

Project 4: - Flask Web Application

1) Home Page: -

Flask Application

Log InContact Us

What is flask?

Flask is a web framework. This means flask provides you with tools, libraries and technologies that allow you to build a web application. This web application can be some web pages, a blog, a wiki or go as big as a web-based calendar application or a commercial website.

Flask is part of the categories of the micro-framework. Micro-framework are normally framework with little to no dependencies to external libraries. This has pros and cons. Pros would be that the framework is light, there are little dependency to update and watch for security bugs, cons is that some time you will have to do more work by yourself or increase yourself the list of dependencies by adding plugins. In the case of Flask, its dependencies are:

- Werkzeug a WSGI utility library
- Jinja2 which is its template engine

Need to see the Player Table, Click on "Log In" button.

2) Contact Us Page: -

Flask Application

Log InContact Us

CONTACT US

Flask Application

yp32@njit.edu

3) Login Page: -

Flask Application

Log InContact Us

Sign In

Type same as given placeholder in input

Email

Password

Submit

4) Validation on Login Page: -

Sign In

Type same as given placeholder in input

Email

Not a valid email address.

Password

Select a stronger password.

Submit

5) Correction on Login Page: -

Sign In

Type same as given placeholder in input

Email

Password

Submit

6) Index Page: -

Flask Application									Log In Contact Us	
Add New Player										
ID	Name	Team	Position	Height	Weight	Age	Edit	Delete		
1	Adam Donachie	BAL	Catcher	74	180	23	edit	Delete		
2	Paul Bako	BAL	Catcher	74	215	35	edit	Delete		
3	Ramon Hernandez	BAL	Catcher	72	210	31	edit	Delete		
4	Kevin Millar	BAL	First Baseman	72	210	35	edit	Delete		
5	Chris Gomez	BAL	First Baseman	73	188	36	edit	Delete		
6	Brian Roberts	BAL	Second Baseman	69	176	29	edit	Delete		
7	Miguel Tejada	BAL	Shortstop	69	209	31	edit	Delete		
8	Melvin Mora	BAL	Third Baseman	71	200	35	edit	Delete		
9	Aubrey Huff	BAL	Third Baseman	76	231	30	edit	Delete		

7) View Page: -

Flask Application		Log In Contact Us			
Name:					
<input type="text" value="Adam Donachie"/>					
Team:					
<input type="text" value="BAL"/>					
Position:					
<input type="text" value="Catcher"/>					
Height:					
<input type="text" value="74"/>					
Weight:					
<input type="text" value="180"/>					
Age:					
<input type="text" value="23"/>					
Back					

8) New Page: -

Flask Application

[Log In](#) [Contact Us](#)

Name

Team

Position

Height

Weight

Age

Success

9) Edit Page: -

Flask Application

[Log In](#) [Contact Us](#)

Name

Team

Position

Height

Weight

Age

Save

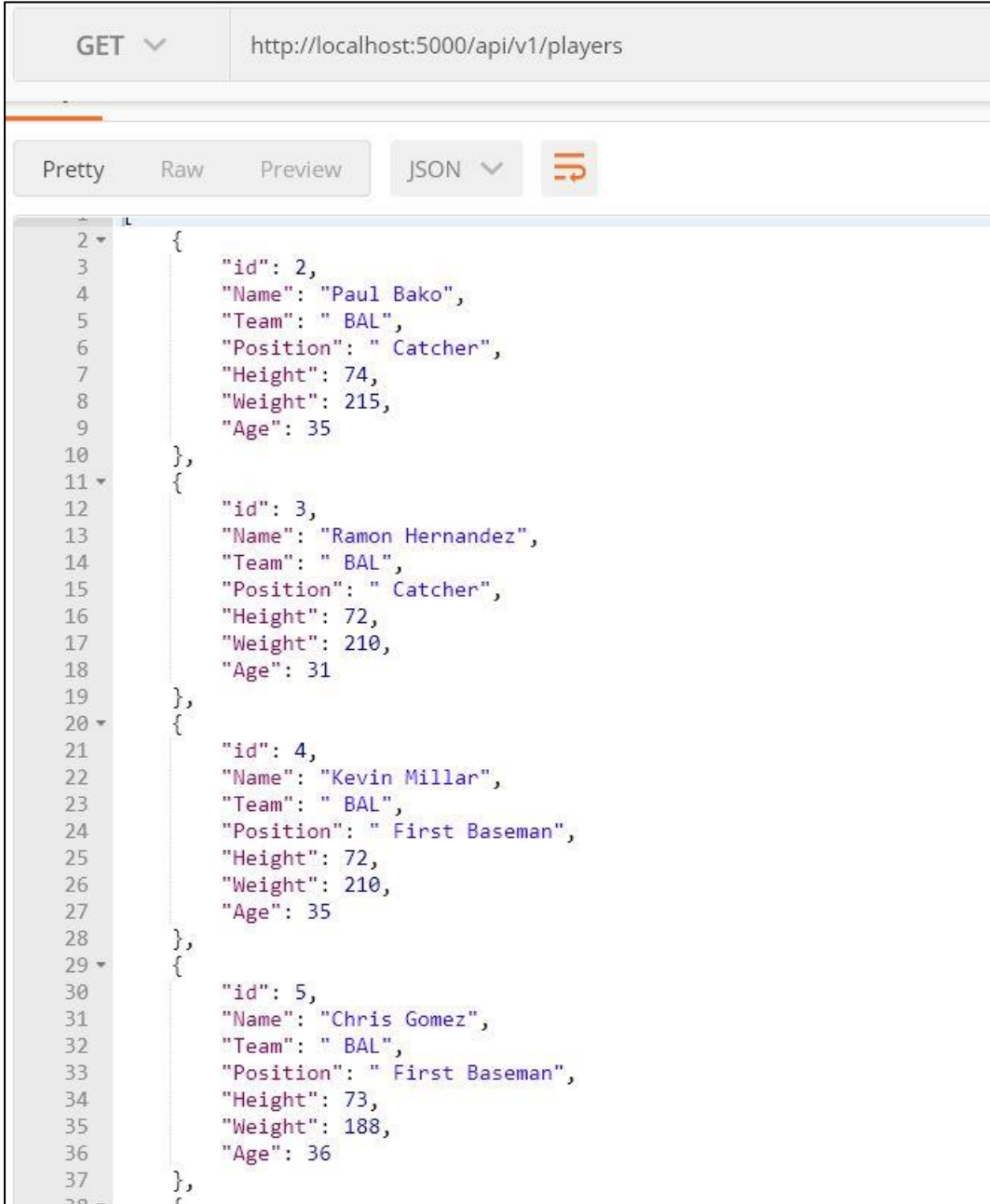
10) Delete Page: -

Flask Application								
Log In Contact Us								
Add New Player								
ID	Name	Team	Position	Height	Weight	Age	Edit	Delete
2	Paul Bako	BAL	Catcher	74	215	35	edit	Delete
3	Ramon Hernandez	BAL	Catcher	72	210	31	edit	Delete
4	Kevin Millar	BAL	First Baseman	72	210	35	edit	Delete
5	Chris Gomez	BAL	First Baseman	73	188	36	edit	Delete
6	Brian Roberts	BAL	Second Baseman	69	176	29	edit	Delete
7	Miguel Tejada	BAL	Shortstop	69	209	31	edit	Delete
8	Melvin Mora	BAL	Third Baseman	71	200	35	edit	Delete
9	Aubrey Huff	BAL	Third Baseman	76	231	30	edit	Delete
10	A.J. Pierzynski	CWS	Catcher	75	245	30	edit	Delete

11) PyCharm Table: -

WHERE		ORDER BY					
	id	Name	Team	Position	Height	Weight	Age
1	1	Adam Donachie	BAL	Catcher	74	180	22
2	2	Paul Bako	BAL	Catcher	74	215	35
3	3	Ramon Hernandez	BAL	Catcher	72	210	31
4	4	Kevin Millar	BAL	First Baseman	72	210	35
5	5	Chris Gomez	BAL	First Baseman	73	188	36
6	6	Brian Roberts	BAL	Second Baseman	69	176	29
7	7	Miguel Tejada	BAL	Shortstop	69	209	31
8	8	Melvin Mora	BAL	Third Baseman	71	200	35
9	9	Aubrey Huff	BAL	Third Baseman	76	231	30
10	10	A.J. Pierzynski	CWS	Catcher	75	245	30
11	11	Toby Hall	CWS	Catcher	75	240	31
12	12	Paul Konerko	CWS	First Baseman	74	215	31
13	13	Tadahito Iguchi	CWS	Second Baseman	69	185	32
14	14	Juan Uribe	CWS	Shortstop	71	175	28
15	15	Alex Cintron	CWS	Shortstop	74	199	28
16	16	Joe Crede	CWS	Third Baseman	73	200	29
17	17	Josh Fields	CWS	Third Baseman	73	215	24
18	18	Ryan Sweeney	CWS	Outfielder	76	200	22
19	19	Brian N. Anderson	CWS	Outfielder	74	205	25
20	20	Jose Molina	ANA	Catcher	74	220	32
21	21	Jeff Mathis	ANA	Catcher	72	180	24
22	22	Mike Napoli	ANA	Catcher	72	205	25
23	23	Casey Kotchman	ANA	First Baseman	75	210	24
24	24	Kendry Morales	ANA	First Baseman	73	220	24
25	25	Shea Hillenbrand	ANA	First Baseman	73	211	32
26	26	Robb Quinlan	ANA	First Baseman	73	200	30

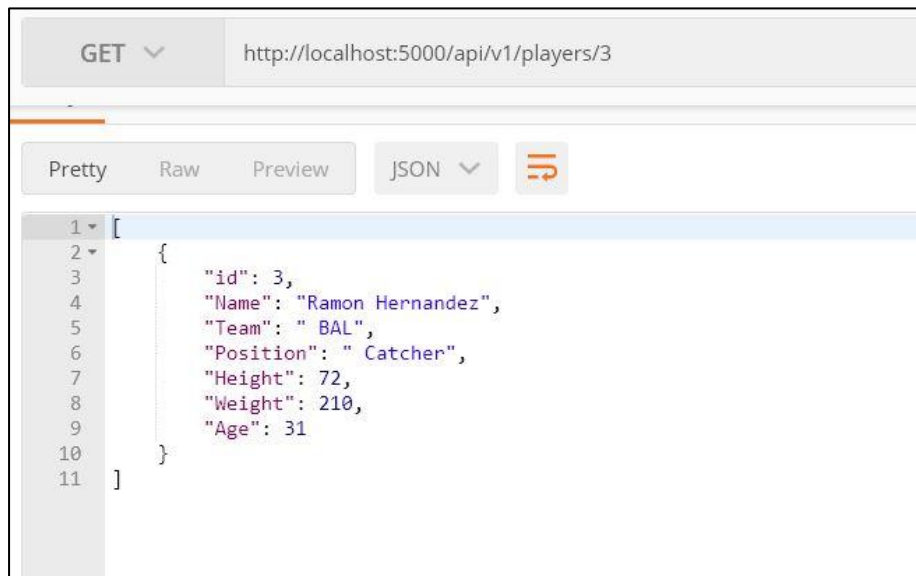
12) View All Data: -



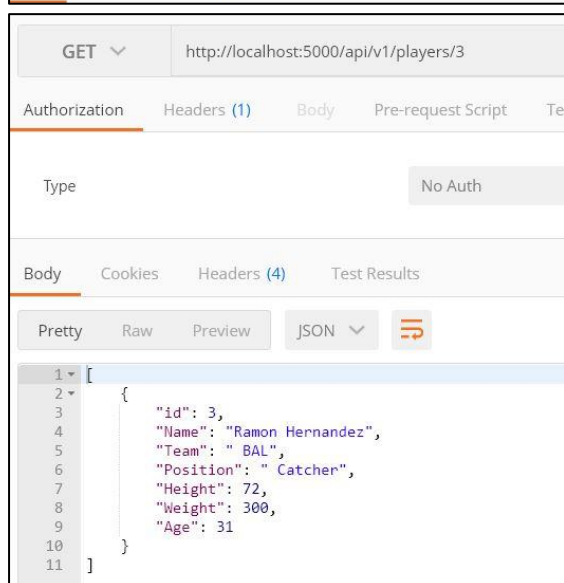
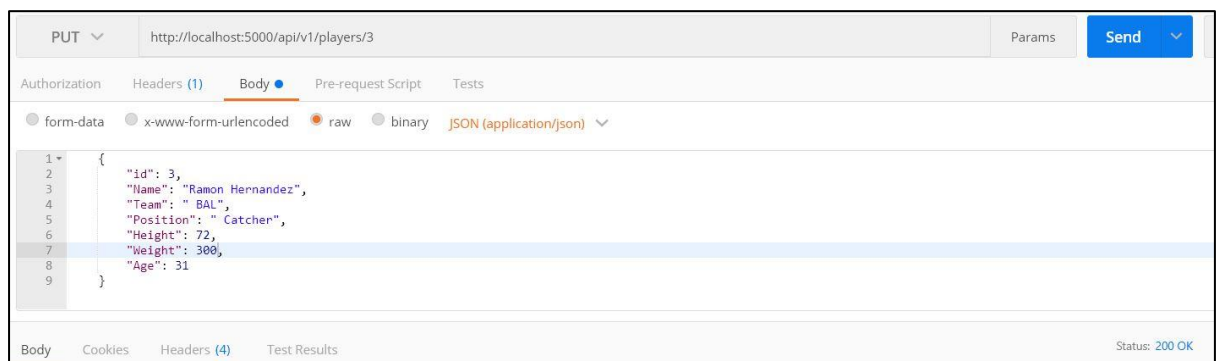
The screenshot shows a REST client interface with a GET request to `http://localhost:5000/api/v1/players`. The response is displayed in JSON format, showing a list of four players. The interface includes tabs for 'Pretty', 'Raw', and 'Preview', and a dropdown menu set to 'JSON'. A line number indicator is visible on the left side of the code editor.

```
2 {
3   "id": 2,
4   "Name": "Paul Bako",
5   "Team": "BAL",
6   "Position": "Catcher",
7   "Height": 74,
8   "Weight": 215,
9   "Age": 35
10 },
11 {
12   "id": 3,
13   "Name": "Ramon Hernandez",
14   "Team": "BAL",
15   "Position": "Catcher",
16   "Height": 72,
17   "Weight": 210,
18   "Age": 31
19 },
20 {
21   "id": 4,
22   "Name": "Kevin Millar",
23   "Team": "BAL",
24   "Position": "First Baseman",
25   "Height": 72,
26   "Weight": 210,
27   "Age": 35
28 },
29 {
30   "id": 5,
31   "Name": "Chris Gomez",
32   "Team": "BAL",
33   "Position": "First Baseman",
34   "Height": 73,
35   "Weight": 188,
36   "Age": 36
37 },
38 }
```

13) View Single Data: -



14) Edit Data: -



15) New Data Created: -

PlayersTable

POST ▼ http://localhost:5000/api/v1/players Params Send ▼

Authorization Headers (1) **Body** Pre-request Script Tests

☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary JSON (application/json) ▼

```
1 {
2   "Name": "Ramon Sphere",
3   "Team": " CAL",
4   "Position": " Front",
5   "Height": 78,
6   "Weight": 250,
7   "Age": 30
8 }
```

Body Cookies Headers (4) Test Results Status: 201 CREATED

GET ▼ http://localhost:5000/api/v1/players

Pretty Raw Preview JSON ▼ ↺

```
857 {
858   {
859     "id": 97,
860     "Name": "B.J. Upton",
861     "Team": " TB",
862     "Position": " Third Baseman",
863     "Height": 75,
864     "Weight": 180,
865     "Age": 23
866   },
867   {
868     "id": 98,
869     "Name": "Carl Crawford",
870     "Team": " TB",
871     "Position": " Outfielder",
872     "Height": 74,
873     "Weight": 219,
874     "Age": 26
875   },
876   {
877     "id": 100,
878     "Name": "Adam",
879     "Team": "BAL",
880     "Position": "Front",
881     "Height": 56,
882     "Weight": 102,
883     "Age": 25
884   },
885   {
886     "id": 101,
887     "Name": "Ramon Sphere",
888     "Team": " CAL",
889     "Position": " Front",
890     "Height": 78,
891     "Weight": 250,
892     "Age": 30
893   }
894 }
```


16) Delete Data: -

DELETE Params

Authorization Headers (1) Body **Pre-request Script** Tests

Type

Body Cookies Headers (4) Test Results Status: 200 OK

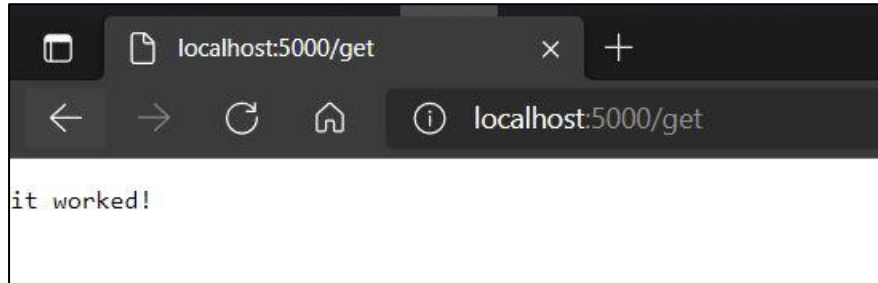
GET

Pretty Raw Preview

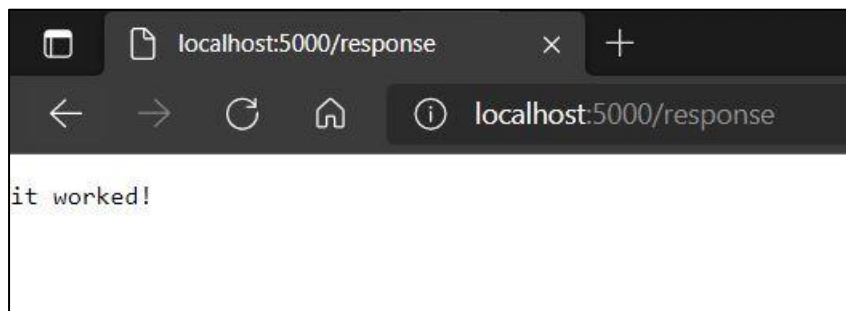
```
848 {
849   "id": 96,
850   "Name": "Ben Zobrist",
851   "Team": "TB",
852   "Position": "Shortstop",
853   "Height": 75,
854   "Weight": 200,
855   "Age": 26
856 },
857 {
858   "id": 97,
859   "Name": "B.J. Upton",
860   "Team": "TB",
861   "Position": "Third Baseman",
862   "Height": 75,
863   "Weight": 180,
864   "Age": 23
865 },
866 {
867   "id": 98,
868   "Name": "Carl Crawford",
869   "Team": "TB",
870   "Position": "Outfielder",
871   "Height": 74,
872   "Weight": 219,
873   "Age": 26
874 },
875 {
876   "id": 100,
877   "Name": "Adam",
878   "Team": "BAL",
879   "Position": "Front",
880   "Height": 56,
881   "Weight": 102,
882   "Age": 25
883 }
884 ]
```

- **Example.py File Screenshots**

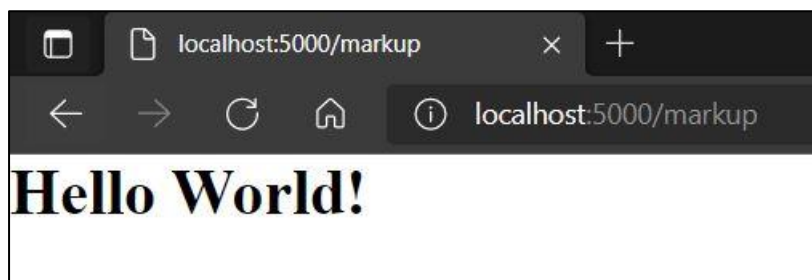
1) Get API



2) Response API



3) Markup API



4) Logic API

