

The screenshot shows the Postman application interface. At the top, there are navigation links: Home, Workspaces, API Network, and Explore. A search bar is located at the top right. On the left sidebar, there are sections for Collections, Environments, and History. The main area displays a POST request to <http://127.0.0.1:8000/visualai>. The request parameters are set to "query": "claims by provider status" and "filter": {"claim\_completion\_month\_adj": "'July', 'May'"}.

**Body (8)**

```
1
2   ...
3     "query": "claims by provider status",
4     "filter": {
5       ...
6         "claim_completion_month_adj": "'July', 'May'"
7     }
8
9
10
```

**Headers (4)**

**Test Results**

Status: 200 OK Time: 8.09 s Size: 586 B Save as Example

**Pretty** **Raw** **Preview** **Visualize** **JSON**

```
1
2   "response_desc": "Provider_Status",
3   "response_data": {
4     "metrics": [
5       {
6         "xvalue": "",
7         "yvalue": 1,
8         "zvalue": null
9       },
10      {
11        "xvalue": "
```

The screenshot shows the Postman application interface. At the top, there are navigation links: Home, Workspaces, API Network, and Explore. A search bar is located at the top right. On the left sidebar, there are sections for Collections, Environments, and History. The main area displays a POST request to <http://127.0.0.1:8000/visualai>. The request parameters are set to "query": "claims by population volume" and "filter": {"claim\_completion\_month\_adj": "'July', 'May'"}.

**Body (8)**

```
1
2   ...
3     "query": "claims by population volume",
4     "filter": {
5       ...
6         "claim_completion_month_adj": "'July', 'May'"
7     }
8
9
10
```

**Headers (4)**

**Test Results**

Status: 200 OK Time: 3.87 s Size: 527 B Save as Example

**Pretty** **Raw** **Preview** **Visualize** **JSON**

```
1
2   "response_desc": "Population_Volume",
3   "response_data": {
4     "metrics": [
5       {
6         "xvalue": "HOME",
7         "yvalue": 101,
8         "zvalue": null
9       },
10      {
11        "xvalue": "
```

Home Workspaces API Network Explore Search Postman Invite Upgrade

Overview New Collection POST http://127.0.0.1:8000/ No Environment

http://127.0.0.1:8000/visualai

POST http://127.0.0.1:8000/visualai

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body

```
1 "query": "claims by funding id",
2 ...
3 "filter": {
4 ...
5   "claim_completion_month_adj": "July", "May"
6 ...
7 }
```

Pretty Raw Preview Visualize JSON

```
1 "response_desc": "Funding_ID",
2 "response_data": [
3   "metrics": [
4     {
5       "xvalue": "ASO/ITS",
6       "yvalue": 1,
7       "zvalue": null
8     },
9   ]
10 ]
```

Status: 200 OK Time: 6.68 s Size: 1.17 KB Save as Example

Find and replace Console Postbot Runner Start Proxy Cookies Trash

Home Workspaces API Network Explore Search Postman Invite Upgrade

Overview New Collection POST http://127.0.0.1:8000/ No Environment

http://127.0.0.1:8000/visualai

POST http://127.0.0.1:8000/visualai

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body

```
1 "query": "claims by lpp claims",
2 ...
3 "filter": {
4 ...
5   "claim_completion_month_adj": "July", "May"
6 ...
7 }
```

Pretty Raw Preview Visualize JSON

```
1 "response_desc": "LPP_Claims",
2 "response_data": [
3   "metrics": [
4     {
5       "xvalue": "LPP Dt > RCVD Dt",
6       "yvalue": 322,
7       "zvalue": null
8     },
9   ]
10 ]
```

Status: 200 OK Time: 10.20 s Size: 523 B Save as Example

Find and replace Console Postbot Runner Start Proxy Cookies Trash

The screenshot shows the Postman application interface. At the top, there are navigation links for Home, Workspaces, API Network, and Explore, along with a search bar and user authentication options. The main workspace displays a collection named "visualai" with a single POST request. The request URL is `http://127.0.0.1:8000/visualai`. The "Body" tab is selected, showing the following JSON payload:

```
1
2   "query": "claims by prompt pay",
3   "filter": {
4     "claim_completion_month_adj": "July", 'May'
5   }
```

The response status is 200 OK, with a response time of 4.04 s and a size of 454 B. The response body is displayed in Pretty format:

```
1
2   "response_desc": "Prompt_Pay",
3   "response_data": {
4     "metrics": [
5       {
6         "xvalue": "0",
7         "yvalue": 565,
8         "zvalue": null
9       },
10      {
11        "xvalue": "Yes",
12        "yvalue": 640,
```

This screenshot shows the same Postman interface as the first one, but with a different query parameter. The POST request URL is now `http://127.0.0.1:8000/visualai`. The "Body" tab is selected, showing the following JSON payload:

```
1
2   "query": "How am I doing by trend analysis claims volume",
3   "filter": {
4     "claim_completion_month_adj": "July", 'May'
5   }
```

The response status is 200 OK, with a response time of 3.59 s and a size of 1.26 KB. The response body is displayed in Pretty format:

```
1
2   "response_desc": "Trend_Analysis_Claims_Volumes",
3   "response_data": {
4     "metrics": [
5       {
6         "SUM(intrest amt adj)": 0.0,
7         "Prompt pay claims": 337.0,
8         "Claims LPP Error Amt": 2590358076242.0,
9         "Claims LPP Correct Amt": 138570690409.0,
10        "Number of Total Claims": 662,
11        "claim completion month adj": "July"
```

The screenshot shows the Postman application interface. At the top, there are navigation links for Home, Workspaces, API Network, and Explore, along with a search bar and an 'Upgrade' button. The main workspace displays a collection named 'Overview'. A POST request is being made to the URL <http://127.0.0.1:8000/>. The request body contains the following JSON:

```
1 "query": "How am i doing by trend analysis amounts",
2 "filter": {
3     "claim_completion_month_adj": "July", "May"
4 }
```

The response status is 200 OK, with a response time of 6.59 s and a size of 868 B. The response body is displayed in Pretty format:

```
1 "response_desc": "Trend_Analysis_Amounts",
2 "response_data": {
3     "metrics": [
4         {
5             "paid amt adj sum": 7585800.0,
6             "totl chrg amt adj sum": 392330933.0,
7             "intrest amt adj sum": 0.0,
8             "claim completion month adj": "July"
9         },
10        {
11            "paid amt adj sum": 10650250.0
12        }
13    ]
14 }
```

This screenshot shows another instance of the Postman interface. A POST request is being made to the URL <http://127.0.0.1:8000/>. The request body contains the following JSON:

```
1 "query": "Whats the status on LPP claims",
2 "filter": {
3     "claim_completion_month_adj": "July", "May"
4 }
```

The response status is 200 OK, with a response time of 3.45 s and a size of 656 B. The response body is displayed in Pretty format:

```
1 "response_desc": "LPP_Claims",
2 "response_data": {
3     "metrics": [
4         {
5             "xvalue": "LPP Dt > RCVD Dt",
6             "yvalue": 322,
7             "zvalue": null
8         },
9         {
10            "xvalue": "LPP Dt = RCVD Dt",
11            "yvalue": 100
12        }
13    ]
14 }
```

The screenshot shows the Postman application interface. At the top, there are navigation links: Home, Workspaces, API Network, Explore, and a search bar. On the right side, there are buttons for Invite, Log In, Upgrade, and a user profile icon.

In the main workspace, there is a collection named "Overview". A new collection button is visible. The URL is set to `POST http://127.0.0.1:8000/visualai`. The status bar indicates "No Environment".

The request details show a POST method to `http://127.0.0.1:8000/visualai`. The "Body" tab is selected, showing the following JSON payload:

```
1
2   "query": "claims by provider status",
3   "filter": [
4     {
5       "claim_completion_month_adj": "July", 'May'
6     }
7   ]
```

The response status is 200 OK, with a time of 5.62 s and a size of 589 B. The response body is displayed in Pretty, Raw, Preview, and Visualize formats. The Pretty format shows:

```
1
2   "response_desc": "Provider_Status",
3   "response_data": {
4     "metrics": [
5       {
6         "xvalue": "",
7         "yvalue": 1,
8         "zvalue": null
9       },
10      {
11        "xvalue": "OUT",
12        "yvalue": 247
13      }
14    ]
15  }
```

The screenshot shows the Postman application interface, similar to the first one but with a different workspace name.

The workspace is named "My Workspace". The URL is set to `POST http://127.0.0.1:8000/visualai`. The status bar indicates "No Environment".

The request details show a POST method to `http://127.0.0.1:8000/visualai`. The "Body" tab is selected, showing the following JSON payload:

```
1
2   "query": "claims by trend analysis by amounts",
3   "filter": [
4     {
5       "claim_completion_month_adj": "July", 'May'
6     }
7   ]
```

The response status is 200 OK, with a time of 7.08 s and a size of 857 B. The response body is displayed in Pretty, Raw, Preview, and Visualize formats. The Pretty format shows:

```
1
2   "response_desc": "Trend_Analysis_Amounts",
3   "response_data": {
4     "metrics": [
5       {
6         "paid amt adj sum": 7585800.0,
7         "totl chrg amt adj sum": 3923309933.0,
8         "intrst amt adj sum": 0.0,
9         "claim completion month adj": "July"
10      },
11      {
12        "paid amt adj sum": 14650550.0
13      }
14    ]
15  }
```

At the bottom, there are various status indicators and links: Online, Find and replace, Console, Postbot, Runner, Start Proxy, Cookies, and Trash.

POST http://127.0.0.1:8000/visualai

```
query: "lpp trend analysis by amounts",
filter: {
  "claim_completion_month_adj": "July", "May"
}
```

Status: 200 OK Time: 3.19 s Size: 1.04 KB

```
response_desc: "LPP_Trend_Analysis_Amounts",
response_data: [
  {
    "metrics": [
      {
        "SUM(intrst amt adj)": 0.0,
        "PROMPT Pay LPP Correct Amt": 0.0,
        "Prompt Pay LPP Error Amt": 0.0,
        "claim completion month adj": "August"
      }
    ]
  }
]
```

POST http://127.0.0.1:8000/visualai

```
query: "give me the claims by lpp",
filter: {
  "claim_completion_month_adj": "July", "May"
}
```

Status: 200 OK Time: 4.07 s Size: 611 B

```
response_desc: "LPP_Claims",
response_data: [
  {
    "metrics": [
      {
        "xvalue": "LPP Dt > RCVD Dt",
        "yvalue": 322,
        "zvalue": null
      },
      {
        "xvalue": "LPP Dt = RCVD Dt",
        "yvalue": 922
      }
    ]
  }
]
```

The screenshot shows the Postman application interface. At the top, there are navigation links for Home, Workspaces, API Network, and Explore, along with a search bar and user authentication options. The main workspace displays a collection named "Overview". A specific POST request is selected, targeting the URL `http://127.0.0.1:8000/visualai`. The request body is defined as follows:

```
query:"give me claims by lpp",
filter:[
  {"claim_completion_month_adj": "July", "May"}
]
```

The response status is 200 OK, with a response time of 2.86 s and a size of 625 B. The response body is displayed in Pretty JSON format:

```
response_desc: "LPP_Claims",
response_data: [
  {
    "metrics": [
      {
        "xvalue": "LPP Dt > RCVD Dt",
        "yvalue": 322,
        "zvalue": null
      },
      {
        "xvalue": "LPP Dt = RCVD Dt",
        "yvalue": 922
      }
    ]
  }
]
```

This screenshot shows the same Postman interface as the first one, but with a different query parameter. The request body is now:

```
query:"claims by lpp trends",
filter:[
  {"claim_completion_month_adj": "July", "May"}
]
```

The response status is 200 OK, with a response time of 3.02 s and a size of 1 KB. The response body is displayed in Pretty JSON format:

```
response_desc: "LPP_Trend_Analysis_Amounts",
response_data: [
  {
    "metrics": [
      {
        "SUM(intrest amt adj)": 0.0,
        "PROMPT Pay LPP Correct Amt": 0.0,
        "Prompt Pay LPP Error Amt": 0.0,
        "claim completion month adj": "August"
      },
      {
        "SUM(intrest amt adj)": 0.0
      }
    ]
  }
]
```

At the bottom of the interface, there are various status indicators and utility buttons.

The screenshot shows the Postman application interface. At the top, there are navigation tabs: Home, workspaces, API Network, and explore. A search bar is located at the top right. Below the header, there's a sidebar with sections for Overview, Collections, Environments, and History. The main workspace displays a request card for a GET request to `http://127.0.0.1:8000/visualai?query=claims by lpp trends`. The Body tab is selected, showing a JSON response with metrics. The response body is as follows:

```
1
2 "response_desc": "LPP_Trend_Analysis_Amounts",
3 "response_data": {
4     "metrics": [
5         {
6             "SUM(intrist amt adj)": 0.0,
7             "PROMPT Pay LPP Correct Amnt": 0.0,
8             "Prompt Pay LPP Error Amnt": 0.0,
9             "claim completion month adj": "August"
10        },
11        {
12            "CLM/Intrest+amt adj": 0.0
13        }
14    ]
15 }
```

Below the response, the status is shown as 200 OK with a time of 4.47 s and a size of 966 B. There are tabs for Body, Cookies, Headers (4), and Test Results. The Body tab is currently active, showing the Pretty view of the JSON response.

S

The screenshot shows the Postman application interface, similar to the previous one but with a different URL. The request card shows a GET request to `http://127.0.0.1:8000/visualai?query=claims by trend analysis by amounts`. The Body tab is selected, showing a JSON response with metrics. The response body is as follows:

```
1
2 "response_desc": "Trend_Analysis_Amounts",
3 "response_data": {
4     "metrics": [
5         {
6             "paid amt adj sum": 0.0,
7             "totl chrg amt adj sum": 96860.0,
8             "intrist amt adj sum": 0.0,
9             "claim completion month adj": "August"
10        },
11        {
12            "paid amt adj sum": 7505000.0
13        }
14    ]
15 }
```

The status is 200 OK with a time of 9.86 s and a size of 1.05 KB. The interface is identical to the first screenshot, with tabs for Body, Cookies, Headers (4), and Test Results.

POST http://127.0.0.1:8000/visualai

```
1 ... "query": "claims by state",
2 ... "filter": {
3 ...   "claim_completion_month_adj": "July", "May"
4 }
```

Status: 200 OK Time: 3.60 s Size: 723 B Save as Example

```
1
2   "response_desc": "State",
3   "response_data": [
4     {
5       "metrics": [
6         {
7           "xvalue": "CA",
8           "yvalue": 183,
9           "zvalue": null
10      },
11      {
12        "xvalue": "NY",
13        "yvalue": 454,
14      }
15    ]
16  }
17]
```

POST http://127.0.0.1:8000/visualai

```
1 ... "query": "claims by provider",
2 ... "filter": {
3 ...   "claim_completion_month_adj": "July", "May"
4 }
```

Status: 200 OK Time: 54.33 s Size: 52.58 KB Save as Example

```
1
2   "response_desc": "Provider",
3   "response_data": [
4     {
5       "metrics": [
6         {
7           "xvalue": "247 SUPPLY INC",
8           "yvalue": 1,
9           "zvalue": null
10      },
11      {
12        "xvalue": "A P NY COMPREHENSIVE MED",
13        "yvalue": 1,
14      }
15    ]
16  }
17]
```

Online Find and replace Console Postbot Runner Start Proxv Cookies Trash

The screenshot shows the Postman application interface. At the top, there are navigation buttons for Home, Workspaces, API Network, and Explore, along with a search bar labeled "Search Postman". On the right side, there are buttons for "Invite", "Upgrade", and a close button. The main workspace is titled "My Workspace" and contains a collection named "New Collection". A POST request is being made to "http://127.0.0.1:8000/visualai". The "Body" tab is selected, showing the following JSON payload:

```
1
2   "query": "claims by group",
3   "filter": {
4     "claim_completion_month_adj": "July", "May"
5   }
6 }
```

The response status is 200 OK, with a time of 17.98 s and a size of 20.68 KB. The response body is displayed in Pretty format:

```
1
2   "response_desc": "Group",
3   "response_data": {
4     "metrics": [
5       {
6         "xvalue": "2012 USP HOLDINGS, INC.",
7         "yvalue": 1,
8         "zvalue": null
9       },
10      {
11        "xvalue": "2012 USP HOLDINGS, INC.",
12        "yvalue": 1,
13        "zvalue": null
14      }
15    ]
16  }
17 }
```

The screenshot shows the Postman application interface, identical to the one above but with a different URL. The main workspace is titled "My Workspace" and contains a collection named "New Collection". A POST request is being made to "http://127.0.0.1:8000/visualai". The "Body" tab is selected, showing the following JSON payload:

```
1
2   "query": "claims by subgroup",
3   "filter": {
4     "claim_completion_month_adj": "July", "May"
5   }
6 }
```

The response status is 200 OK, with a time of 17.55 s and a size of 21.83 KB. The response body is displayed in Pretty format:

```
1
2   "response_desc": "SubGroup",
3   "response_data": {
4     "metrics": [
5       {
6         "xvalue": "005709M235",
7         "yvalue": 1,
8         "zvalue": null
9       },
10      {
11        "xvalue": "005709M235",
12        "yvalue": 1,
13        "zvalue": null
14      }
15    ]
16  }
17 }
```

At the bottom of the interface, there are various status indicators and links: "Online" (with a checkmark), "Find and replace", "Console", "Postbot", "Runner", "Start Proxy", "Cookies", "Trash", and a "Help" link.

The screenshot shows the Postman application interface. A collection named "Overview" is selected. A POST request is made to <http://127.0.0.1:8000/visualai>. The request body contains the following JSON:

```
1 {
2   "query": "claims by lob",
3   "filter": {
4     "claim_completion_month_adj": "July", "May"
5   }
6 }
```

The response status is 200 OK, with a response time of 4.33 s and a size of 1.28 KB. The response body is:

```
1 {
2   "response_desc": "LOB",
3   "response_data": [
4     {
5       "metrics": [
6         {
7           "LOB": "Commercial Business",
8           "Number of claims": 53,
9           "Zone": "Corporate"
10        }
11      ]
12    }
13  ]
14 }
```

The screenshot shows the Postman application interface. A collection named "Overview" is selected. A POST request is made to <http://127.0.0.1:8000/visualai>. The request body contains the following JSON:

```
1 {
2   "query": "claims by member",
3   "filter": {
4     "claim_completion_month_adj": "July", "May"
5   }
6 }
```

The response status is 200 OK, with a response time of 22.80 s and a size of 25.05 KB. The response body is:

```
1 {
2   "response_desc": "Member",
3   "response_data": [
4     {
5       "metrics": [
6         {
7           "xvalue": "100247095",
8           "yvalue": 1,
9           "zvalue": null
10        }
11      ]
12    }
13  ]
14 }
```

Home Workspaces API Network Explore Search Postman

POST http://127.0.0.1:8000/visualai

Body (8) Headers Params Authorization Tests Settings Cookies

Text

```
1
2
3
4
5
6
```

Body Cookies Headers (4) Test Results

Pretty Raw Preview Visualize JSON

```
1
2
3
4
5
6
7
8
9
10
```

Status: 200 OK Time: 4.83 s Size: 908 B Save as Example

```
"response_desc": "Adj_Reason",
"response_data": {
  "metrics": [
    {
      "xvalue": "S10 MRU DECISION",
      "yvalue": 1,
      "zvalue": null
    }
  ]
}
```

Home Workspaces API Network Explore Search Postman

POST http://127.0.0.1:8000/visualai

Body (8) Headers Params Authorization Tests Settings Cookies

Text

```
1
2
3
4
5
6
```

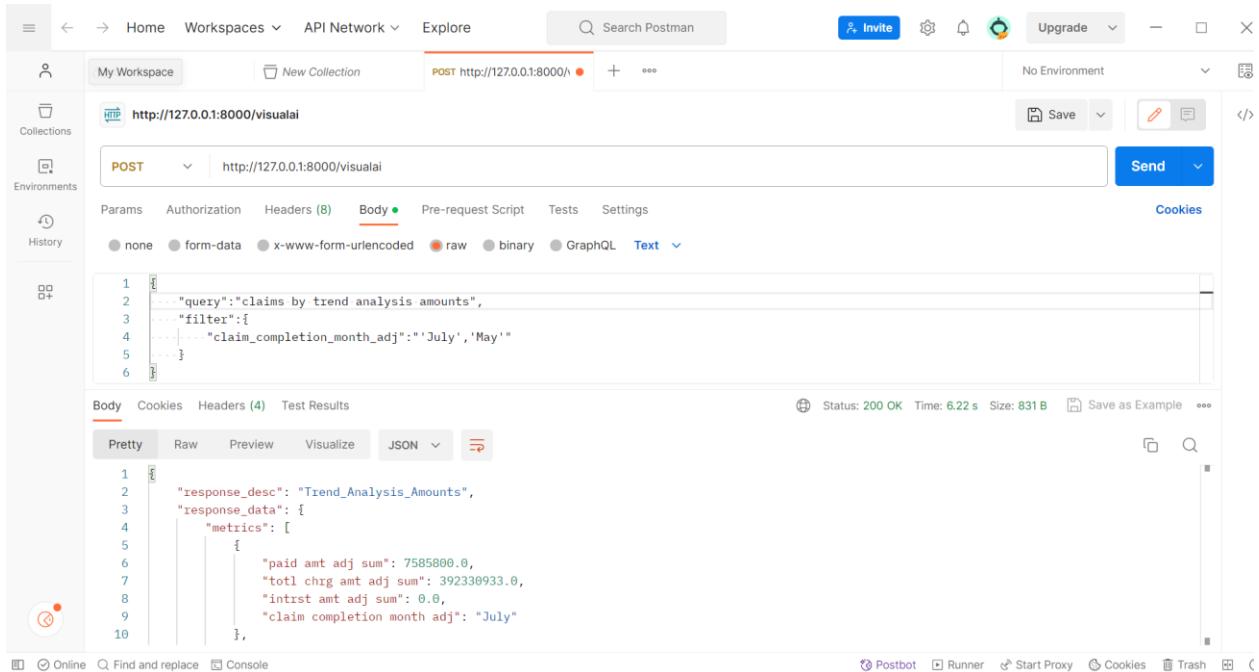
Body Cookies Headers (4) Test Results

Pretty Raw Preview Visualize JSON

```
1
2
3
4
5
6
7
8
9
10
```

Status: 200 OK Time: 3.08 s Size: 1.18 KB Save as Example

```
"response_desc": "Trend_Analysis_Claims_Volumes",
"response_data": {
  "metrics": [
    {
      "SUM(intrst amt adj)": 0.0,
      "Prompt pay claims": 337.0,
      "Claims LPP Error Amt": 2590358076242.0,
      "Claims LPP Correct Amt": 138570699409.0,
      "Number of Total Claims": 602,
      "Avg LPP Error Amt": 428392.0
    }
  ]
}
```



POST http://127.0.0.1:8000/visualai

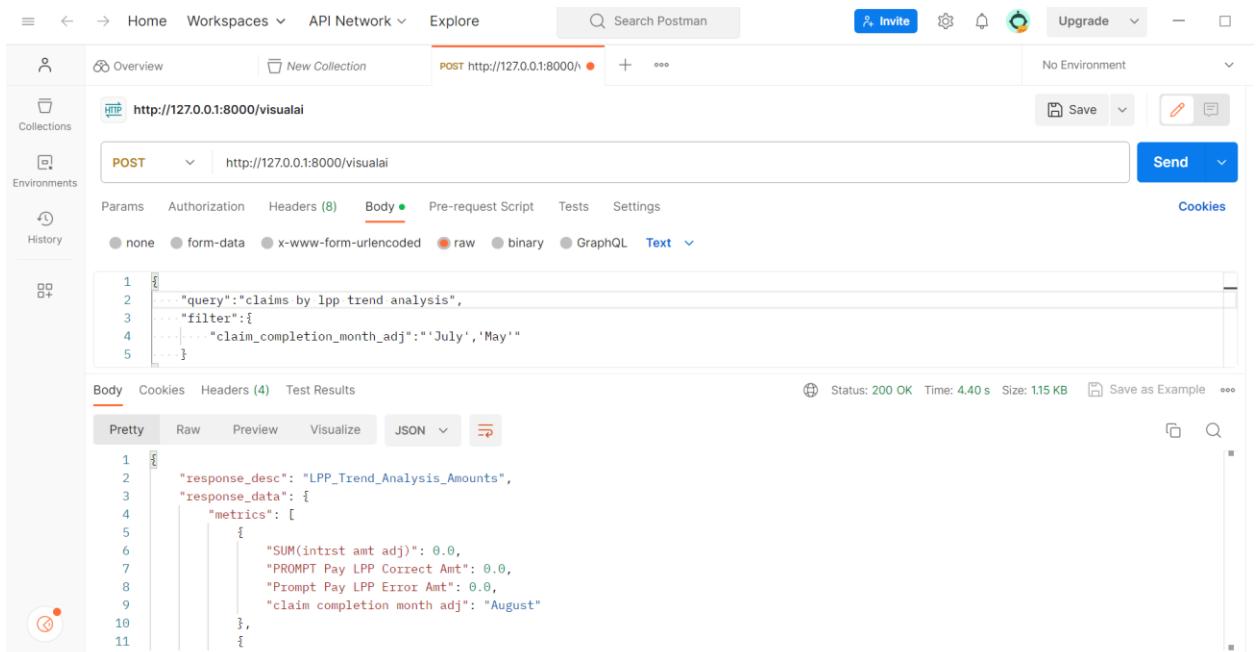
Body (8)

```
1 "query": "claims by trend analysis amounts",
2 "filter": [
3     {"claim_completion_month_adj": "July", "May"}
4 ]
```

Pretty Raw Preview Visualize JSON

1 "response\_desc": "Trend\_Analysis\_Amounts",
2 "response\_data": {
3 "metrics": [
4 {
5 "paid amt adj sum": 7585800.0,
6 "totl chrg amt adj sum": 392330933.0,
7 "intrst amt adj sum": 0.0,
8 "claim completion month adj": "July"
9 }
10 ]
11 }

Status: 200 OK Time: 6.22 s Size: 831 B Save as Example



POST http://127.0.0.1:8000/visualai

Body (8)

```
1 "query": "claims by lpp trend analysis",
2 "filter": [
3     {"claim_completion_month_adj": "July", "May"}
4 ]
```

Pretty Raw Preview Visualize JSON

1 "response\_desc": "LPP\_Trend\_Analysis\_Amounts",
2 "response\_data": {
3 "metrics": [
4 {
5 "SUM(intrest amt adj)": 0.0,
6 "PROMPT Pay LPP Correct Amt": 0.0,
7 "Prompt Pay LPP Error Amt": 0.0,
8 "claim completion month adj": "August"
9 }
10 ]
11 }

Status: 200 OK Time: 4.40 s Size: 1.15 KB Save as Example

The screenshot shows the Postman application interface. At the top, there are navigation icons (Home, Workspaces, API Network, Explore) and a search bar. On the left, there's a sidebar with sections for Overview, Collections, Environments, and History. The main area displays a POST request to `http://127.0.0.1:8000/visualai`. The request details show the method as POST, the URL as `http://127.0.0.1:8000/visualai`, and the Body tab selected. The body content is a JSON object:

```
1 "query": "How am i doing by LPP claims",
2 "filter": {
3     "claim_completion_month_adj": "July", "May"
4 }
```

The response status is 200 OK, with a time of 5.84 s and a size of 620 B. The response body is displayed in Pretty format:

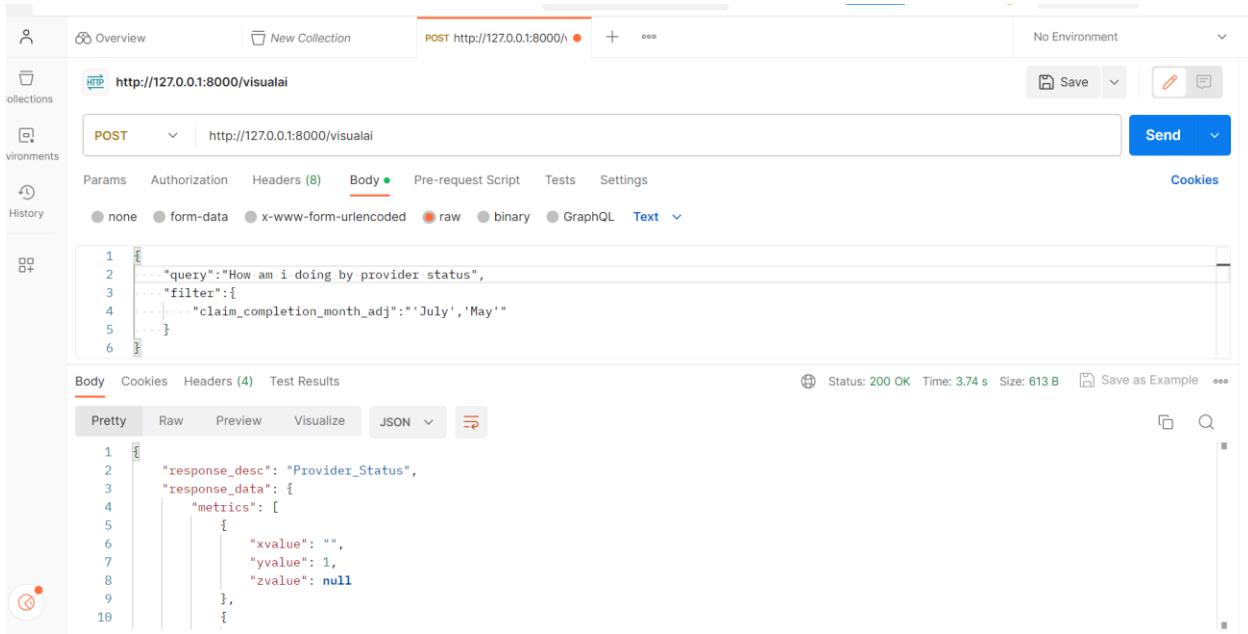
```
1 "response_desc": "LPP_Claims",
2 "response_data": {
3     "metrics": [
4         {
5             "xvalue": "LPP Dt > RCVD Dt",
6             "yvalue": 322,
7             "zvalue": null
8         }
9     ]
10 }
```

This screenshot shows another instance of the Postman interface with a similar setup. It has a POST request to `http://127.0.0.1:8000/visualai`. The body content is a JSON object:

```
1 "query": "How am i doing by population volume",
2 "filter": {
3     "claim_completion_month_adj": "July", "May"
4 }
```

The response status is 200 OK, with a time of 3.54 s and a size of 474 B. The response body is displayed in Pretty format:

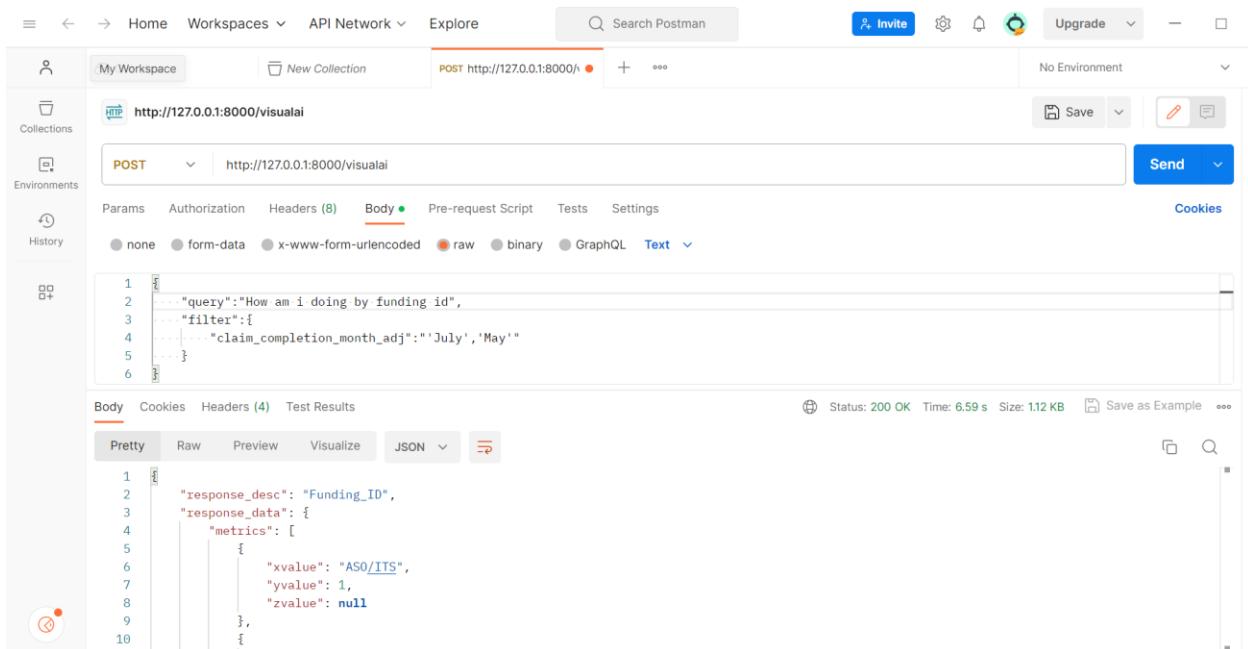
```
1 "response_desc": "Population_Volume",
2 "response_data": {
3     "metrics": [
4         {
5             "xvalue": "HOME",
6             "yvalue": 101,
7             "zvalue": null
8         }
9     ]
10 }
```



POST http://127.0.0.1:8000/visualai

```
1 "query": "How am i doing by provider status",
2 "filter": {
3     "claim_completion_month_adj": "July", 'May'
4 }
```

Status: 200 OK Time: 3.74 s Size: 613 B Save as Example



POST http://127.0.0.1:8000/visualai

```
1 "query": "How am i doing by funding id",
2 "filter": {
3     "claim_completion_month_adj": "July", 'May'
4 }
```

Status: 200 OK Time: 6.59 s Size: 1.12 KB Save as Example

The screenshot shows the Postman application interface. At the top, there are navigation links for Home, Workspaces, API Network, and Explore, along with a search bar and user account options. The main workspace displays a collection named "Overview". A specific POST request is selected, targeting the URL `http://127.0.0.1:8000/visualai`. The request's Body tab is active, showing the following JSON payload:

```
1
2   "query": "How am i doing by lpp claims",
3   "filter": [
4     {
5       "claim_completion_month_adj": "July", "May"
6     }
7   ]
```

The response status is 200 OK, with a time of 3.09 s and a size of 626 B. The response body is displayed in Pretty format:

```
1
2   "response_desc": "LPP_Claims",
3   "response_data": {
4     "metrics": [
5       {
6         "xvalue": "LPP Dt > RCVD Dt",
7         "yvalue": 322,
8         "zvalue": null
9       },
10      {
11        "xvalue": "LPP Dt > RCVD Dt",
12        "yvalue": 322,
13        "zvalue": null
14      }
15    ]
16  }
```

This screenshot shows another instance of the Postman interface with a similar setup. A POST request is made to `http://127.0.0.1:8000/visualai`. The Body tab is active, displaying the same JSON payload as the first screenshot:

```
1
2   "query": "How am i doing by prompt pay",
3   "filter": [
4     {
5       "claim_completion_month_adj": "July", "May"
6     }
7   ]
```

The response status is 200 OK, with a time of 3.21 s and a size of 447 B. The response body is displayed in Pretty format:

```
1
2   "response_desc": "Prompt_Pay",
3   "response_data": {
4     "metrics": [
5       {
6         "xvalue": "0",
7         "yvalue": 505,
8         "zvalue": null
9       },
10      {
11        "xvalue": "0",
12        "yvalue": 505,
13        "zvalue": null
14      }
15    ]
16  }
```

The screenshot shows the Postman application interface. At the top, there are navigation tabs: Home, Workspaces, API Network, and Explore. A search bar says "Search Postman". On the right, there are buttons for Invite, Upgrade, and a close button. The main area shows a collection named "Overview" with a "New Collection" button. A POST request is selected with the URL "http://127.0.0.1:8000/visualai". The "Body" tab is active, showing a JSON payload:

```
1 {  
2   "query": "How am i doing by state",  
3   "filter": [  
4     {"claim_completion_month_adj": "July", "May": "  
5   }  
6 ]
```

Below the body, the response status is shown as "Status: 200 OK Time: 5.68 s Size: 571 B". The response body is displayed in "Pretty" format:

```
1 {  
2   "response_desc": "State",  
3   "response_data": {  
4     "metrics": [  
5       {  
6         "xvalue": "CA",  
7         "yvalue": 183,  
8         "zvalue": null  
9       },  
10      {
```

The screenshot shows the Postman application interface. At the top, there are navigation tabs: Home, workspaces, API Network, and Explore. A search bar is located at the top right. Below the header, there's a sidebar with sections for Overview, Collections, Environments, and History.

The main workspace displays a POST request to <http://127.0.0.1:8000/visualai>. The request details are as follows:

- Method: POST
- URL: http://127.0.0.1:8000/visualai
- Body tab is selected, showing the following JSON payload:

```
1 ...
2 ...
3 ...
4 ...
5 ...
6 ...
7 ...
8 ...
9 ...
10 ...
```

The payload contains a query string with parameters like "claims by provider status", "filter", and "claim\_completion\_month\_adj".

Below the Body section, the Response tab is active, showing the following JSON response:

```
1 {
2   "response_desc": "Provider_Status",
3   "response_data": [
4     {
5       "metrics": [
6         {
7           "xvalue": "OUT",
8           "yvalue": 164,
9           "zvalue": null
10      }
11    ]
12  }
13]
```

The response indicates a successful 200 OK status with a time of 16.31s and a size of 434 B. There are also links to "Save as Example" and other options.



```
      ^^^^^^^^^^
File "C:\ProgramData\Anaconda3\envs\venv\Lib\site-packages\pandas\io\sql.py", line 654, in read_sql
    return pandas_sql.read_query(
          ^^^^^^^^^^
File "C:\ProgramData\Anaconda3\envs\venv\Lib\site-packages\pandas\io\sql.py", line 2330, in read_query
    cursor = self.execute(sql, params)
          ^^^^^^^^^^
File "C:\ProgramData\Anaconda3\envs\venv\Lib\site-packages\pandas\io\sql.py", line 2278, in execute
    raise ex from exc
pandas.errors.DatabaseError: Execution failed on sql 'SELECT in_out_ntwk_cd_ori, COUNT(in_out_ntwk_cd_ori)  FROM claimsDeepDive
where claim_completion_month_adj IN ('July','May') and edit_cd_ori IN ('LUM WGP DP1','BNQ SPS PAM') and edit_cd_adj IN ('LUM
WGP DP1','BNQ SPS PAM') and hlth_srvc_cd IN ('UNK',' 99212') and rvnu_cd IN ('UNK',' 0361') and diag_cd IN ('I10','F411') and u
b_or_prof_ori IN ('Facility','Professional') and fncl_mbu_lvl_5_desc_ori IN ('Commercial Business','Speciality Products BU') and bi
llg_tax_id IN ('840166760','134070634') and pay_actn_cd_ori IN ('RMCR00','RMCR00') GROUP BY in_out_ntwk_cd_ori;': 1054 (42S22): U
nknown column 'billg_tax_id' in 'where clause'
```

SS

```
      ^^^^^^
return pandas_sql.read_query(
      ^^^^^^
File "C:\ProgramData\Anaconda3\envs\venv\Lib\site-packages\pandas\io\sql.py", line 2330, in read_query
    cursor = self.execute(sql, params)
          ^^^^^^
File "C:\ProgramData\Anaconda3\envs\venv\Lib\site-packages\pandas\io\sql.py", line 2278, in execute
    raise ex from exc
pandas.errors.DatabaseError: Execution failed on sql 'SELECT in_out_ntwk_cd_ori, COUNT(in_out_ntwk_cd_ori)  FROM claimsDeepDive
where claim_completion_month_adj IN ('July','May') and edit_cd_ori IN ('LUM WGP DP1','BNQ SPS PAM') and edit_cd_adj IN ('LUM
WGP DP1','BNQ SPS PAM') and hlth_srvc_cd IN ('UNK',' 99212') and rvnu_cd IN ('UNK',' 0361') and diag_cd IN ('I10','F411') and u
b_or_prof_ori IN ('Facility','Professional') and fncl_mbu_lvl_5_desc_ori IN ('Commercial Business','Speciality Products BU') and pa
y_actn_cd_ori IN ('RMCR00','RMCR00') GROUP BY in_out_ntwk_cd_ori;': 1054 (42S22): Unknown column 'pay_actn_cd_ori' in 'where clau
se'
```

{

```
  "query":"claims by provider status",
  "filter":{

    "claim_completion_month_adj":"July','May",

    "edit_cd_ori":"LUM WGP DP1','BNQ SPS PAM",

    "edit_cd_adj":"LUM WGP DP1','BNQ SPS PAM",

    "hlth_srvc_cd":"UNK","\n99212",

    "rvnu_cd":"UNK","\n0361",

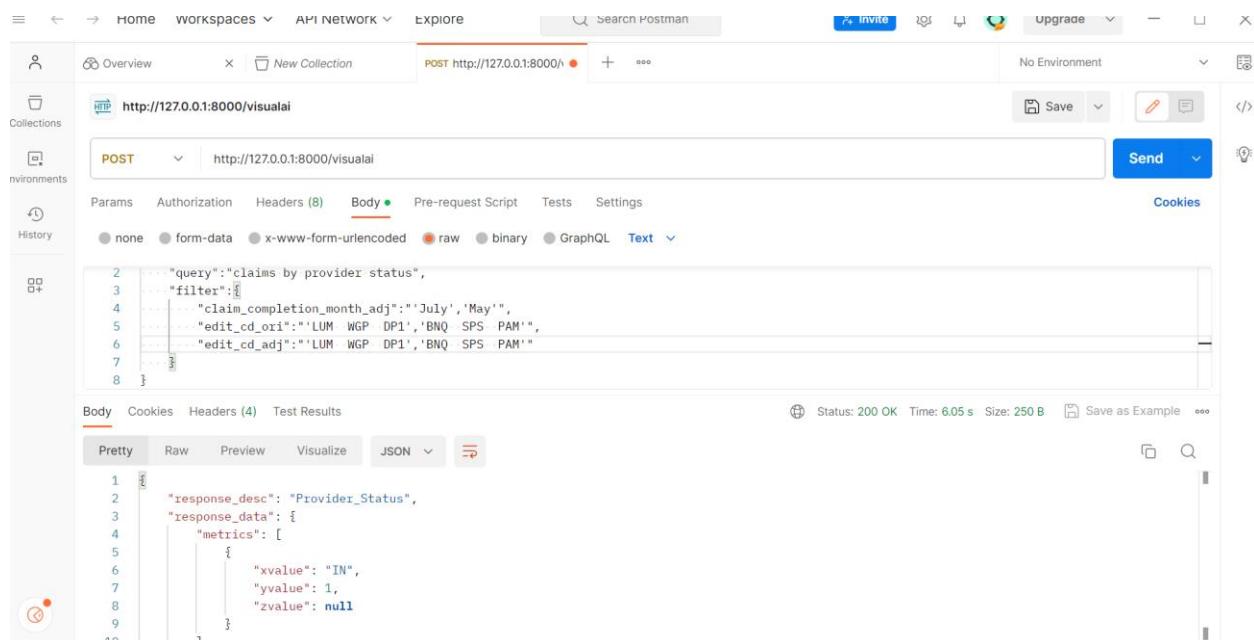
    "diag_cd":"I10','F411",

    "ub_or_prof_ori":"Facility','Professional",

    "fncl_mbu_lvl_5_desc_ori":"Commercial Business','Speciality Products BU"

  }
```

}

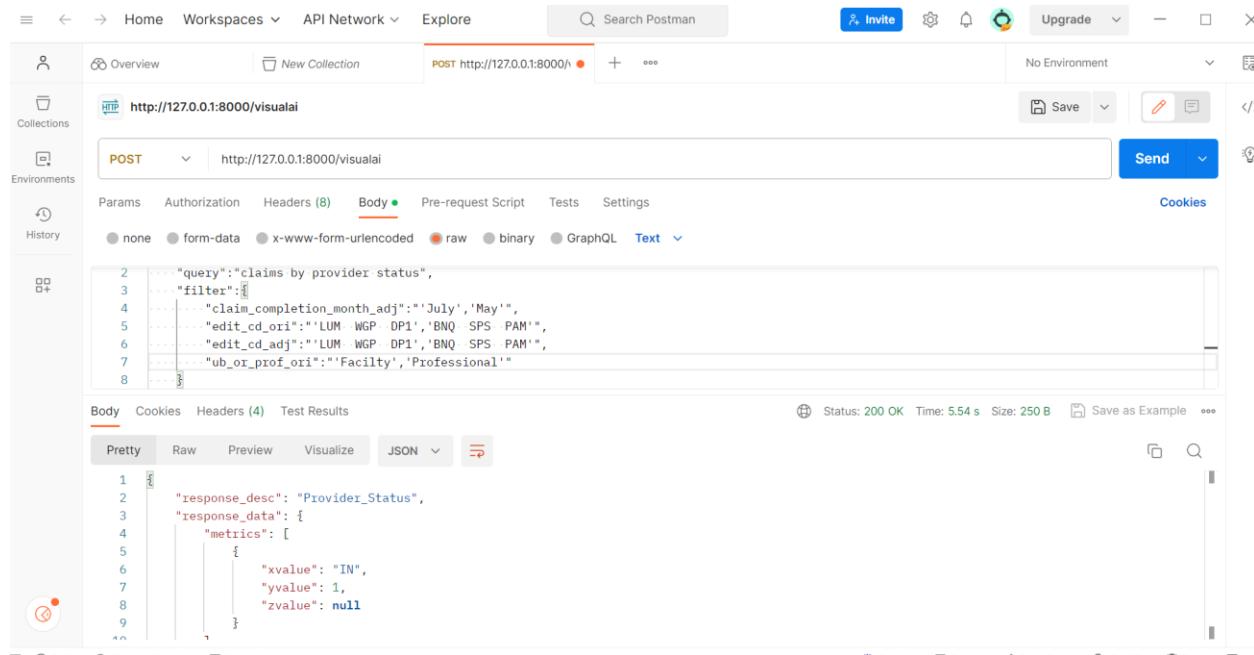


POST http://127.0.0.1:8000/visualai

```
query: "claims by provider status",
filter: [
  "claim_completion_month_adj": "July", "May",
  "edit_cd_ori": "LUM WGP DP1", "BNQ SPS PAM",
  "edit_cd_adj": "LUM WGP DP1", "BNQ SPS PAM"
]
```

Status: 200 OK Time: 6.05 s Size: 250 B

```
response_desc: "Provider_Status",
response_data: {
  "metrics": [
    {
      "xvalue": "IN",
      "yvalue": 1,
      "zvalue": null
    }
  ]
}
```



POST http://127.0.0.1:8000/visualai

```
query: "claims by provider status",
filter: [
  "claim_completion_month_adj": "July", "May",
  "edit_cd_ori": "LUM WGP DP1", "BNQ SPS PAM",
  "edit_cd_adj": "LUM WGP DP1", "BNQ SPS PAM",
  "ub_or_prof_ori": "Facility", "Professional"
]
```

Status: 200 OK Time: 5.54 s Size: 250 B

```
response_desc: "Provider_Status",
response_data: {
  "metrics": [
    {
      "xvalue": "IN",
      "yvalue": 1,
      "zvalue": null
    }
  ]
}
```

Home Workspaces API Network Explore Search Postman Invite Upgrade

My Workspace New Collection POST http://127.0.0.1:8000/ No Environment

HTTP http://127.0.0.1:8000/visualai

POST http://127.0.0.1:8000/visualai

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body (Text)

```
3 "filter":{  
4     "claim_completion_month_adj": "July", "May"  
5     "edit_cd_ori": "LUM WGP DP1", "BNQ SPS PAM",  
6     "edit_cd_adj": "LUM WGP DP1", "BNQ SPS PAM",  
7     "fncl_mbu_lvl_5_desc_ori": "Commercial Business", "Speciality Products BU"  
8 }  
9 }
```

Status: 200 OK Time: 35.25 s Size: 250 B Save as Example

Pretty Raw Preview Visualize JSON

```
1 {  
2     "response_desc": "Provider_Status",  
3     "response_data": {  
4         "metrics": [  
5             {  
6                 "xvalue": "IN",  
7                 "value": 1,  
8                 "zvalue": null  
9             }  
10        ]  
11    }  
12}
```