

# Student Informartion API (v1)

Download OpenAPI specification: [Download](#)

E-mail: [contact@crud.local](mailto:contact@crud.local) | License: [BSD License](#) | [Terms of Service](#)

API documentation for the CRUD project

## crud

### crud\_list

List all students

AUTHORIZATIONS: 

*Basic*

 **HTTP: Basic**

HTTP Authorization Scheme: `basic`

## Responses


 **200**

RESPONSE SCHEMA: `application/json`

Array [

 id

integer (ID)

 name  
*required*

string (Name) `[ 1 .. 100 ] characters`  
Name of the student

email required	string <email> (Email) [ 1 .. 254 ] characters Mail id of the student
password required	string (Password) [ 1 .. 100 ] characters

✓ 201 Student added successfully

RESPONSE SCHEMA: application/json

id	integer (ID)
name required	string (Name) [ 1 .. 100 ] characters Name of the student
email required	string <email> (Email) [ 1 .. 254 ] characters Mail id of the student
password required	string (Password) [ 1 .. 100 ] characters

— 400 Bad request

GET /crud/

## Response samples

200

201

Content type  
application/json

Copy Expand all Collapse all

```
[
  - {
    "id": 0,
    "name": "string",
    "email": "user@example.com",
    "password": "string"
  }
]
```

# crud\_create




AUTHORIZATIONS: 

Basic

 HTTP: Basic

HTTP Authorization Scheme: `basic`





REQUEST BODY SCHEMA: `application/json`

 name required	string (Name) <code>[ 1 .. 100 ] characters</code> Name of the student
 email required	string <email> (Email) <code>[ 1 .. 254 ] characters</code> Mail id of the student
 password required	string (Password) <code>[ 1 .. 100 ] characters</code>

## Responses

 201

RESPONSE SCHEMA: `application/json`

 id	integer (ID)
 name required	string (Name) <code>[ 1 .. 100 ] characters</code> Name of the student
 email required	string <email> (Email) <code>[ 1 .. 254 ] characters</code> Mail id of the student
 password required	string (Password) <code>[ 1 .. 100 ] characters</code>

POST /crud/



## Request samples

### Payload

Content type  
application/json

Copy

```
{
  "name": "string",
  "email": "user@example.com",
  "password": "string"
}
```

## Response samples

201

Content type  
application/json

Copy

```
{
  "id": 0,
  "name": "string",
  "email": "user@example.com",
  "password": "string"
}
```

## crud\_read

AUTHORIZATIONS:

Basic

HTTP: Basic

HTTP Authorization Scheme: `basic`

## PATH PARAMETERS

<code>id</code> <code>required</code>	integer A unique integer value identifying this student.
--	---

## Responses

✓ 200

RESPONSE SCHEMA: `application/json`

<code>id</code>	integer (ID)
<code>name</code> <code>required</code>	string (Name) <code>[ 1 .. 100 ] characters</code> Name of the student
<code>email</code> <code>required</code>	string <email> (Email) <code>[ 1 .. 254 ] characters</code> Mail id of the student
<code>password</code> <code>required</code>	string (Password) <code>[ 1 .. 100 ] characters</code>

GET `/crud/{id}/`

## Response samples

200

Content type  
`application/json`

Copy

```
{
  "id": 0,
  "name": "string",
  "email": "user@example.com",
  "password": "string"
```

```
}
```

## crud\_update


AUTHORIZATIONS: 

*Basic*


 HTTP: Basic


HTTP Authorization Scheme: `basic`


### PATH PARAMETERS

 `id` integer  
`required` A unique integer value identifying this student.

REQUEST BODY SCHEMA: `application/json`

 `name` string (Name) `[ 1 .. 100 ] characters`  
`required` Name of the student


 `email` string <email> (Email) `[ 1 .. 254 ] characters`  
`required` Mail id of the student


 `password` string (Password) `[ 1 .. 100 ] characters`  
`required`

## Responses

 200

RESPONSE SCHEMA: `application/json`

 `id` integer (ID)

 `name` string (Name) `[ 1 .. 100 ] characters`  
`required` Name of the student

email required	string <email> (Email) [ 1 .. 254 ] characters Mail id of the student
password required	string (Password) [ 1 .. 100 ] characters

PUT /crud/{id}/

## Request samples

### Payload

Content type  
application/json

Copy

```
{
  "name": "string",
  "email": "user@example.com",
  "password": "string"
}
```

## Response samples

200

Content type  
application/json

Copy

```
{
  "id": 0,
  "name": "string",
  "email": "user@example.com",
  "password": "string"
}
```

crud\_partial\_update


## AUTHORIZATIONS:

### Basic


#### HTTP: Basic


HTTP Authorization Scheme: `basic`


## PATH PARAMETERS

 `id` integer  
`required` A unique integer value identifying this student.

REQUEST BODY SCHEMA: `application/json`

 `name` string (Name) `[ 1 .. 100 ] characters`  
`required` Name of the student


 `email` string <email> (Email) `[ 1 .. 254 ] characters`  
`required` Mail id of the student


 `password` string (Password) `[ 1 .. 100 ] characters`  
`required`


## Responses


 **200**

RESPONSE SCHEMA: `application/json`

 `id` integer (ID)

 `name` string (Name) `[ 1 .. 100 ] characters`  
`required` Name of the student

 `email` string <email> (Email) `[ 1 .. 254 ] characters`  
`required` Mail id of the student

 `password` string (Password) `[ 1 .. 100 ] characters`  
`required`



PATCH /crud/{id}/



## Request samples

### Payload

Content type  
application/json

Copy

```
{
  "name": "string",
  "email": "user@example.com",
  "password": "string"
}
```

## Response samples

200

Content type  
application/json

Copy

```
{
  "id": 0,
  "name": "string",
  "email": "user@example.com",
  "password": "string"
}
```

## crud\_delete

AUTHORIZATIONS: ▼

Basic

🔒 HTTP: Basic

HTTP Authorization Scheme: `basic`

## PATH PARAMETERS

<code>id</code> <code>required</code>	<code>integer</code> A unique integer value identifying this student.
--	--

## Responses

— 204

**DELETE** `/crud/{id}/`

`http://127.0.0.1:8000/api/crud/{id}/`