

Lending API – Django

In this challenge you have to create a Rest API to manage the Lending data using Django.

Step – 1:

To install all the required packages. Open Terminal, go to **kickoffs-django-Lending-api** directory, type “**python3 -m pip install -r requirements.txt**” (or) “**sh install.sh**”.

Step –2:

Models.py- Create models with appropriate fields as mentioned below

Borrower:

Column Name	Constraints	Data Type	Comments
name		String	
email		email	Unique=True

Loan:

Column Name	Constraints	Data Type	Comments
id	Primary key	Integer	
amount		Float	
Interest_rate		Float	
Borrower_id	Foreign Key	Integer	One to many relationships with borrower model

Payment:

ColumnName	Constraints	DataType	Comments
id	Primary_key	Integer	
amount		Float	
date		Date	Default=" today date"
Loan_id	Foreign Key	Integer	One to many relationships with loan model.

End Points:

In `lendingapi/urls.py` assign URL for your endpoints.

“/lends/borrower”:

Post Method:

Get all the details (name,email) from the borrower and store it in the borrower model, then return it as a response with status code 201.

Name

Email

POST

Request

```
POST /lends/borrower/

HTTP 201 Created
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
  "id": 1,
  "name": "john",
  "email": "john@gmail.com"
}
```

Response

Get Method:

Return all data from the borrower model as a response with status code 200.

```
GET /lends/borrower/

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

[
  {
    "id": 1,
    "name": "john",
    "email": "john@gmail.com"
  },
  {
    "id": 2,
    "name": "doe",
    "email": "doe@gmail.com"
  }
]
```

Response

“/lends/borrower/{id}/”:

Put Method:

Get the detail from the borrower to be patched and update the data in the crop object with given id and return it as a response with status code 200.

Delete Method:

Delete the crop object with given id from the crop model and return 204 as status code.

“/lends/loan/”:

Post Method:

Get details (amount,interest_rate,borrower_id) from the loan and store it in the loan model then return it as a response with status code 201.

```
POST /lends/loan/

HTTP 201 Created
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
  "id": 1,
  "amount": 2000.0,
  "interest_rate": 10.0,
  "borrower_id": 1
}
```

Response

Get Method:

Return all data from the loan model as a response with status code 200.

```
GET /lends/loan/

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

[
  {
    "id": 1,
    "amount": 2000.0,
    "interest_rate": 10.0,
    "borrower_id": 1
  }
]
```

Response

“lends/loan/{int:id}”:

Patch Method:

Get the detail from the loan to be patched and update the data in the loan object with given id and return it as a response with status code 200.

Delete Method:

Delete the loan object with given id from the loan model and return 204 as status code.

“lends/payment/”:

Post Method:

Get the details (amount,date,loan_id) from the payment and store it in the payment model then return it as a response with status code 201.

Amount	<input type="text" value="100"/>
Date	<input type="text" value="02/02/2023"/>
Loan id	<input type="text" value="1"/>
<input type="button" value="POST"/>	

Request

POST /lends/payment/

```
HTTP 201 Created
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
  "id": 1,
  "amount": 100.0,
  "date": "2023-02-02",
  "loan_id": 1
}
```

Response

Get Method:

Return all data from the payment model as a response with status code 200.

```
GET /lends/payment/

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

[
  {
    "id": 1,
    "amount": 100.0,
    "date": "2023-02-02",
    "loan_id": 1
  }
]
```

Response

“lends/payment/{int:id}”:

Patch Method:

Get the detail from the payment to be patched and update the data in the payment object with given id and return it as a response with status code 200.

Delete Method:

Delete the payment object with given id from the payment model and return 204 as status code.

Step-3:

Instructions:

Note: Before run the application make sure to run the following migration commands.

- Go to **kickoffs-django-lending-api** and type “**python3 manage.py makemigrations**” and type “**python3 manage.py migrate**” “**python3 manage.py runserver**”.
- **To test the application:** Open terminal go to **kickoffs-django-lending-api** folder, type as “**python3 manage.py test**”.

To preview the application: Open the browser and type <http://localhost:8000/> in address bar and hit enter.

Click **Submit** to complete the test