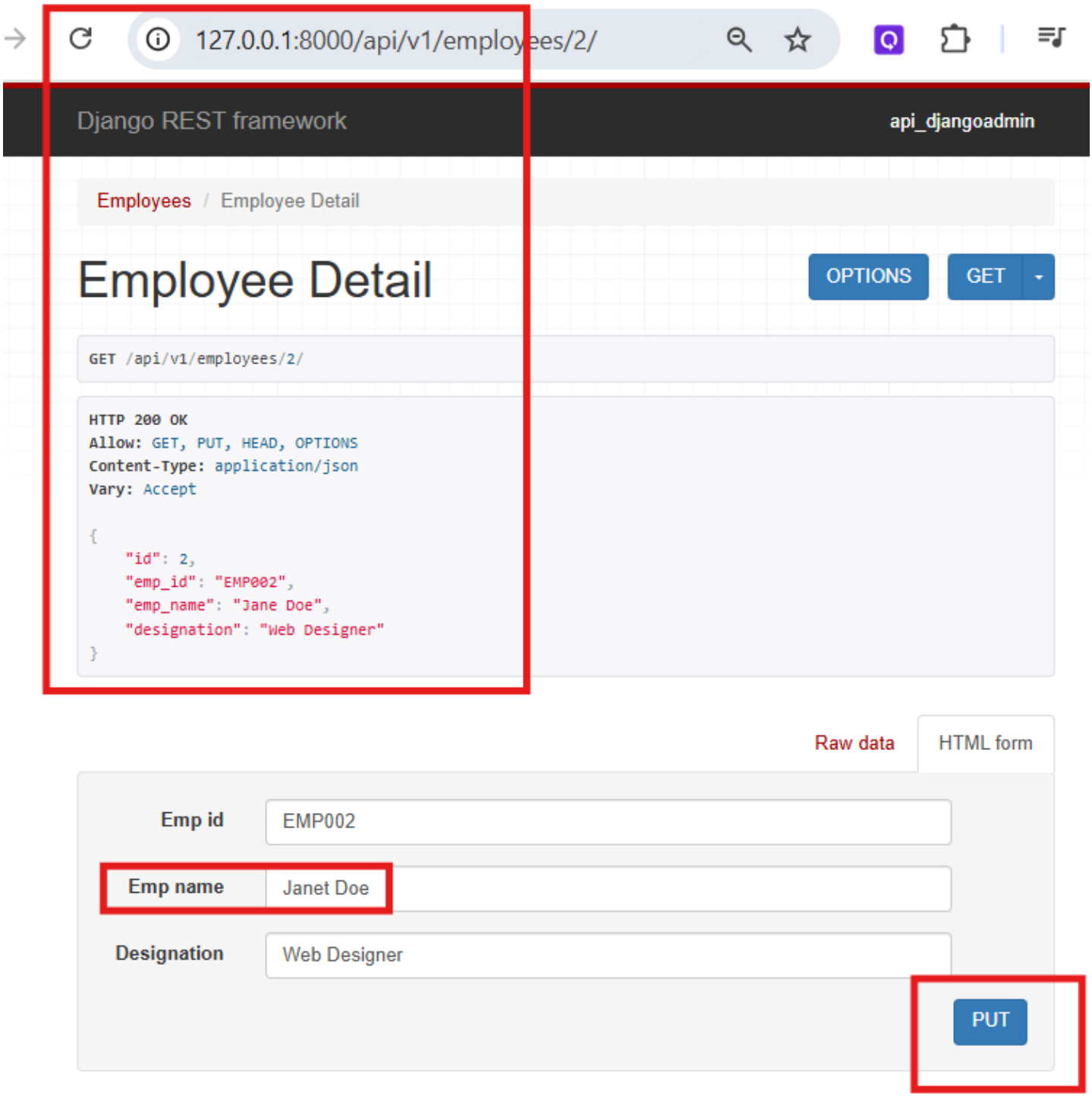
5. For single - record Read/Update/Delete operations, we update our API\APPS.PY as :



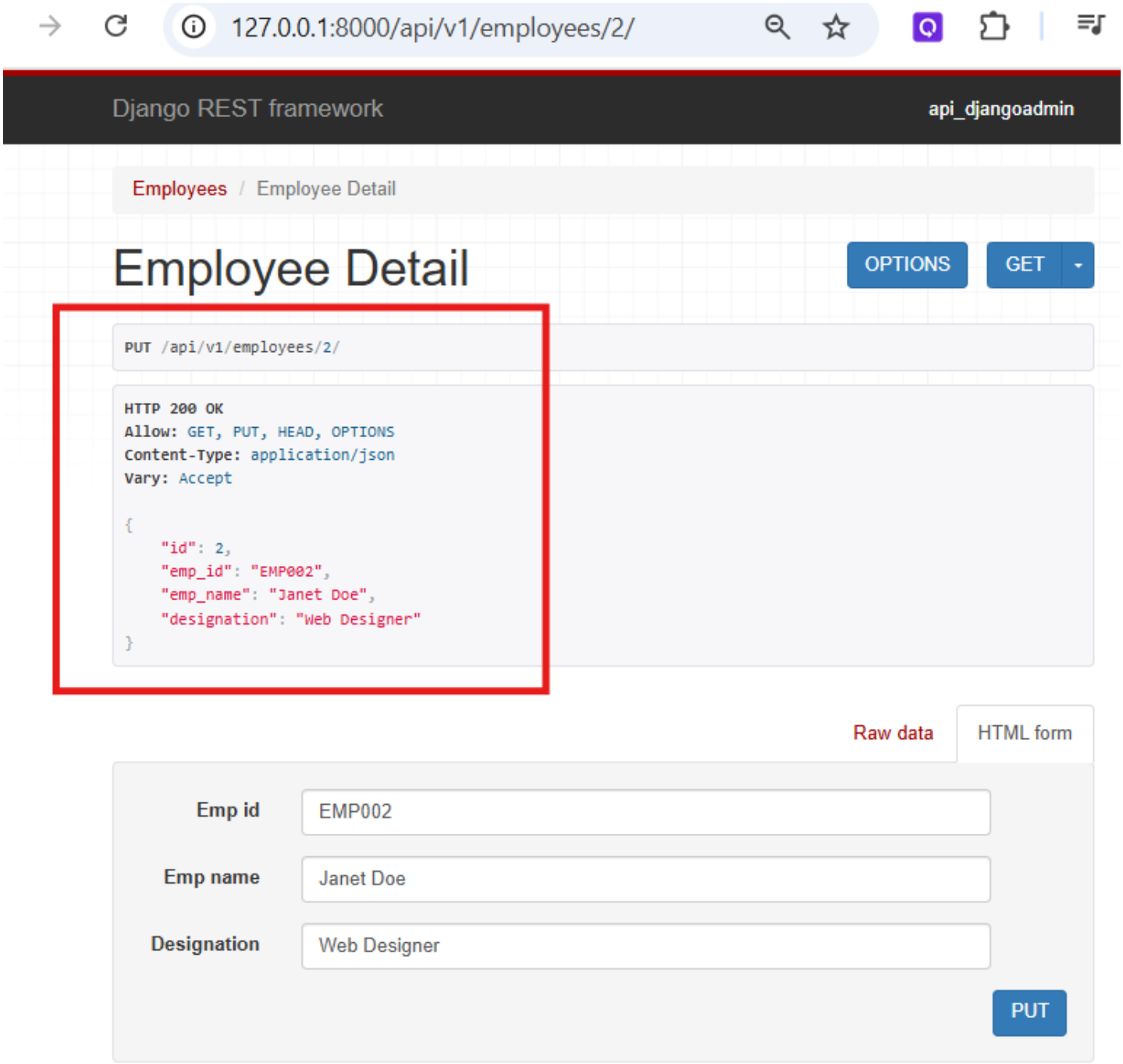
6. To update a specific record, we use the UPDATEMODELMIXIN.



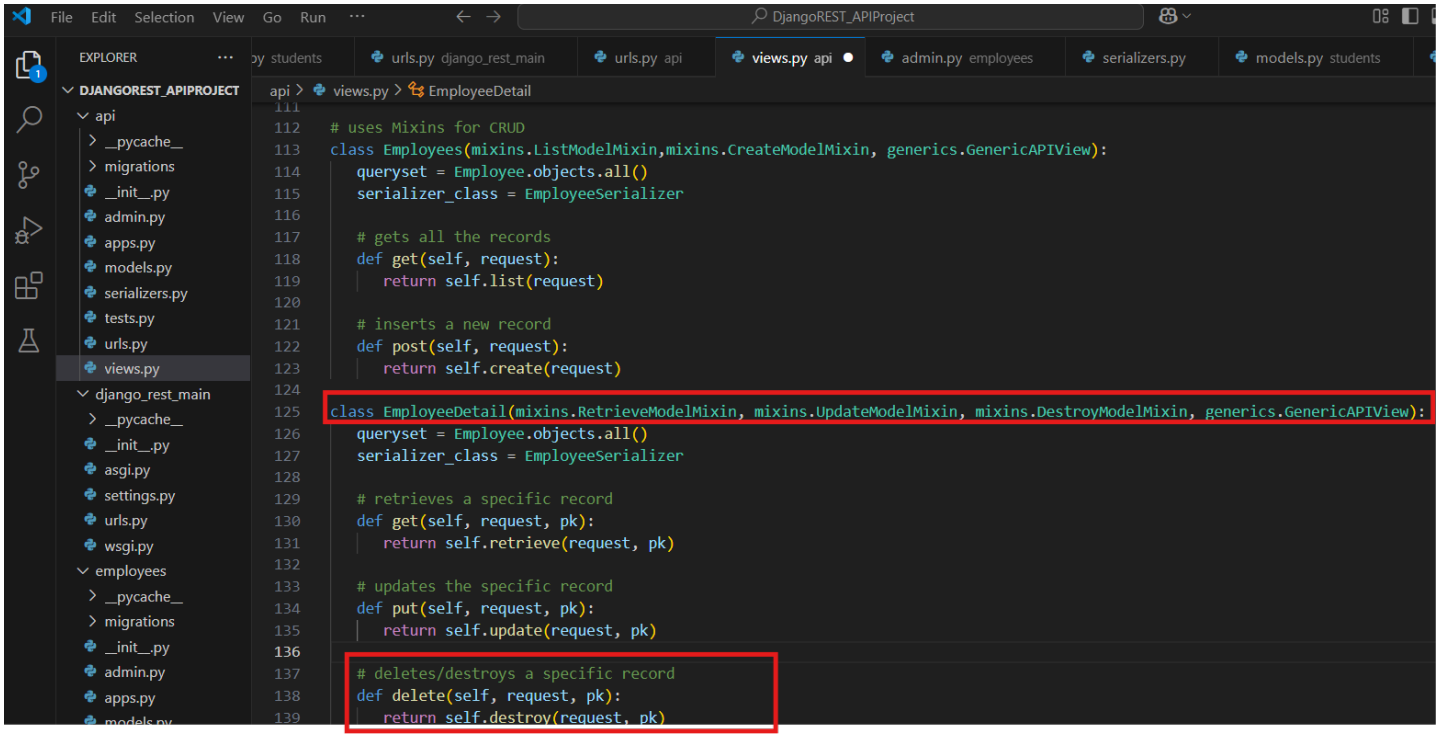
To test:



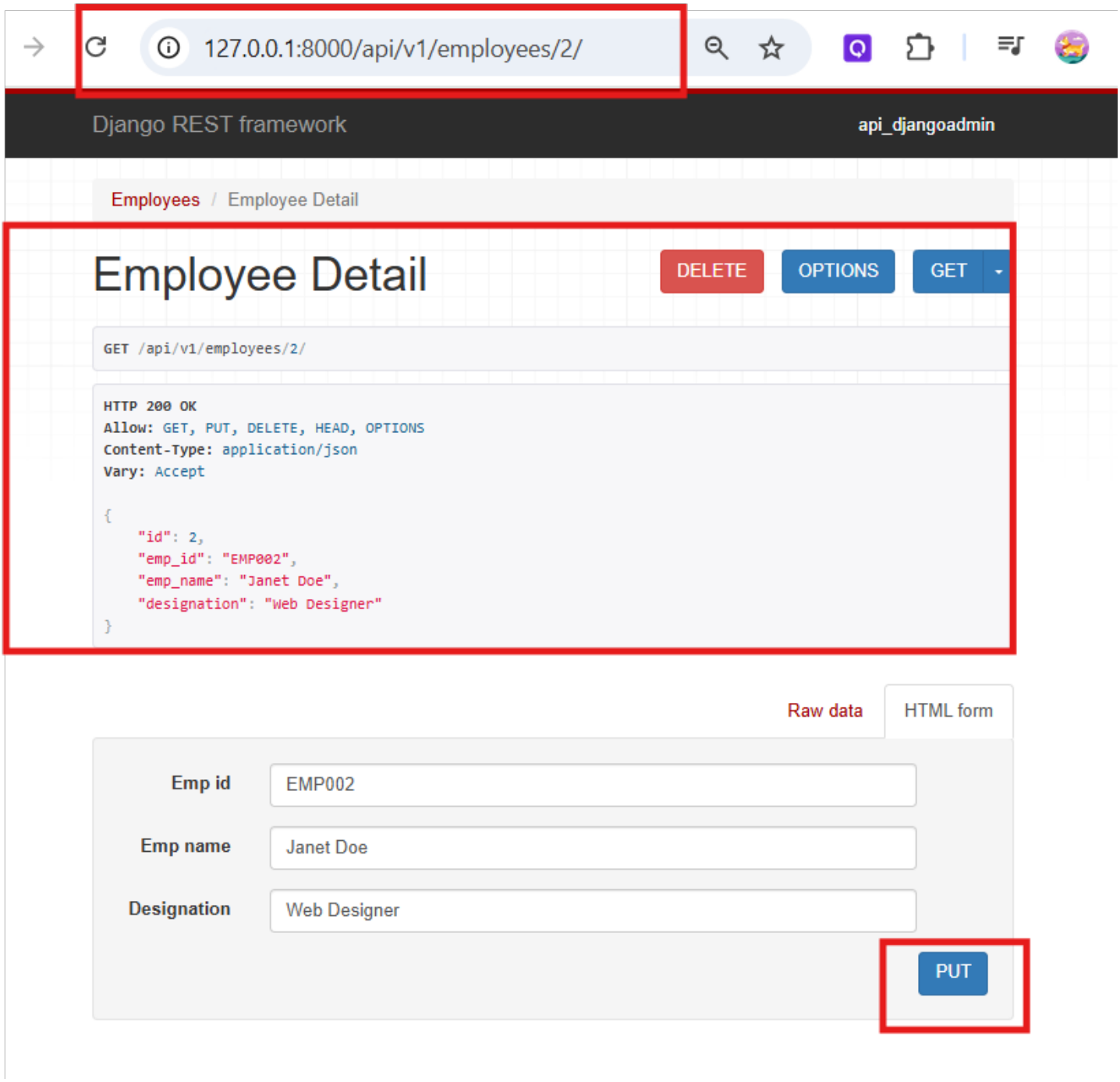
This will update the record to:

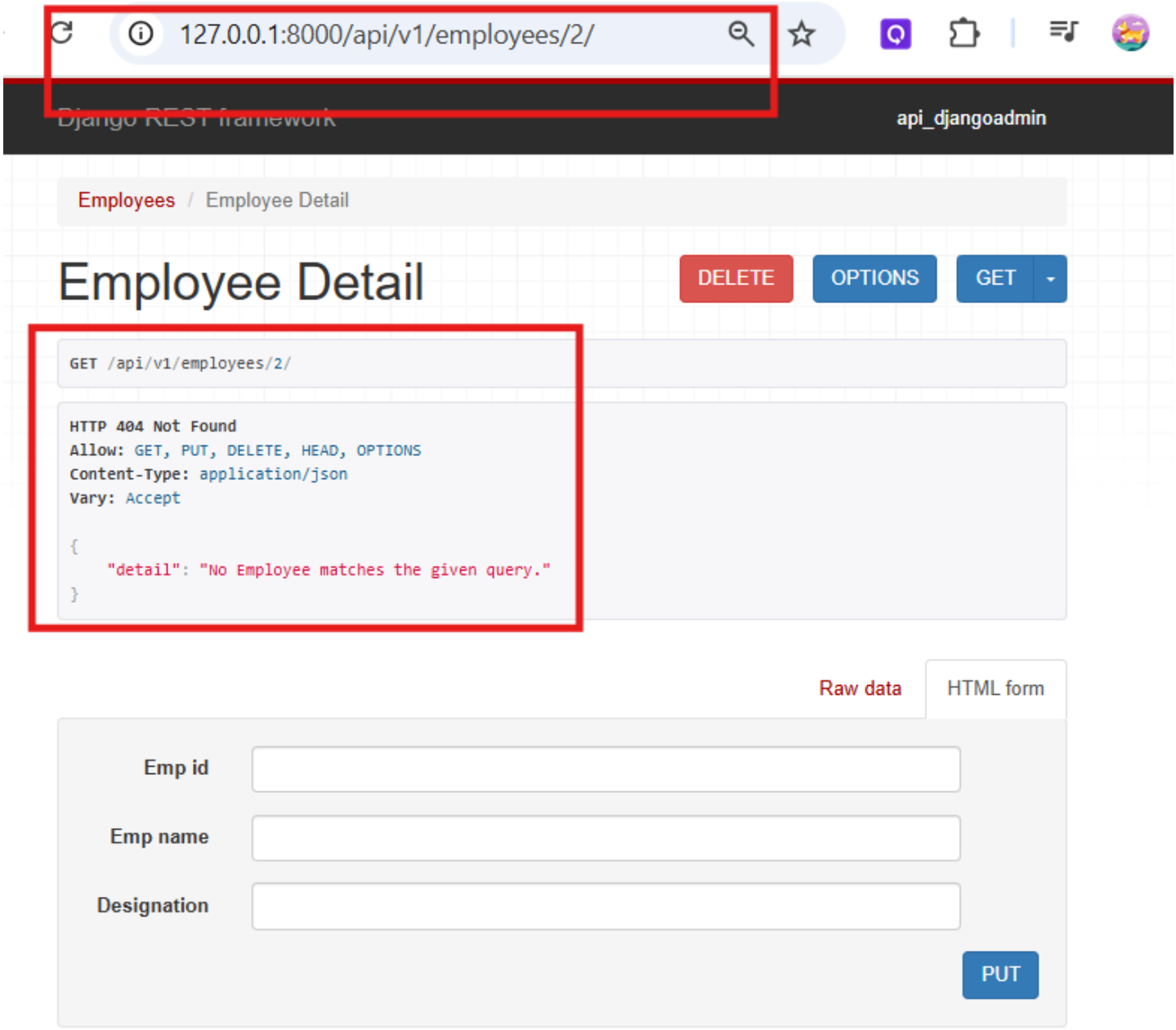


7. To delete a specific record.



8. To test and delete record 2,


127.0.0.1:8000/api/v1/employees/2/
Django REST framework
api_djangoadmin
Employees / Employee Detail
Employee Detail
DELETE
OPTIONS
GET
GET /api/v1/employees/2/
HTTP 200 OK
Allow: GET, PUT, DELETE, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept
{
"id": 2,
"emp_id": "EMP002",
"emp_name": "Janet Doe",
"designation": "Web Designer"
}
Raw data
HTML form
Emp id
EMP002
Emp name
Janet Doe
Designation
Web Designer
PUT



9.