

Test: **GET: /api/<resource>[query_expression],[&field_expression]**

<http://127.0.0.1:5000/api/people?birthCountry=USA&birthYear=1981&fields=playerID,birthDay,nameFirst,nameLast>

GET <http://127.0.0.1:5000/api/people?birthCountry=USA&birthYear=1981&fields=playerID,birthDay,nameFirst,nameLast> Params Send Save

Pretty Raw Preview JSON

```
25  {
26    "playerID": "avilem101"
27  },
28  {
29    "nameLast": "Baker",
30    "birthDay": "20",
31    "nameFirst": "John",
32    "playerID": "bakerj001"
33  },
34  {
35    "nameLast": "Baker",
36    "birthDay": "19",
37    "nameFirst": "Scott",
38    "playerID": "bakerc02"
39  },
40  {
41    "nameLast": "Baldelli",
42    "birthDay": "25",
43    "nameFirst": "Rocco",
44    "playerID": "baldero01"
45  },
46  {
47    "nameLast": "Bannister",
48    "birthDay": "28",
49    "nameFirst": "Brian",
50    "playerID": "bannibr01"
51  },
52  {
53    "nameLast": "Barden",
54    "birthDay": "2",
55    "nameFirst": "Brian",
56    "playerID": "bardebr01"
57  },
58  {
59    "nameLast": "Bergmann",
60    "birthDay": "25",
61    "nameFirst": "Jason",
62    "playerID": "bergmja01"
63  },
64  },
65  "links": [
66    {
67      "current": "/api/people?birthYear=1981&birthCountry=USA&fields=playerID,birthDay,nameFirst,nameLast&offset=0&limit=10"
68    },
69    {
70      "next": "/api/people?birthYear=1981&birthCountry=USA&fields=playerID,birthDay,nameFirst,nameLast&offset=10&limit=10"
71    }
72  ]
}
```

<http://127.0.0.1:5000/api/batting?yearID=2004&teamID=SFN&fields=playerID,yearID,teamID,H,AB>

GET <http://127.0.0.1:5000/api/batting?yearID=2004&teamID=SFN&fields=playerID,yearID,teamID,H,AB> Params Send Save

Pretty Raw Preview JSON

```
35  {
36    "playerID": "burbada01",
37    "yearID": "2004"
38  },
39  {
40    "H": "0",
41    "AB": "0",
42    "teamID": "SFN",
43    "playerID": "chrisja01",
44    "yearID": "2004"
45  },
46  {
47    "H": "0",
48    "AB": "3",
49    "teamID": "SFN",
50    "playerID": "coopebr01",
51    "yearID": "2004"
52  },
53  {
54    "H": "1",
55    "AB": "3",
56    "teamID": "SFN",
57    "playerID": "correke01",
58    "yearID": "2004"
59  },
60  {
61    "H": "116",
62    "AB": "397",
63    "teamID": "SFN",
64    "playerID": "cruzde01",
65    "yearID": "2004"
66  },
67  {
68    "H": "12",
69    "AB": "43",
70    "teamID": "SFN",
71    "playerID": "dallibr01",
72    "yearID": "2004"
73  },
74  },
75  "links": [
76    {
77      "current": "/api/batting?teamID=SFN&yearID=2004&fields=playerID,yearID,teamID,H,AB&offset=0&limit=10"
78    },
79    {
80      "next": "/api/batting?teamID=SFN&yearID=2004&fields=playerID,yearID,teamID,H,AB&offset=10&limit=10"
81    }
82  ]
}
```

Test: **POST: /api/<resource>**

<http://127.0.0.1:5000/api/people>

The screenshot shows a REST client interface with a POST request to `http://127.0.0.1:5000/api/people`. The request body is a JSON object: `{ "playerID": "yuliu123123", "birthDay": "13", "birthYear": "1995" }`. The interface includes tabs for Authorization, Headers (1), Body, Pre-request Script, and Tests. The Body tab is selected, and the content is displayed in a JSON (application/json) format. Below the request, there are tabs for Body, Cookies, Headers (4), and Test Results. The Body tab is selected, and the response is displayed in a Pretty format. The status is 200 OK and the time is 83 ms.

```
1 {
2   "playerID": "yuliu123123",
3   "birthDay": "13",
4   "birthYear": "1995"
5 }
```

Method POST on resource people implemented!

Get the new row to test if the row is inserted into the table

<http://127.0.0.1:5000/api/people?playerID=yuliu123123>

The screenshot shows a REST client interface with a GET request to `http://127.0.0.1:5000/api/people?playerID=yuliu123123`. The response body is a JSON object: `{ "data": { "birthCity": null, "deathCountry": null, "birthYear": "1995", "weight": null, "deathMonth": null, "bbrefID": null, "nameFirst": null, "birthCountry": null, "nameGiven": null, "throws": null, "playerID": "yuliu123123", "bats": null, "nameLast": null, "birthState": null, "deathCity": null, "finalGame": null, "birthDay": "13", "deathYear": null, "retroID": null, "debut": null, "birthMonth": null, "height": null, "deathState": null, "deathDay": null }, "links": { "current": "/api/people?playerID=yuliu123123&offset=0&limit=10", "next": "/api/people?playerID=yuliu123123&offset=10&limit=10" } }`. The interface includes tabs for Body, Cookies, Headers (4), and Test Results. The Body tab is selected, and the content is displayed in a JSON format. Below the response, there are tabs for Pretty, Raw, Preview, and JSON. The JSON tab is selected, and the response is displayed in a JSON format. The status is 200 OK and the time is 31 ms.

```
1 {
2   "data": {
3     "birthCity": null,
4     "deathCountry": null,
5     "birthYear": "1995",
6     "weight": null,
7     "deathMonth": null,
8     "bbrefID": null,
9     "nameFirst": null,
10    "birthCountry": null,
11    "nameGiven": null,
12    "throws": null,
13    "playerID": "yuliu123123",
14    "bats": null,
15    "nameLast": null,
16    "birthState": null,
17    "deathCity": null,
18    "finalGame": null,
19    "birthDay": "13",
20    "deathYear": null,
21    "retroID": null,
22    "debut": null,
23    "birthMonth": null,
24    "height": null,
25    "deathState": null,
26    "deathDay": null
27  },
28  "links": {
29    "current": "/api/people?playerID=yuliu123123&offset=0&limit=10"
30  },
31  "next": "/api/people?playerID=yuliu123123&offset=10&limit=10"
32 }
```

Test: **GET: /api/<resource>/<primary_key>[?fields_expression]**

http://127.0.0.1:5000/api/batting/aguilje01_2016_1_CLE?fields=playerID,H,G,AB

GET http://127.0.0.1:5000/api/batting/aguilje01_2016_1_CLE?fields=playerID,H,G,AB Params Send Save

Body Cookies Headers (4) Test Results Status: 200 OK Time: 88 ms

Pretty Raw Preview JSON

```
1 {
2   "data": [
3     {
4       "AB": "6",
5       "G": "g",
6       "H": "g",
7       "playerID": "aguilje01"
8     }
9   ],
10  "links": [
11    {
12      "current": "/api/batting/aguilje01_2016_1_CLE?fields=playerID,H,G,AB&offset=0&limit=10"
13    },
14    {
15      "next": "/api/batting/aguilje01_2016_1_CLE?fields=playerID,H,G,AB&offset=10&limit=10"
16    }
17  ]
18 }
```

Test: **PUT: /api/<resource>/<primary_key>**

First get the original row.

GET <http://127.0.0.1:5000/api/people/yuliu123123> Params Send Save

Pretty Raw Preview JSON

```
1 {
2   "data": [
3     {
4       "birthCity": null,
5       "deathCountry": null,
6       "birthYear": "1995",
7       "weight": null,
8       "deathMonth": null,
9       "bbrefID": null,
10      "nameFirst": null,
11      "birthCountry": null,
12      "nameGiven": null,
13      "throws": null,
14      "playerID": "yuliu123123",
15      "bats": null,
16      "nameLast": null,
17      "birthState": null,
18      "deathCity": null,
19      "finalGame": null,
20      "birthDay": "13",
21      "deathYear": null,
22      "retroID": null,
23      "debut": null,
24      "birthMonth": null,
25      "height": null,
26      "deathState": null,
27      "deathDay": null
28    }
29  ],
30  "links": [
31    {
32      "current": "/api/people/yuliu123123?offset=0&limit=10"
33    },
34    {
35      "next": "/api/people/yuliu123123?offset=10&limit=10"
36    }
37  ]
38 }
```

Then use PUT to update the data.

<http://127.0.0.1:5000/api/people/yuliu123123>

The screenshot shows a REST client interface with a PUT request to `http://127.0.0.1:5000/api/people/yuliu123123`. The request body is a JSON object: `{ "birthDay": "30", "birthYear": "1996" }`. The status is 200 OK, and the response message is "Method PUT on resource people implemented!".

```
PUT http://127.0.0.1:5000/api/people/yuliu123123
```

```
{
  "birthDay": "30",
  "birthYear": "1996"
}
```

Status: 200 OK Time: 33 ms

Method PUT on resource people implemented!

The result shows that the data is updated.

The screenshot shows a REST client interface with a GET request to `http://127.0.0.1:5000/api/people/yuliu123123`. The response is a JSON object containing a list of person data and links. The status is 200 OK, and the response time is 33 ms.

```
GET http://127.0.0.1:5000/api/people/yuliu123123
```

```
{
  "data": [
    {
      "birthCity": null,
      "deathCountry": null,
      "birthYear": "1996",
      "weight": null,
      "deathMonth": null,
      "bbrefID": null,
      "nameFirst": null,
      "birthCountry": null,
      "nameGiven": null,
      "throws": null,
      "playerID": "yuliu123123",
      "bats": null,
      "nameLast": null,
      "birthState": null,
      "deathCity": null,
      "finalGame": null,
      "birthDay": "30",
      "deathYear": null,
      "retroID": null,
      "debut": null,
      "birthMonth": null,
      "height": null,
      "deathState": null,
      "deathDay": null
    }
  ],
  "links": [
    {
      "current": "/api/people/yuliu123123?offset=0&limit=10"
    },
    {
      "next": "/api/people/yuliu123123?offset=10&limit=10"
    }
  ]
}
```

Status: 200 OK Time: 33 ms

Test: **DELETE: /api/<resource>/<primary_key>**

Firstly show the original data.

GET http://127.0.0.1:5000/api/people/yuliu123123

Type: No Auth

Status: 200 OK Time: 42 ms

Body: Pretty Raw Preview JSON

```
1 {
2   "data": [
3     {
4       "birthCity": null,
5       "deathCountry": null,
6       "birthYear": "1996",
7       "weight": null,
8       "deathMonth": null,
9       "bbrefID": null,
10      "nameFirst": null,
11      "birthCountry": null,
12      "nameGiven": null,
13      "throws": null,
14      "playerID": "yuliu123123",
15      "bats": null,
16      "nameLast": null,
17      "birthState": null,
18      "deathCity": null,
19      "finalGame": null,
20      "birthDay": "30",
21      "deathYear": null,
22      "retroID": null,
23      "debut": null,
24      "birthMonth": null,
25      "height": null,
26      "deathState": null,
27      "deathDay": null
28    }
29  ],
30  "links": [
31    {
32      "current": "/api/people/yuliu123123?offset=0&limit=10"
33    },
34    {
35      "next": "/api/people/yuliu123123?offset=10&limit=10"
36    }
37  ]
38 }
```

Then use DELETE to delete the data.

<http://127.0.0.1:5000/api/people/yuliu123123>

DELETE http://127.0.0.1:5000/api/people/yuliu123123

Type: No Auth

Status: 200 OK Time: 34 ms

Body: Pretty Raw Preview

Method DELETE on resource people implemented!

Query the data, we find that the data is deleted.

GET http://127.0.0.1:5000/api/people/yuliu123123

Type: No Auth

Status: 200 OK Time: 31 ms

Body: Pretty Raw Preview JSON

```
1 {
2   "data": [],
3   "links": [
4     {
5       "current": "/api/people/yuliu123123?offset=0&limit=10"
6     },
7     {
8       "next": "/api/people/yuliu123123?offset=10&limit=10"
9     }
10  ]
11 }
```

Test: GET: /api/<resource>/<primary_key>/<related_resource>[query_expression][fields_expression]

<http://127.0.0.1:5000/api/people/willite01/batting?teamID=BOS>

GET <http://127.0.0.1:5000/api/people/willite01/batting?teamID=BOS> Params Send Save

Pretty Raw Preview JSON

```
285 {
286   "teamID": "BOS",
287   "playerID": "willite01",
288   "SID": "12",
289   "SB": "24",
290   "SO": "21",
291   "HR": "28",
292   "yearID": "1958",
293   "R": "82",
294   "H": "186",
295   "SH": "6",
296   "HBP": "0",
297   "BB": "1",
298 },
299 {
300   "BB": "144",
301   "CS": "1",
302   "AB": "531",
303   "G": "148",
304   "RBI": "126",
305   "lgID": "AL",
306   "stint": "1",
307   "SF": "",
308   "IBB": "",
309   "teamID": "BOS",
310   "playerID": "willite01",
311   "GIDP": "18",
312   "SB": "18",
313   "SO": "45",
314   "HR": "38",
315   "yearID": "1951",
316   "R": "109",
317   "H": "169",
318   "SH": "9",
319   "HBP": "0",
320   "BB": "4",
321   "SB": "1"
322 }
323 },
324 "links": [
325   {
326     "current": "/api/people/willite01/batting?teamID=BOS&offset=0&limit=10"
327   },
328   {
329     "next": "/api/people/willite01/batting?teamID=BOS&offset=10&limit=10"
330   }
331 ]
332 }
```

Test: POST: /api/<resource>/<primary_key>/<related_resource>

<http://127.0.0.1:5000/api/people/willite01/batting>

POST <http://127.0.0.1:5000/api/people/willite01/batting> Params Send Save

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

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

```
1 {
2   "teamID": "LLL",
3   "yearID": "2000",
4   "stint": "1"
5 }
```

Body Cookies Headers (4) Test Results Status: 200 OK Time: 276 ms

Pretty Raw Preview

Method POST on resource people implemented!

GET <http://127.0.0.1:5000/api/batting?teamID=LLL> Params Send Save

Type No Auth

Body Cookies Headers (4) Test Results Status: 200 OK Time: 149 ms

Pretty Raw Preview JSON

```
1 {
2   "data": [
3     {
4       "BB": null,
5       "IBB": null,
6       "AB": null,
7       "G": null,
8       "RBI": null,
9       "lgID": null,
10      "stint": "1",
11      "SF": null,
12      "CS": null,
13      "teamID": "LLL",
14      "playerID": "willite01",
15      "GIDP": null,
16      "SB": null,
17      "SO": null,
18      "HR": null,
19      "yearID": "2000",
20      "R": null,
21      "H": null,
22      "SH": null,
23      "HBP": null,
24      "BB": null,
25      "SB": null
26    }
27   ],
28   "links": [
29     {
30       "current": "/api/batting?teamID=LLL&offset=0&limit=10"
31     },
32     {
33       "next": "/api/batting?teamID=LLL&offset=10&limit=10"
34     }
35   ]
36 }
```

Test: **GET: /api/teammates/<playerid>**
<http://127.0.0.1:5000/api/teammates/willite01>

GET Params

Type

Body Cookies Headers (4) Test Results Status: 200 OK Time: 678 ms

Pretty Raw Preview JSON

```
1 [
2   {
3     "nameLast": "Agganis",
4     "team_year": "BOS_1954,BOS_1955",
5     "nameFirst": "Harry",
6     "playerID": "agganha01"
7   },
8   {
9     "nameLast": "Andres",
10    "team_year": "BOS_1946",
11    "nameFirst": "Ernie",
12    "playerID": "andreer01"
13  },
14  {
15    "nameLast": "Aspromonte",
16    "team_year": "BOS_1957,BOS_1958",
17    "nameFirst": "Ken",
18    "playerID": "asproke01"
19  },
20  {
21    "nameLast": "Atkins",
22    "team_year": "BOS_1950,BOS_1952",
23    "nameFirst": "James",
24    "playerID": "atkinja01"
25  },
26  {
27    "nameLast": "Auker",
28    "team_year": "BOS_1939",
29    "nameFirst": "Elden",
30    "playerID": "aukerel01"
31  },
32  {
33    "nameLast": "Aulds",
34    "team_year": "BOS_1947",
35    "nameFirst": "Tex",
36    "playerID": "auldsle01"
37  },
38 ]
```

Test: **GET: /api/people/<playerid>/career_stats**
http://127.0.0.1:5000/api/people/aardsda01/career_stats

GET Params

Pretty Raw Preview JSON

```
55   "hits": "0",
56   "yearID": "2013",
57   "g_all": "43",
58   "ABS": "0",
59   "teamID": "NHN",
60   "playerID": "aardsda01",
61   "errors": 0
62 },
63 {
64   "Assits": 5,
65   "hits": "0",
66   "yearID": "2009",
67   "g_all": "73",
68   "ABS": "0",
69   "teamID": "SEA",
70   "playerID": "aardsda01",
71   "errors": 0
72 },
73 {
74   "Assits": 3,
75   "hits": "0",
76   "yearID": "2010",
77   "g_all": "53",
78   "ABS": "0",
79   "teamID": "SEA",
80   "playerID": "aardsda01",
81   "errors": 1
82 },
83 {
84   "Assits": 0,
85   "hits": "0",
86   "yearID": "2004",
87   "g_all": "11",
88   "ABS": "0",
89   "teamID": "SFN",
90   "playerID": "aardsda01",
91   "errors": 0
92 },
93 ],
94 "links": {
95   {
96     "current": "/api/people/aardsda01/career_stats?offset=0&limit=10"
97   },
98   {
99     "next": "/api/people/aardsda01/career_stats?offset=10&limit=10"
100  }
101 }
```

Test: GET: /api/roster?teamid=<teamid>&yearid=<yearid>

<http://127.0.0.1:5000/api/roster?teamID=SFN&yearID=2004>

GET [http://127.0.0.1:5000/api/roster?teamID=SFN&yearID=2004](#) Params Send Save

Pretty Raw Preview JSON

```
85      "attempts": 1
86    },
87    {
88      "nameLast": "Correia",
89      "hits": "1",
90      "errors": 0,
91      "yearID": "2004",
92      "abs": "3",
93      "g_all": "12",
94      "nameFirst": "Kevin",
95      "teamID": "SFN",
96      "playerID": "correke01",
97      "attempts": 1
98    },
99    {
100     "nameLast": "Cruz",
101     "hits": "116",
102     "errors": 8,
103     "yearID": "2004",
104     "abs": "397",
105     "g_all": "127",
106     "nameFirst": "Deivi",
107     "teamID": "SFN",
108     "playerID": "cruzde01",
109     "attempts": 266
110   },
111   {
112     "nameLast": "Dallimore",
113     "hits": "12",
114     "errors": 2,
115     "yearID": "2004",
116     "abs": "43",
117     "g_all": "20",
118     "nameFirst": "Brian",
119     "teamID": "SFN",
120     "playerID": "dallibr01",
121     "attempts": 26
122   }
123 ],
124 "links": {
125   {
126     "current": "/api/roster?teamID=SFN&yearID=2004&offset=0&limit=10"
127   },
128   {
129     "next": "/api/roster?teamID=SFN&yearID=2004&offset=10&limit=10"
130   }
131 }
```