PhonePe is an Indian Digital payments company headquartered in Bengaluru Karnataka India. It is could be one of the most common UPI app you are using for making peer to peer transactions and payments.

> **Analysis Statement.**
>
> Every quarter company provides JSON data of different aggregate values in different categories this tradition is happening from 2018 onwards. Lets say you are an economist try to make an analysis of penetration of digital payments in different parts of India state wise and year wise. As a stock analyst perspective which year would have been the best year to invest in this company.

- Here it the link of the Repository Provided by the PhonePe Company. The README data of the repository tries to explain the folder structure. But We have provided the grid view of the `json` data in different folders in a grid view for your understanding.
- Now based on the data in the repository Phone pay has an extraordinary website have a look into it here which could help you on how to do analysis on the data .

What is `JSON` and how to work with it.

What is JSON Format?

JSON (JavaScript Object Notation) is a lightweight data-interchange format that is easy for humans to read and write and easy for machines to parse and generate. It is primarily used to transmit data between a server and a web application as text. JSON is language-independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others.

> In simple terms it is the query output from your data base.

Here is an example of JSON data:

```
{
    "name": "John",
    "age": 30,
    "isStudent": false,
    "courses": ["Math", "Science", "History"],
    "address": {
        "street": "123 Main St",
        "city": "Anytown",
        "zipcode": "12345"
    }
}
```

> Have you noticed something here JSON is nothing but a dictionary in python which could be nested and can have lists in them as well. So one you import it you can work with it like a dictionary.

Working with JSON in Python

Python provides a built-in library called `json` for handling JSON data. Here's how you can work with JSON files in Python with that library:

Step-by-Step Example

1. Lets say you have a `data.json` file by the way it can have any name as long as it is a JSON file.
2. Below is an example showing you how to import that data.

```python
import json

# Specify the path to the JSON file
file_path = 'data.json'

# Open the JSON file and load its content
with open(file_path, 'r') as file:
    data = json.load(file)

# Print the loaded data
print(data)
```

Explanation

1. **Import the `json` Library**: Import the built-in `json` library, which provides functions for working with JSON data.
2. **Specify the Path to the JSON File**: Define the path to the JSON file. In this example, it's `'data.json'`.
3. **Open the JSON File and Load Its Content**: Use a `with` statement to open the JSON file in read mode (`'r'`). The `json.load()` function reads the JSON data from the file and converts it into a Python dictionary.
4. **Print the Loaded Data**: Print the data to verify that it has been correctly loaded into the Python program.

Output

When you run the above code, it will print the content of the JSON file as a Python dictionary:

```
{
    'name': 'John',
    'age': 30,
    'isStudent': False,
    'courses': ['Math', 'Science', 'History'],
    'address': {
        'street': '123 Main St',
        'city': 'Anytown',
        'zipcode': '12345'
    }
}
```

This is a simple and effective way to import and work with JSON data in Python.

Now you can even get key values using simple python code you know :

Just an example on how to access different key values :

```python
print(data['name'])         # Output: John

print(data['age'])          # Output: 30

print(data['isStudent'])    # Output: False

print(data['courses'])      # Output: ['Math', 'Science', 'History']

print(data['courses'][0])   # Output: Math

print(data['address'])      # Output: {'street': '123 Main St', 'city': 'Anytown', 'zipcode': '12345'}

print(data['address']['street'])  # Output: 123 Main St

print(data['address']['city'])    # Output: Anytown

print(data['address']['zipcode']) # Output: 12345
```

Like this you can access different key values and have them in a flat file (CSV file.) this is what you are supposed to accomplish so that you can work with the data for analysis purpose.

PhonePe Pulse JSON Data

Complete File Info :

```
===============================================================================
 Language            Files        Lines         Code     Comments       Blanks
===============================================================================
 JSON                 7918         8278         8275            0            3
-------------------------------------------------------------------------------
 Markdown                1          188            0          135           53
 |- JavaScript           1          292          268            0           24
 (Total)                            480          268          135           77
===============================================================================
 Total                7919         8466         8275          135           56
===============================================================================
```

The Data Folder Section :

Total Folder Section : All the data files and How they are arranged :

```
.
├── aggregated
│   ├── insurance
│   │   └── country
│   │       └── india
│   │           ├── 2020
│   │           ├── 2021
│   │           ├── 2022
│   │           ├── 2023
│   │           ├── 2024
│   │           └── state
│   ├── transaction
│   │   └── country
│   │       └── india
│   │           ├── 2018
│   │           ├── 2019
│   │           ├── 2020
│   │           ├── 2021
│   │           ├── 2022
│   │           ├── 2023
│   │           ├── 2024
│   │           └── state
│   └── user
│       └── country
│           └── india
│               ├── 2018
│               ├── 2019
│               ├── 2020
│               ├── 2021
│               ├── 2022
│               ├── 2023
│               ├── 2024
│               └── state
├── map
│   ├── insurance
│   │   ├── country
│   │   │   └── india
│   │   │       ├── 2020
│   │   │       ├── 2021
│   │   │       ├── 2022
│   │   │       ├── 2023
│   │   │       ├── 2024
│   │   │       └── state
│   │   └── hover
│   │       └── country
│   │           └── india
│   ├── transaction
│   │   └── hover
│   │       └── country
│   │           └── india
│   └── user
│       └── hover
│           └── country
│               └── india
└── top
    ├── insurance
    │   └── country
    │       └── india
    │           ├── 2020
    │           ├── 2021
    │                   ├── 2022
    │                   ├── 2023
    │                   ├── 2024
    │                   ├── state
    │                   └── state.zip
    ├── transaction
    │   └── country
    │       └── india
    │           ├── 2018
    │           ├── 2019
    │           ├── 2020
    │           ├── 2021
    │           ├── 2022
    │           ├── 2023
    │           ├── 2024
    │           └── state
    └── user
        └── country
            └── india
                ├── 2018
                ├── 2019
                ├── 2020
                ├── 2021
                ├── 2022
                ├── 2023
                ├── 2024
                └── state
```

There are 3 Major Sections in In the data :

```
.
├── aggregated --> Aggreagte
├── map --> Details on Map
└── top --> Top Districts
```

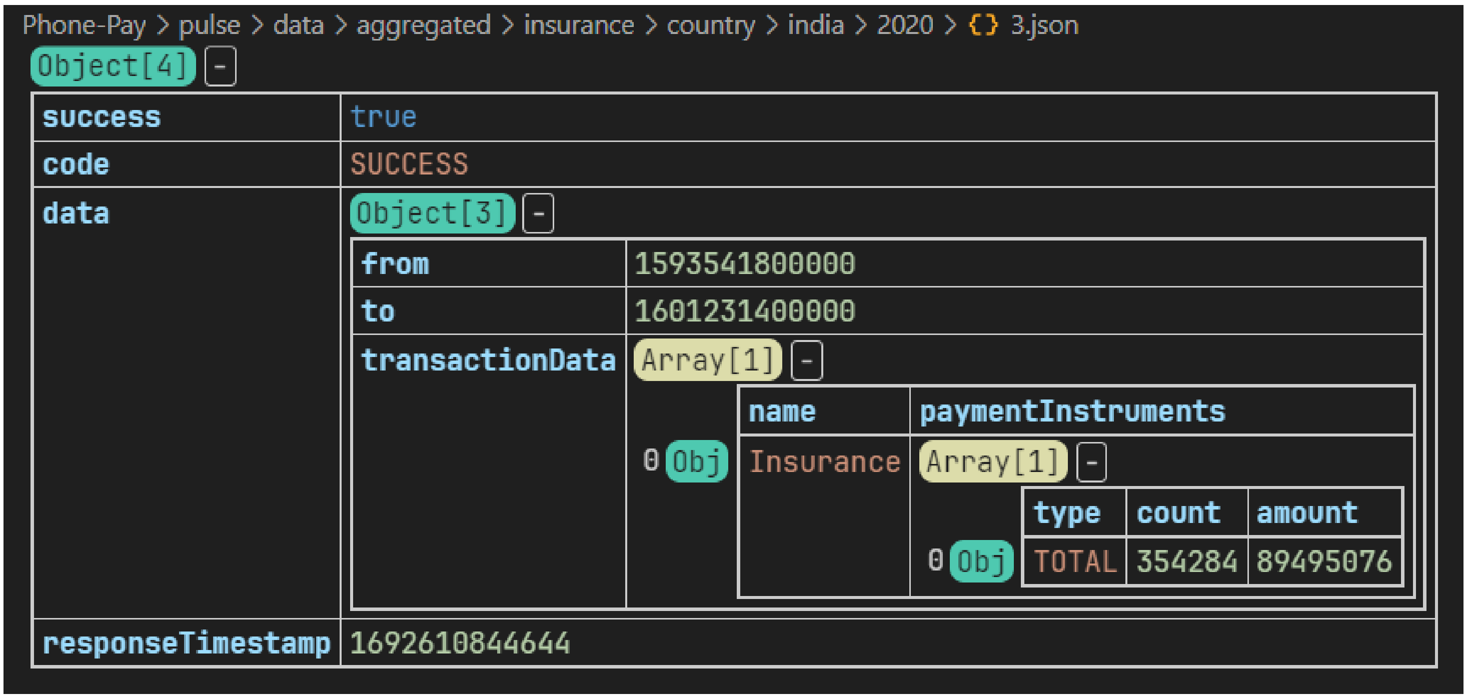
aggregated

Files in Aggregate folder :

```
===============================================================================
 Language            Files        Lines         Code     Comments       Blanks
===============================================================================
 JSON                 2442         2569         2569            0            0
===============================================================================
 Total                2442         2569         2569            0            0
===============================================================================
```

```
.
├── insurance
│   └── country
│       └── india
│           ├── 2020 -
│           ├── 2021   |
│       ├── 2022   | --> Yearly Data
│       ├── 2023   |
│       ├── 2024 -
│       └── state -->
├── transaction
│   └── country
│       └── india
│           ├── 2018
│           ├── 2019
│           ├── 2020
│           ├── 2021
│           ├── 2022
│           ├── 2023
│           ├── 2024
│           └── state
└── user
    └── country
        └── india
            ├── 2018
            ├── 2019
            ├── 2020
            ├── 2021
            ├── 2022
            ├── 2023
            ├── 2024
            └── state
```

- **File Data In insurance → country → India → (Year Folders [2018, 2019, 2020, 2021, 2022, 2023, 2024])**

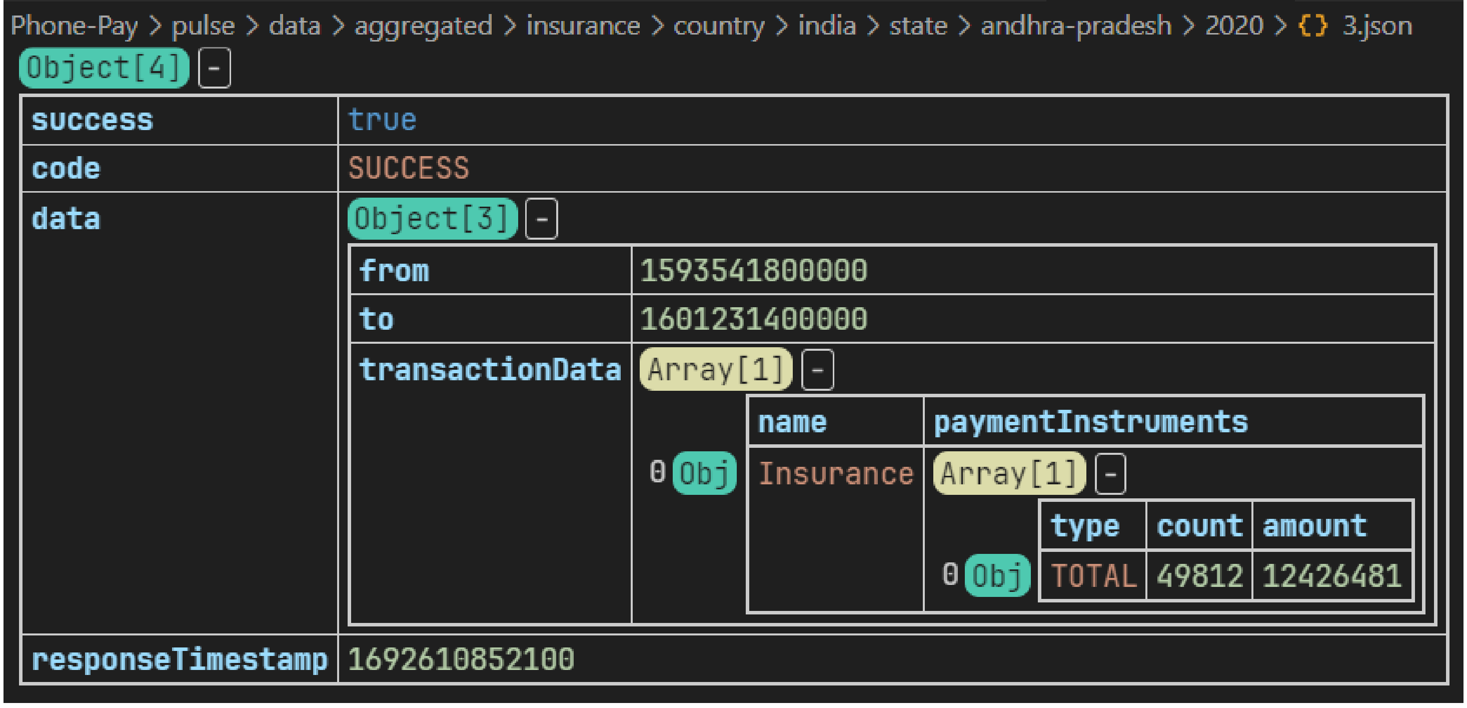


Phone-Pay > pulse > data > aggregated > insurance > country > india > 2020 > {} 3.json

Object[4]

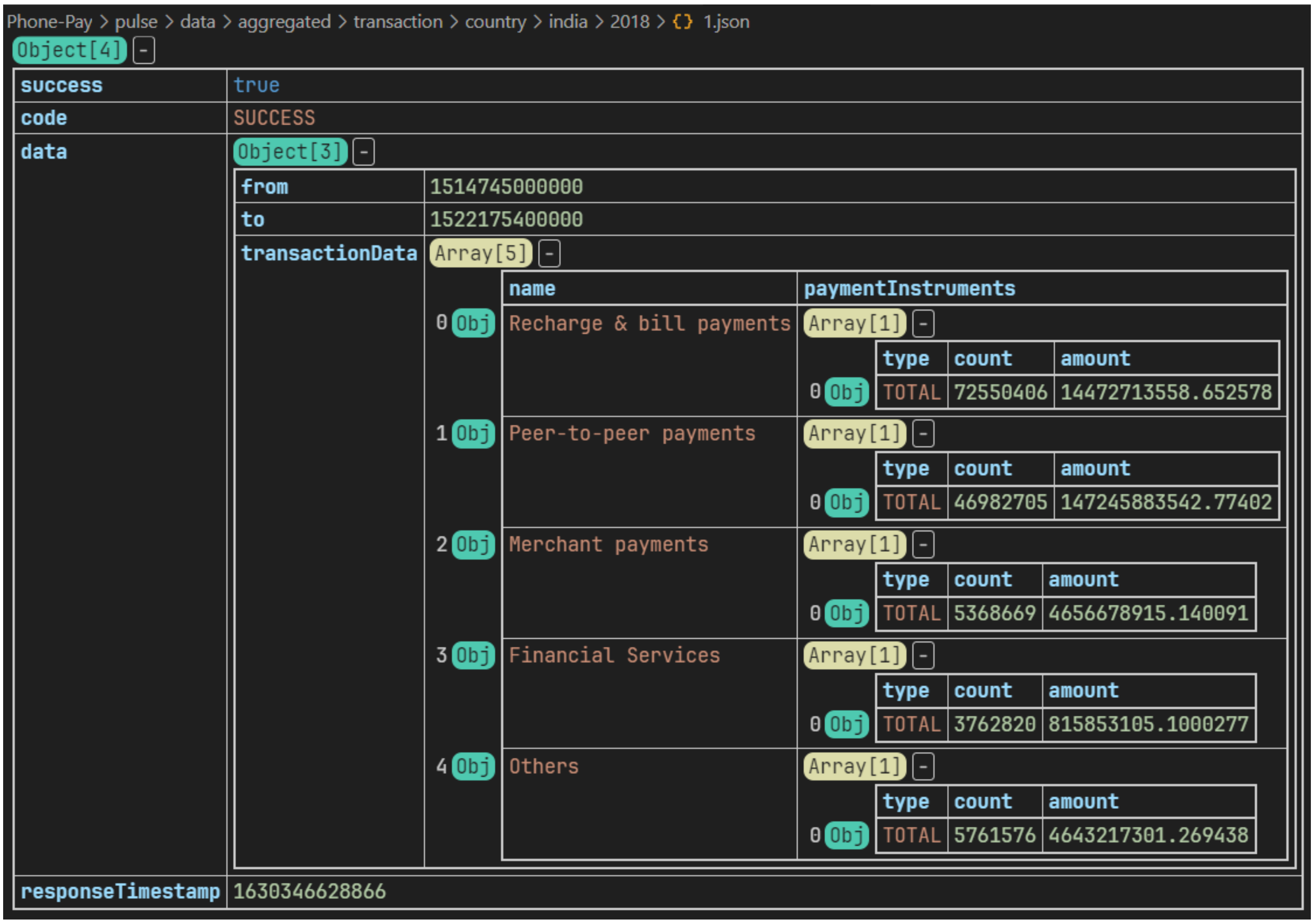
success	true
code	SUCCESS
data	Object[3]: from 1593541800000; to 1601231400000; transactionData Array[1]: 0 Obj — name: Insurance; paymentInstruments Array[1]: 0 Obj — type: TOTAL, count: 354284, amount: 89495076
responseTimestamp	1692610844644

- **File Data In insurance → country → India → (State [Andhra, Telangana, Delhi, ...])**



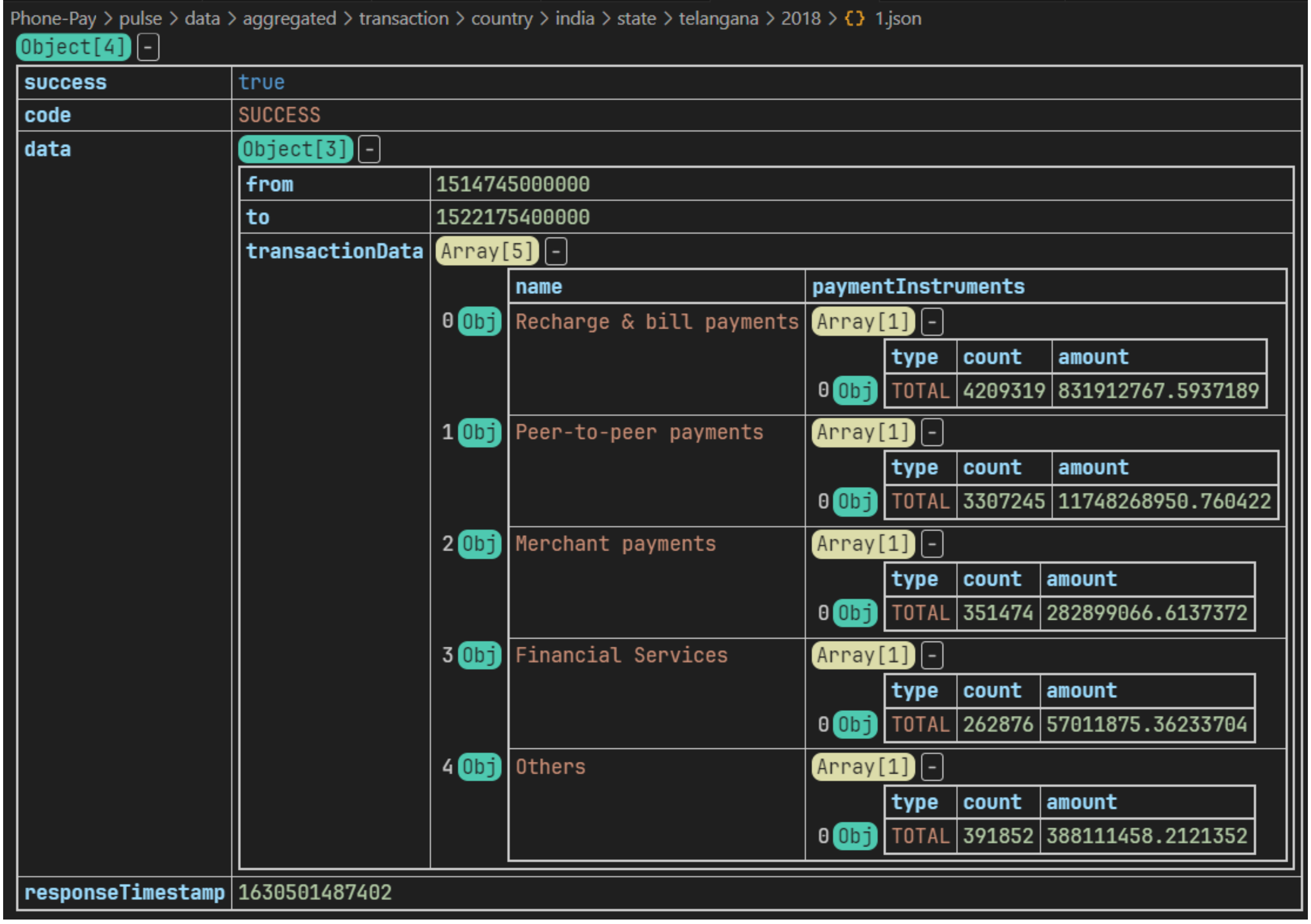
```
Phone-Pay > pulse > data > aggregated > insurance > country > india > state > andhra-pradesh > 2020 > {} 3.json
Object[4] -
success            true
code               SUCCESS
data               Object[3] -
                   from             1593541800000
                   to               1601231400000
                   transactionData  Array[1] -
                                          name       paymentInstruments
                                    0 Obj Insurance  Array[1] -
                                                           type   count  amount
                                                     0 Obj TOTAL  49812  12426481
responseTimestamp  1692610852100
```

- File Data In transaction → country → India → (Year Folders [2018, 2019, 2020, 2021, 2022, 2023, 2024])



```
Phone-Pay > pulse > data > aggregated > transaction > country > india > 2018 > {} 1.json
Object[4] -
success            true
code               SUCCESS
data               Object[3] -
                   from             1514745000000
                   to               1522175400000
                   transactionData  Array[5] -
                                          name                      paymentInstruments
                                    0 Obj Recharge & bill payments  Array[1] -
                                                                          type   count     amount
                                                                    0 Obj TOTAL  72550406  14472713558.652578
                                    1 Obj Peer-to-peer payments     Array[1] -
                                                                          type   count     amount
                                                                    0 Obj TOTAL  46982705  147245883542.77402
                                    2 Obj Merchant payments         Array[1] -
                                                                          type   count    amount
                                                                    0 Obj TOTAL  5368669  4656678915.140091
                                    3 Obj Financial Services        Array[1] -
                                                                          type   count    amount
                                                                    0 Obj TOTAL  3762820  815853105.1000277
                                    4 Obj Others                    Array[1] -
                                                                          type   count    amount
                                                                    0 Obj TOTAL  5761576  4643217301.269438
responseTimestamp  1630346628866
```

- **File Data In transaction → country → India → (State [Andhra, Telangana, Delhi, ...])


Phone-Pay > pulse > data > aggregated > transaction > country > india > state > telangana > 2018 > {} 1.json

Object[4]

success	true
code	SUCCESS
data	Object[3]
responseTimestamp	1630501487402

data:

from	1514745000000
to	1522175400000
transactionData	Array[5]

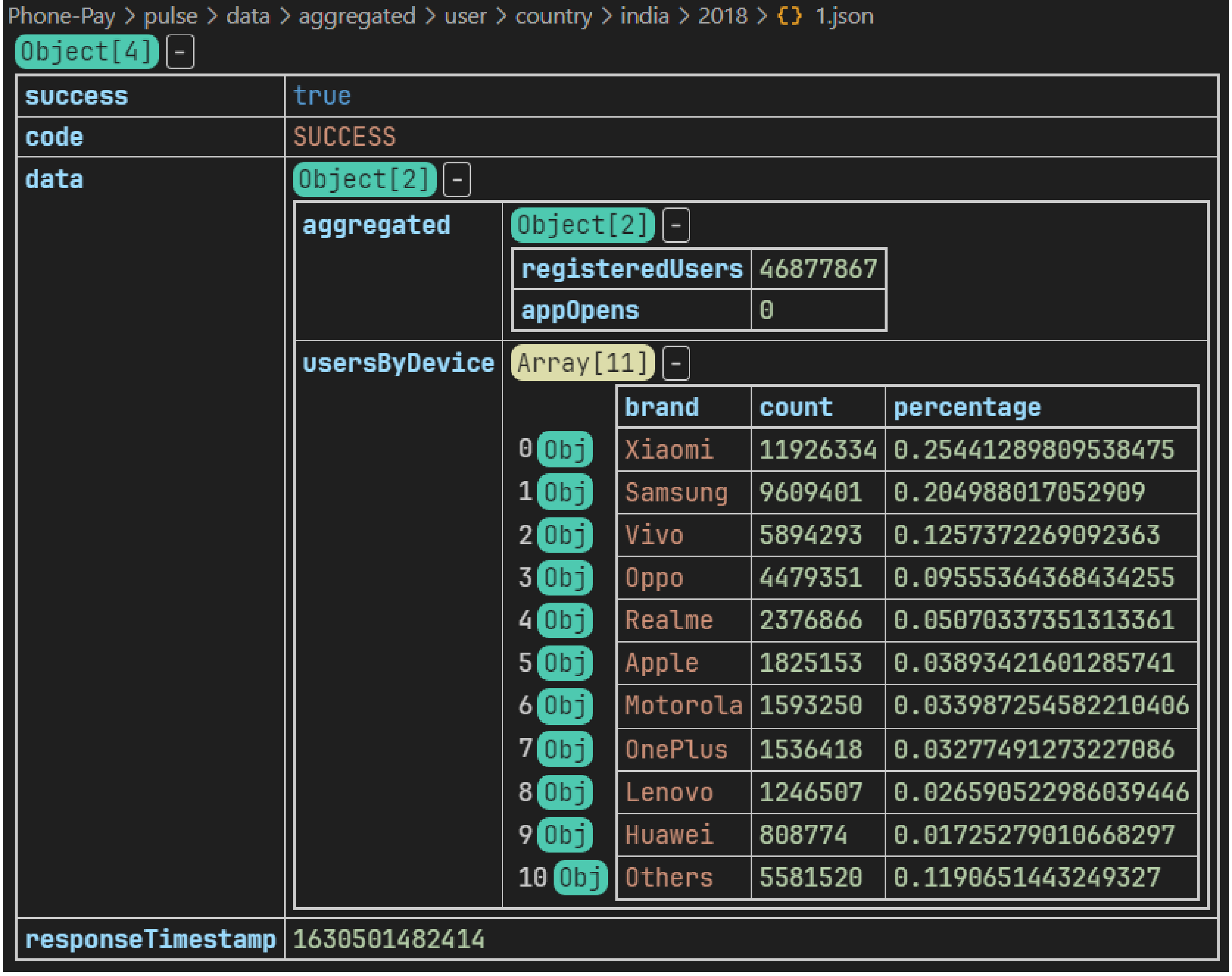
transactionData:

	name	paymentInstruments
0 Obj	Recharge & bill payments	Array[1]
1 Obj	Peer-to-peer payments	Array[1]
2 Obj	Merchant payments	Array[1]
3 Obj	Financial Services	Array[1]
4 Obj	Others	Array[1]

paymentInstruments:

		type	count	amount
Recharge & bill payments	0 Obj	TOTAL	4209319	831912767.5937189
Peer-to-peer payments	0 Obj	TOTAL	3307245	11748268950.760422
Merchant payments	0 Obj	TOTAL	351474	282899066.6137372
Financial Services	0 Obj	TOTAL	262876	57011875.36233704
Others	0 Obj	TOTAL	391852	388111458.2121352

- **File Data In user → country → India → (Year Folders [2018, 2019, 2020, 2021, 2022, 2023, 2024])**


Phone-Pay > pulse > data > aggregated > user > country > india > 2018 > {} 1.json

Object[4]

success	true
code	SUCCESS
data	Object[2]
responseTimestamp	1630501482414

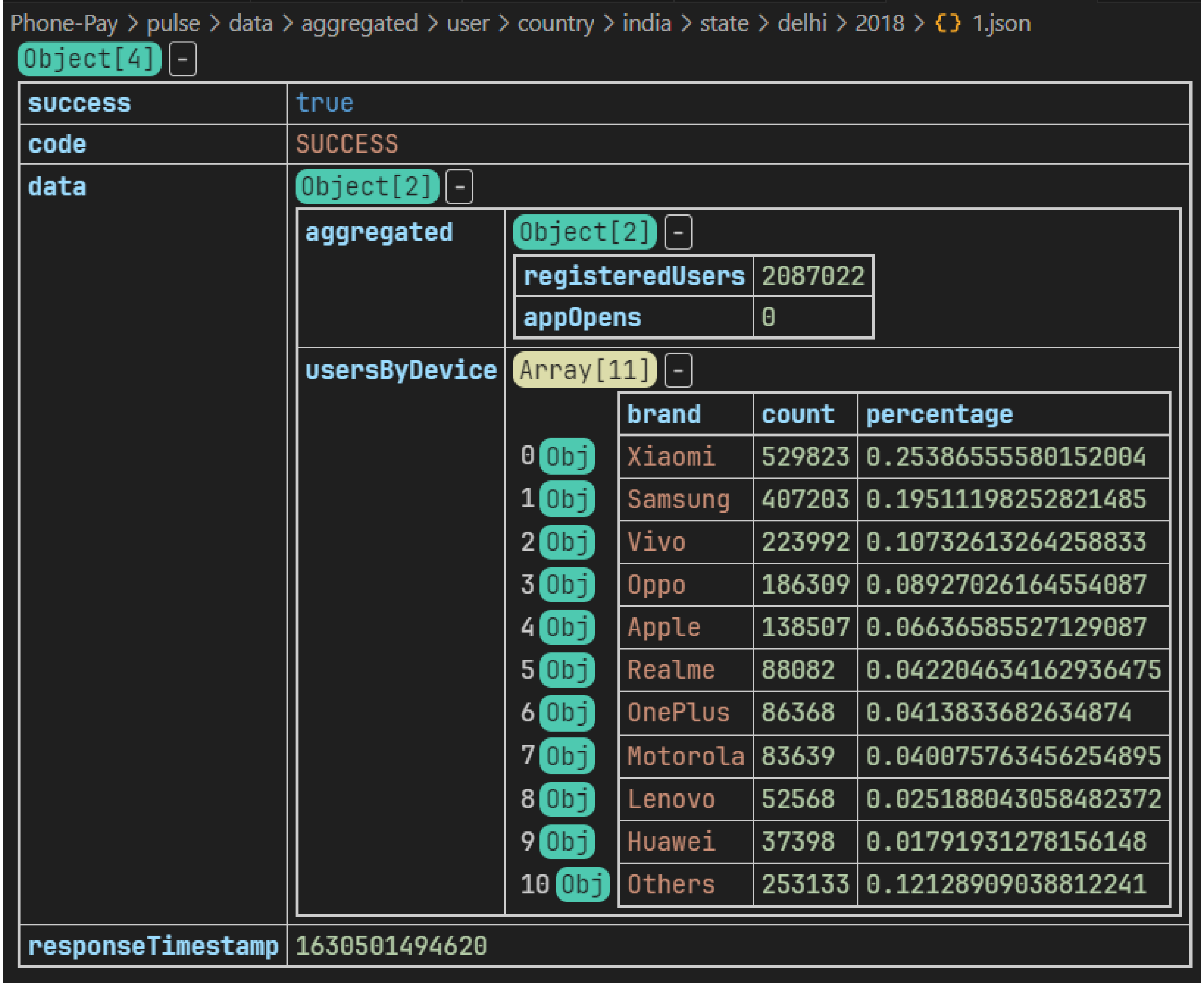
data → aggregated: Object[2]

registeredUsers	46877867
appOpens	0

data → usersByDevice: Array[11]

	brand	count	percentage
0 Obj	Xiaomi	11926334	0.25441289809538475
1 Obj	Samsung	9609401	0.204988017052909
2 Obj	Vivo	5894293	0.1257372269092363
3 Obj	Oppo	4479351	0.09555364368434255
4 Obj	Realme	2376866	0.05070337351313361
5 Obj	Apple	1825153	0.03893421601285741
6 Obj	Motorola	1593250	0.033987254582210406
7 Obj	OnePlus	1536418	0.03277491273227086
8 Obj	Lenovo	1246507	0.026590522986039446
9 Obj	Huawei	808774	0.01725279010668297
10 Obj	Others	5581520	0.1190651443249327

- **File Data In user → country → India → (State [Andhra, Telangana, Delhi, ...])


Phone-Pay > pulse > data > aggregated > user > country > india > state > delhi > 2018 > {} 1.json

Object[4] -

success	true
code	SUCCESS
data	Object[2] -
responseTimestamp	1630501494620

data:

aggregated	Object[2] -
usersByDevice	Array[11] -

aggregated:

registeredUsers	2087022
appOpens	0

usersByDevice:

	brand	count	percentage
0 Obj	Xiaomi	529823	0.25386555580152004
1 Obj	Samsung	407203	0.19511198252821485
2 Obj	Vivo	223992	0.10732613264258833
3 Obj	Oppo	186309	0.08927026164554087
4 Obj	Apple	138507	0.06636585527129087
5 Obj	Realme	88082	0.042204634162936475
6 Obj	OnePlus	86368	0.0413833682634874
7 Obj	Motorola	83639	0.040075763456254895
8 Obj	Lenovo	52568	0.025188043058482372
9 Obj	Huawei	37398	0.01791931278156148
10 Obj	Others	253133	0.12128909038812241

map

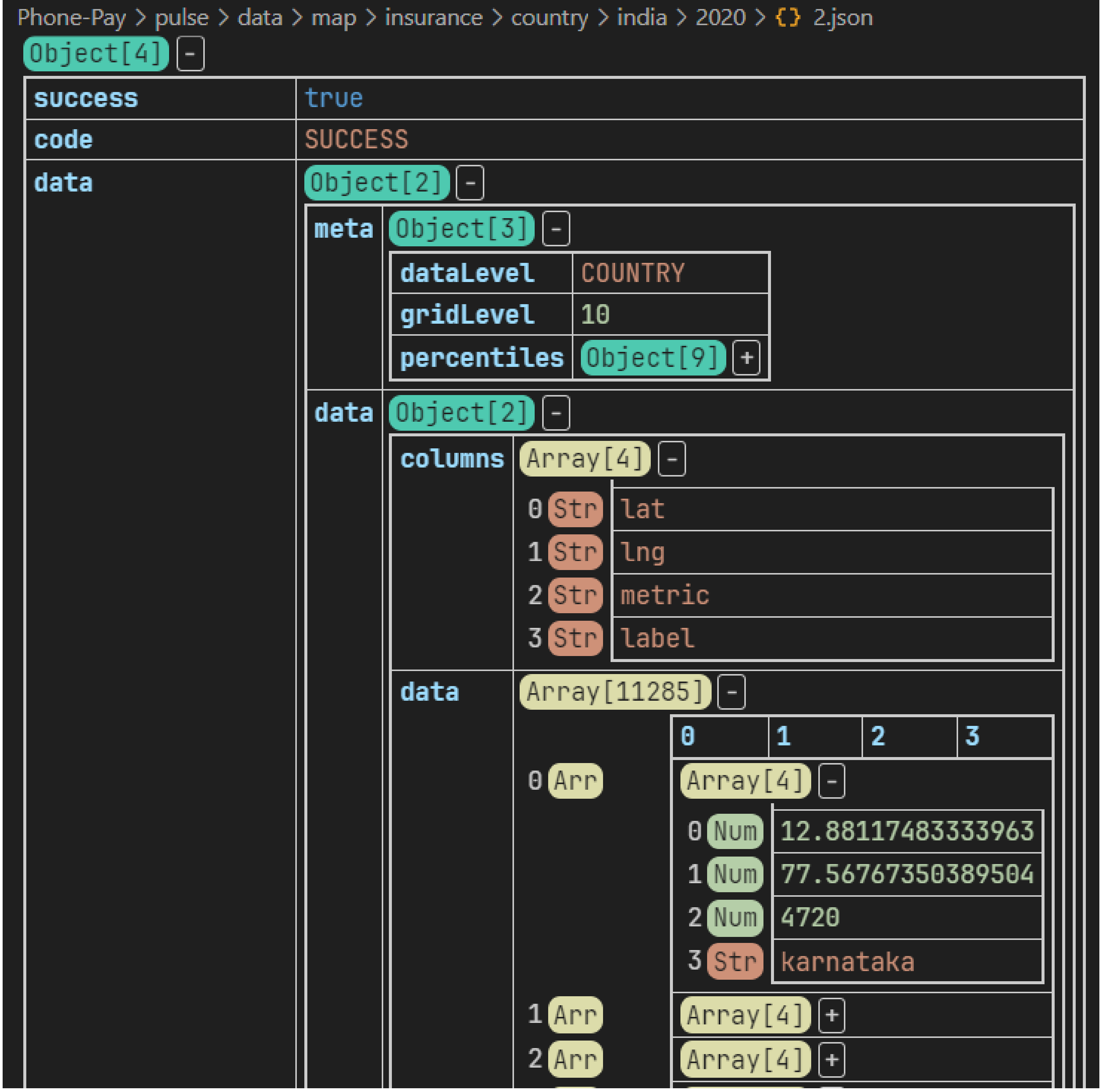
Files in map :

```
===============================================================================
 Language            Files        Lines         Code     Comments       Blanks
===============================================================================
 JSON                 3034         3034         3034            0            0
===============================================================================
 Total                3034         3034         3034            0            0
===============================================================================
```

```
.
├── insurance
│   ├── country
│   │   └── india
│   │       ├── 2020
│   │       ├── 2021
│   │       ├── 2022
│   │       ├── 2023
│   │       ├── 2024
│   │       └── state
│   └── hover
│
```

```
│       └── country
│           └── india
│               ├── 2020
│               ├── 2021
│               ├── 2022
│               ├── 2023
│               ├── 2024
│               └── state
├── transaction
│   └── hover
│       └── country
│           └── india
│               ├── 2018
│               ├── 2019
│               ├── 2020
│               ├── 2021
│               ├── 2022
│               ├── 2023
│               ├── 2024
│               └── state
└── user
    └── hover
        └── country
            └── india
                ├── 2018
                ├── 2019
                ├── 2020
                ├── 2021
                ├── 2022
                ├── 2023
                ├── 2024
                └── state
```

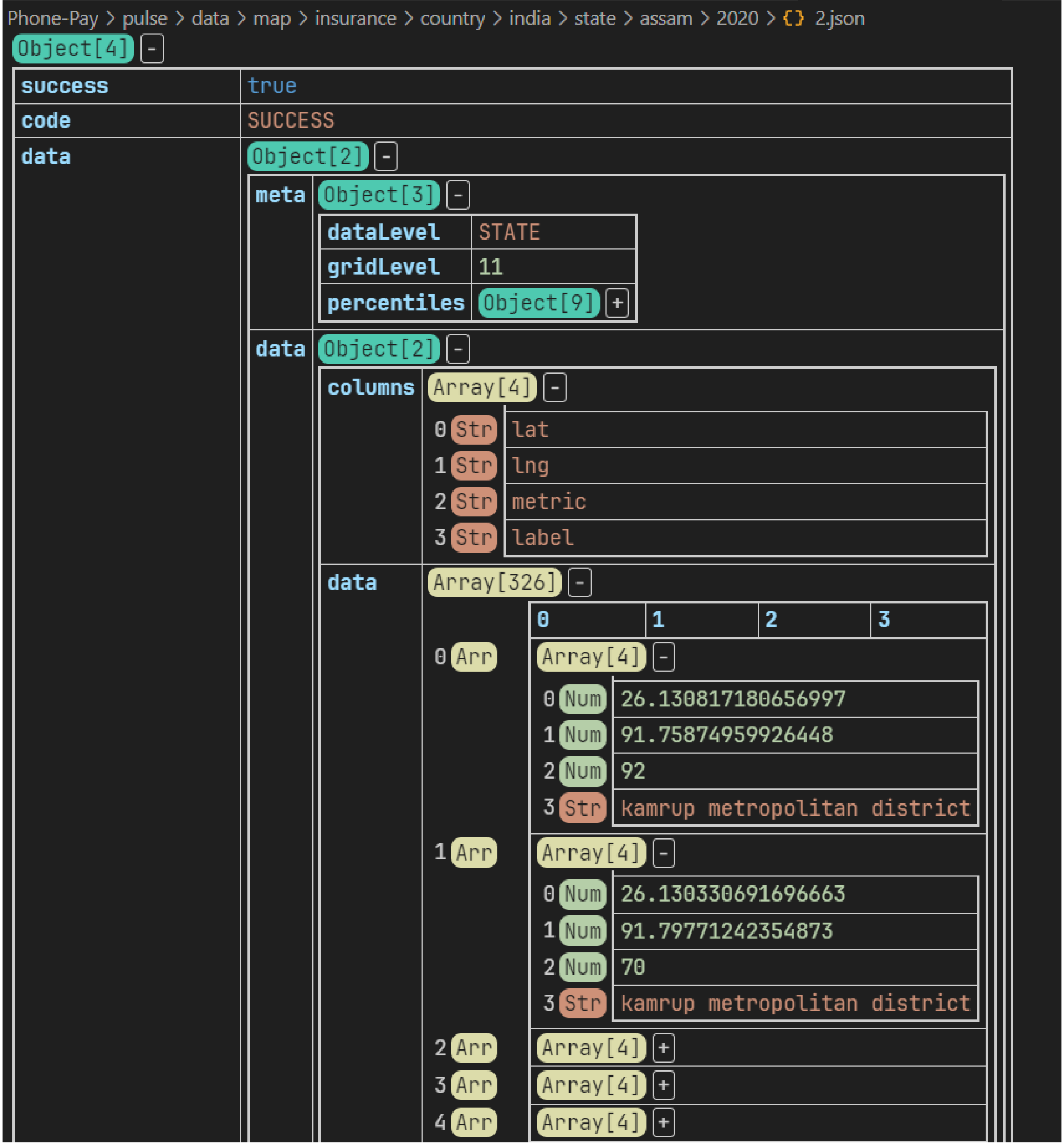
- **File Data In insurance → country → India → (Year Folders [2018, 2019, 2020, 2021, 2022, 2023, 2024])**



Phone-Pay > pulse > data > map > insurance > country > india > 2020 > {} 2.json

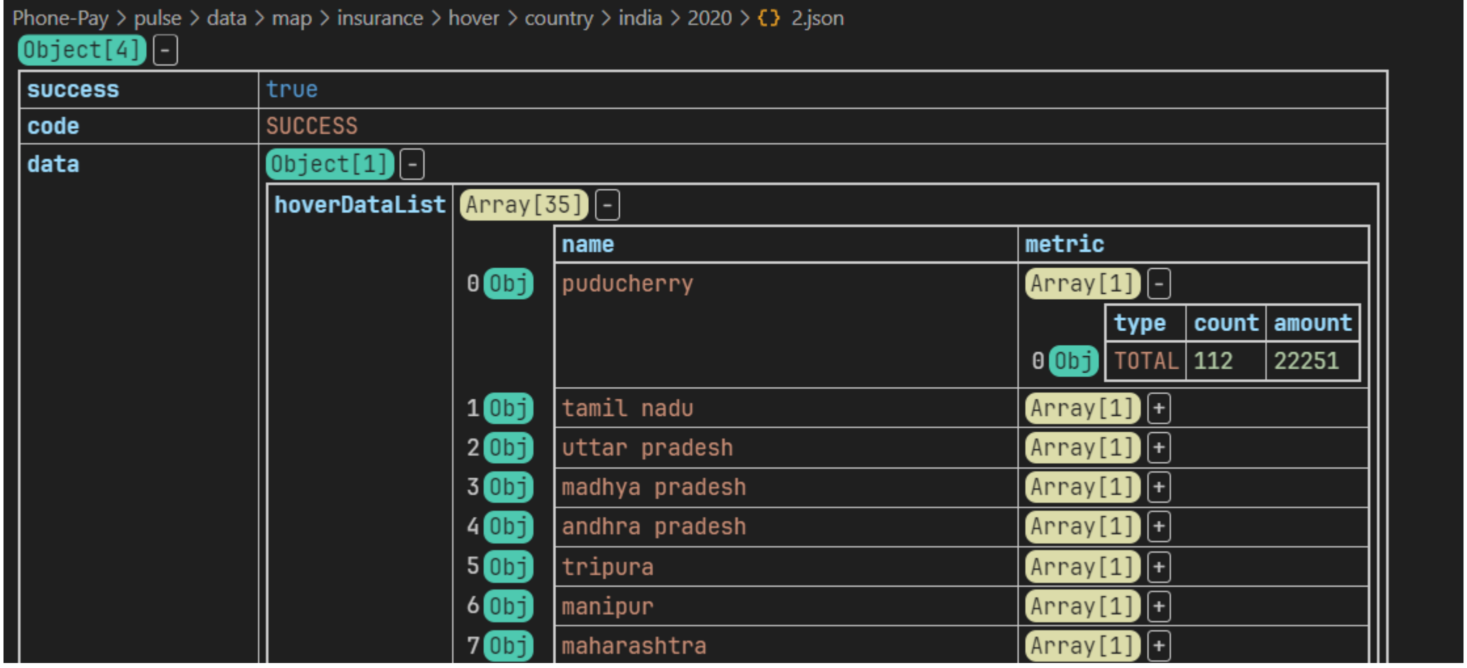
```
Object[4] -
success     true
code        SUCCESS
data        Object[2] -
            meta  Object[3] -
                  dataLevel    COUNTRY
                  gridLevel    10
                  percentiles  Object[9] +
            data  Object[2] -
                  columns  Array[4] -
                           0 Str  lat
                           1 Str  lng
                           2 Str  metric
                           3 Str  label
                  data     Array[11285] -
                                     0        1        2        3
                           0 Arr     Array[4] -
                                     0 Num  12.88117483333963
                                     1 Num  77.56767350389504
                                     2 Num  4720
                                     3 Str  karnataka
                           1 Arr     Array[4] +
                           2 Arr     Array[4] +
```

- **File Data In insurance → country → India → (State [Andhra, Telangana, Delhi, ...])



```
Phone-Pay > pulse > data > map > insurance > country > india > state > assam > 2020 > {} 2.json
Object[4] -
success   true
code      SUCCESS
data      Object[2] -
          meta  Object[3] -
                dataLevel    STATE
                gridLevel    11
                percentiles  Object[9] +
          data  Object[2] -
                columns  Array[4] -
                         0 Str  lat
                         1 Str  lng
                         2 Str  metric
                         3 Str  label
                data     Array[326] -
                                  0          1          2          3
                         0 Arr    Array[4] -
                                  0 Num  26.130817180656997
                                  1 Num  91.75874959926448
                                  2 Num  92
                                  3 Str  kamrup metropolitan district
                         1 Arr    Array[4] -
                                  0 Num  26.130330691696663
                                  1 Num  91.79771242354873
                                  2 Num  70
                                  3 Str  kamrup metropolitan district
                         2 Arr    Array[4] +
                         3 Arr    Array[4] +
                         4 Arr    Array[4] +
```

- **File Data In insurance → hover → country → India → (Year Folders [2018, 2019, 2020, 2021, 2022, 2023, 2024])**


Phone-Pay > pulse > data > map > insurance > hover > country > india > 2020 > {} 2.json

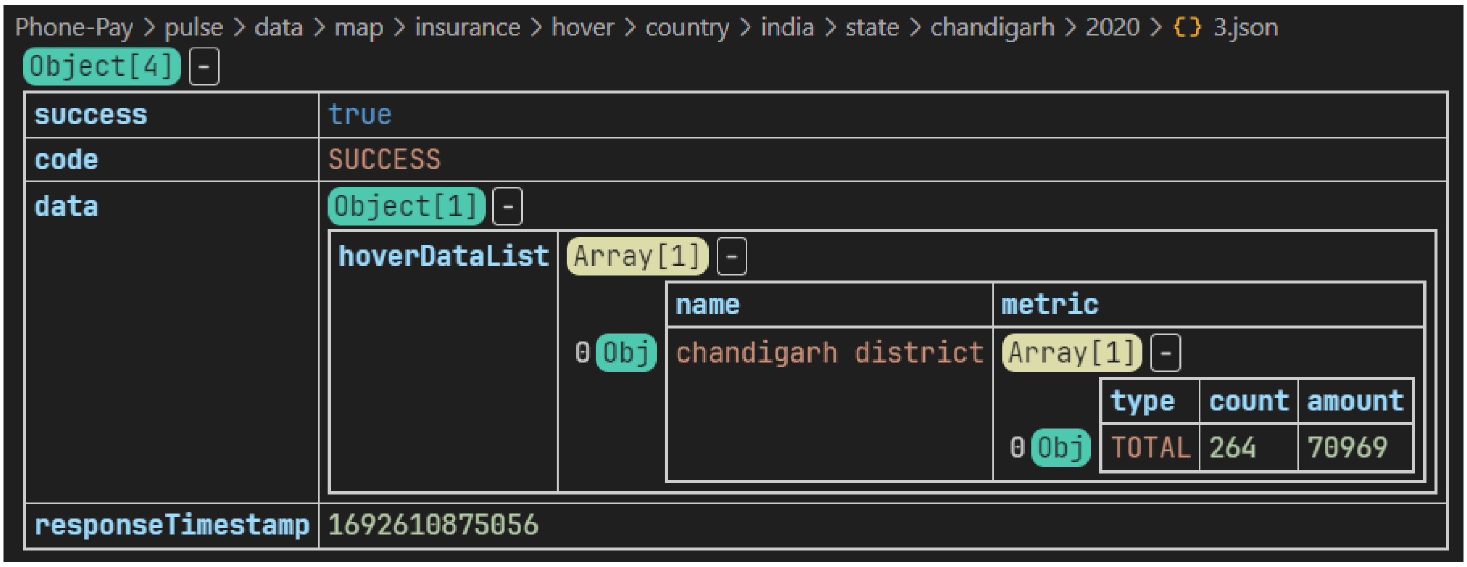
Object[4] -

success	true
code	SUCCESS
data	Object[1] -

hoverDataList Array[35] -

	name	metric
0 Obj	puducherry	Array[1] - ; 0 Obj: type TOTAL, count 112, amount 22251
1 Obj	tamil nadu	Array[1] +
2 Obj	uttar pradesh	Array[1] +
3 Obj	madhya pradesh	Array[1] +
4 Obj	andhra pradesh	Array[1] +
5 Obj	tripura	Array[1] +
6 Obj	manipur	Array[1] +
7 Obj	maharashtra	Array[1] +

- - **File Data In insurance → hover → country → India → (State [Andhra, Telangana, Delhi, ...])


Phone-Pay > pulse > data > map > insurance > hover > country > india > state > chandigarh > 2020 > {} 3.json

Object[4] -

success	true
code	SUCCESS
data	Object[1] -
responseTimestamp	1692610875056

hoverDataList Array[1] -

	name	metric
0 Obj	chandigarh district	Array[1] - ; 0 Obj: type TOTAL, count 264, amount 70969

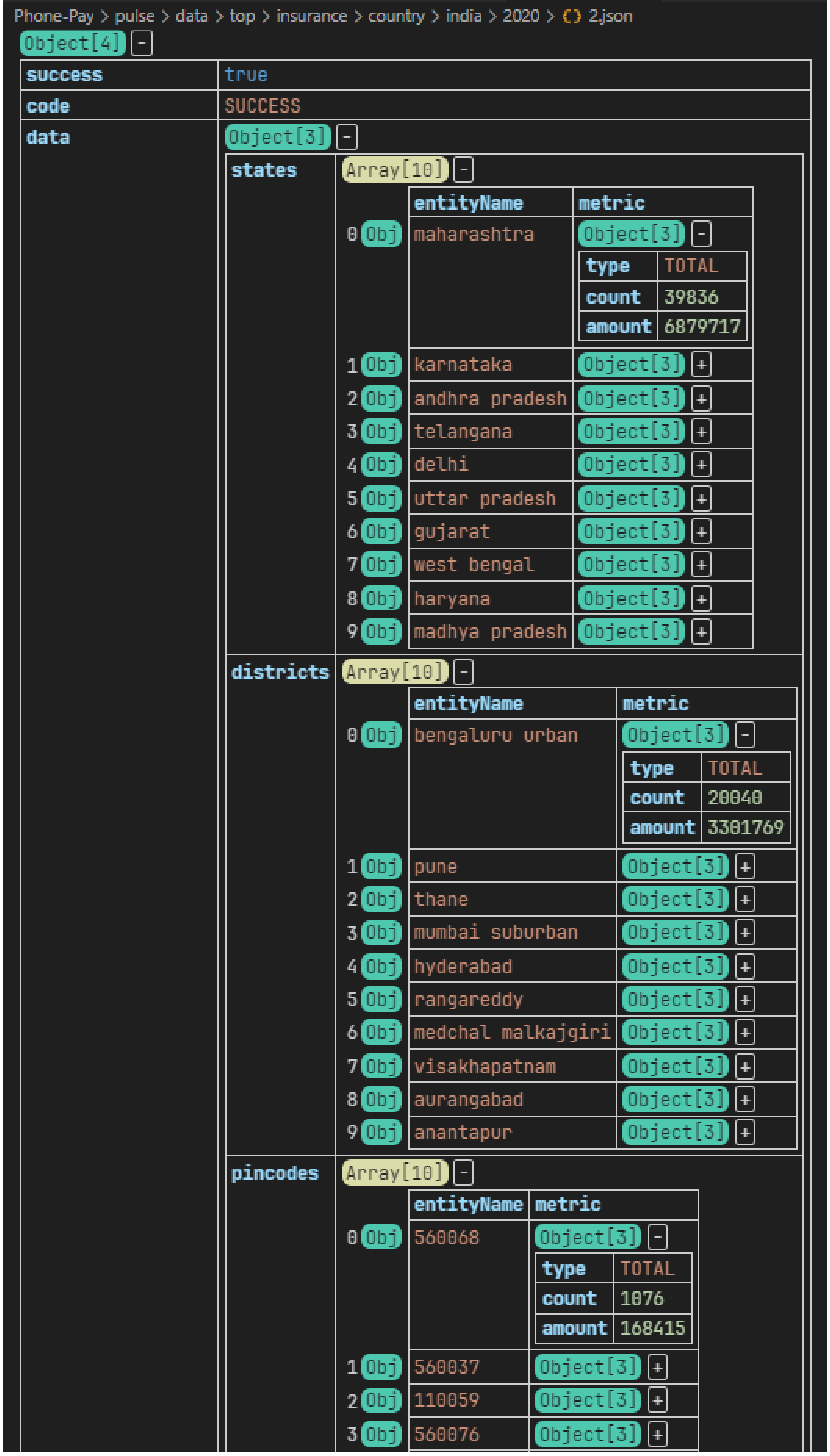
top

Files in top :

```
===============================================================================
 Language            Files        Lines         Code     Comments       Blanks
===============================================================================
 JSON                 2442         2675         2672            0            3
===============================================================================
 Total                2442         2675         2672            0            3
===============================================================================
```

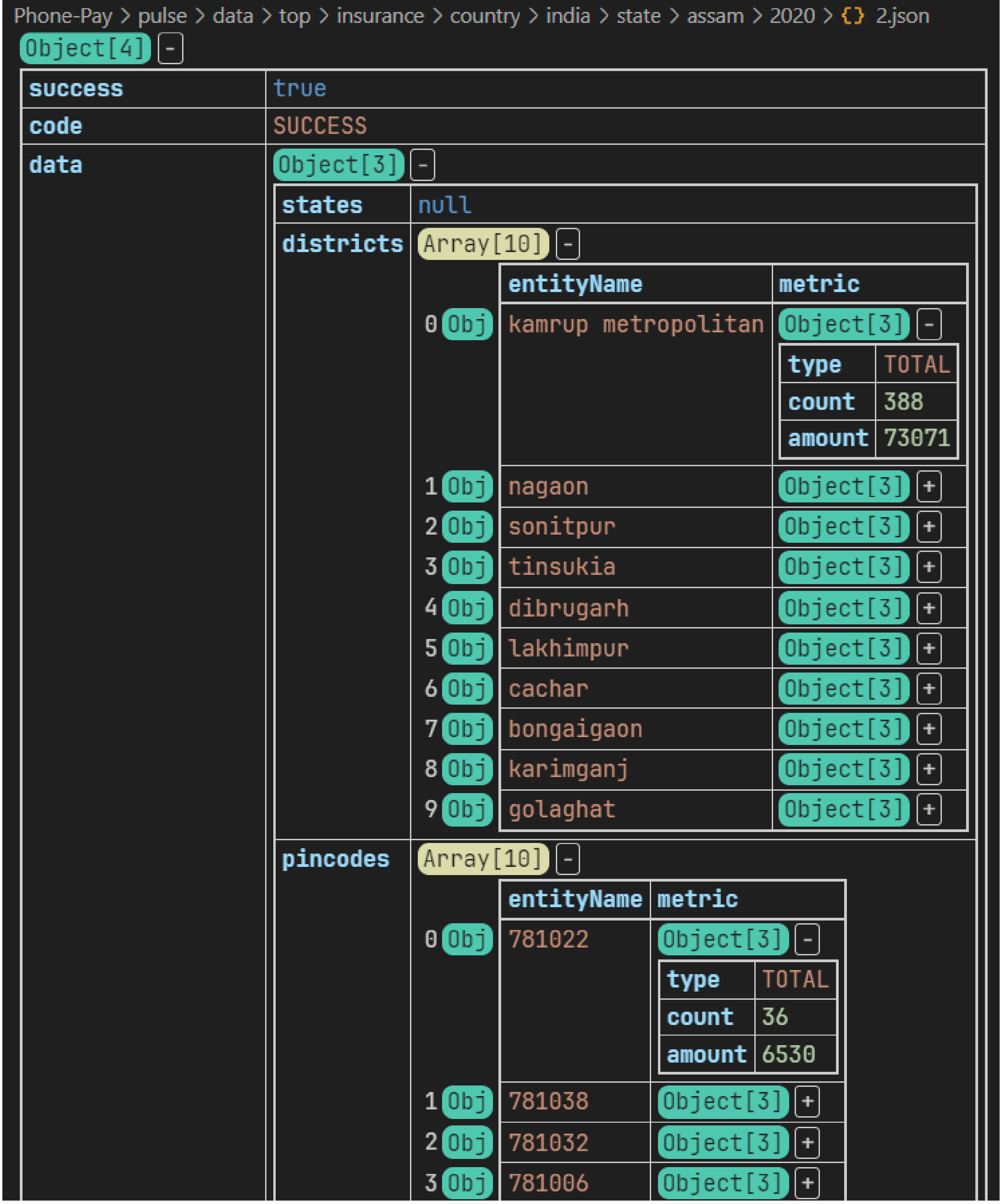
```
.
├── insurance
│   └── country
│       └── india
│           ├── 2020
│           ├── 2021
│           ├── 2022
│           │
│                 ├── 2023
│                 ├── 2024
│                 └── state
├── transaction
│   └── country
│       └── india
│           ├── 2018
│           ├── 2019
│           ├── 2020
│           ├── 2021
│           ├── 2022
│           ├── 2023
│           ├── 2024
│           └── state
└── user
    └── country
        └── india
            ├── 2018
            ├── 2019
            ├── 2020
            ├── 2021
            ├── 2022
            ├── 2023
            ├── 2024
            └── state
```

- **File Data In insurance → country → India → (Year Folders [2018, 2019, 2020, 2021, 2022, 2023, 2024])**



```
Phone-Pay > pulse > data > top > insurance > country > india > 2020 > {} 2.json
Object[4] -
success   true
code      SUCCESS
data      Object[3] -
          states     Array[10] -
                         entityName        metric
                     0 Obj maharashtra     Object[3] -
                                           type    TOTAL
                                           count   39836
                                           amount  6879717
                     1 Obj karnataka       Object[3] +
                     2 Obj andhra pradesh  Object[3] +
                     3 Obj telangana       Object[3] +
                     4 Obj delhi           Object[3] +
                     5 Obj uttar pradesh   Object[3] +
                     6 Obj gujarat         Object[3] +
                     7 Obj west bengal     Object[3] +
                     8 Obj haryana         Object[3] +
                     9 Obj madhya pradesh  Object[3] +
          districts  Array[10] -
                         entityName          metric
                     0 Obj bengaluru urban   Object[3] -
                                             type    TOTAL
                                             count   20040
                                             amount  3301769
                     1 Obj pune              Object[3] +
                     2 Obj thane             Object[3] +
                     3 Obj mumbai suburban   Object[3] +
                     4 Obj hyderabad         Object[3] +
                     5 Obj rangareddy        Object[3] +
                     6 Obj medchal malkajgiri Object[3] +
                     7 Obj visakhapatnam     Object[3] +
                     8 Obj aurangabad        Object[3] +
                     9 Obj anantapur         Object[3] +
          pincodes   Array[10] -
                         entityName metric
                     0 Obj 560068    Object[3] -
                                     type    TOTAL
                                     count   1076
                                     amount  168415
                     1 Obj 560037    Object[3] +
                     2 Obj 110059    Object[3] +
                     3 Obj 560076    Object[3] +
```

- **File Data In insurance → country → India → (State [Andhra, Telangana, Delhi, ...])**


Phone-Pay > pulse > data > top > insurance > country > india > state > assam > 2020 > {} 2.json

Object[4] -

success	true
code	SUCCESS
data	Object[3] -

data:

states	null
districts	Array[10] -

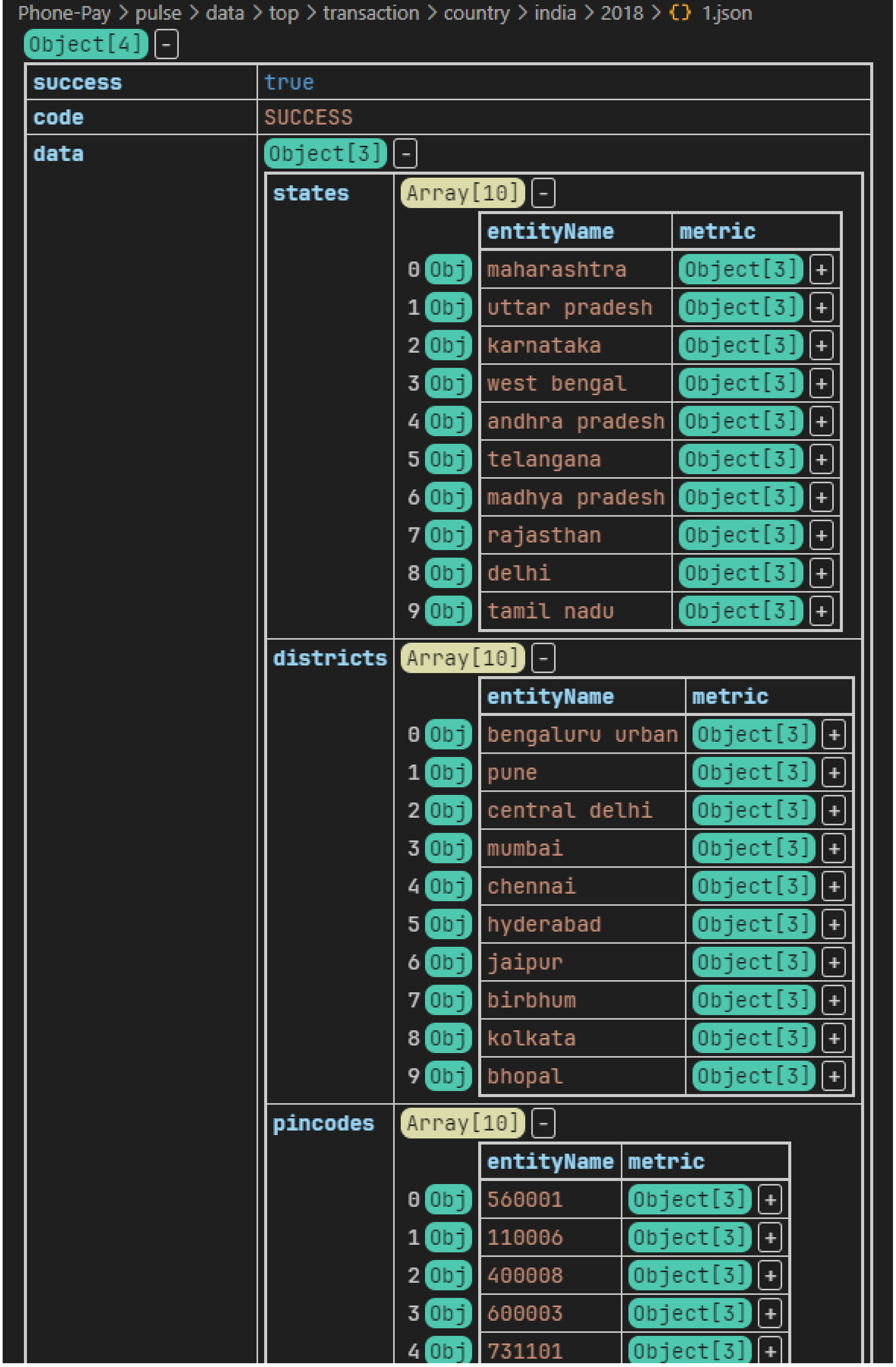
districts:

	entityName	metric
0 Obj	kamrup metropolitan	Object[3] - type TOTAL, count 388, amount 73071
1 Obj	nagaon	Object[3] +
2 Obj	sonitpur	Object[3] +
3 Obj	tinsukia	Object[3] +
4 Obj	dibrugarh	Object[3] +
5 Obj	lakhimpur	Object[3] +
6 Obj	cachar	Object[3] +
7 Obj	bongaigaon	Object[3] +
8 Obj	karimganj	Object[3] +
9 Obj	golaghat	Object[3] +

pincodes Array[10] -

	entityName	metric
0 Obj	781022	Object[3] - type TOTAL, count 36, amount 6530
1 Obj	781038	Object[3] +
2 Obj	781032	Object[3] +
3 Obj	781006	Object[3] +

- File Data In transaction → country → India → (Year Folders [2018, 2019, 2020, 2021, 2022, 2023, 2024])



Phone-Pay > pulse > data > top > transaction > country > india > 2018 > {} 1.json

Object[4] -

success	true
code	SUCCESS
data	Object[3] -

states Array[10] -

		entityName	metric
0	Obj	maharashtra	Object[3] +
1	Obj	uttar pradesh	Object[3] +
2	Obj	karnataka	Object[3] +
3	Obj	west bengal	Object[3] +
4	Obj	andhra pradesh	Object[3] +
5	Obj	telangana	Object[3] +
6	Obj	madhya pradesh	Object[3] +
7	Obj	rajasthan	Object[3] +
8	Obj	delhi	Object[3] +
9	Obj	tamil nadu	Object[3] +

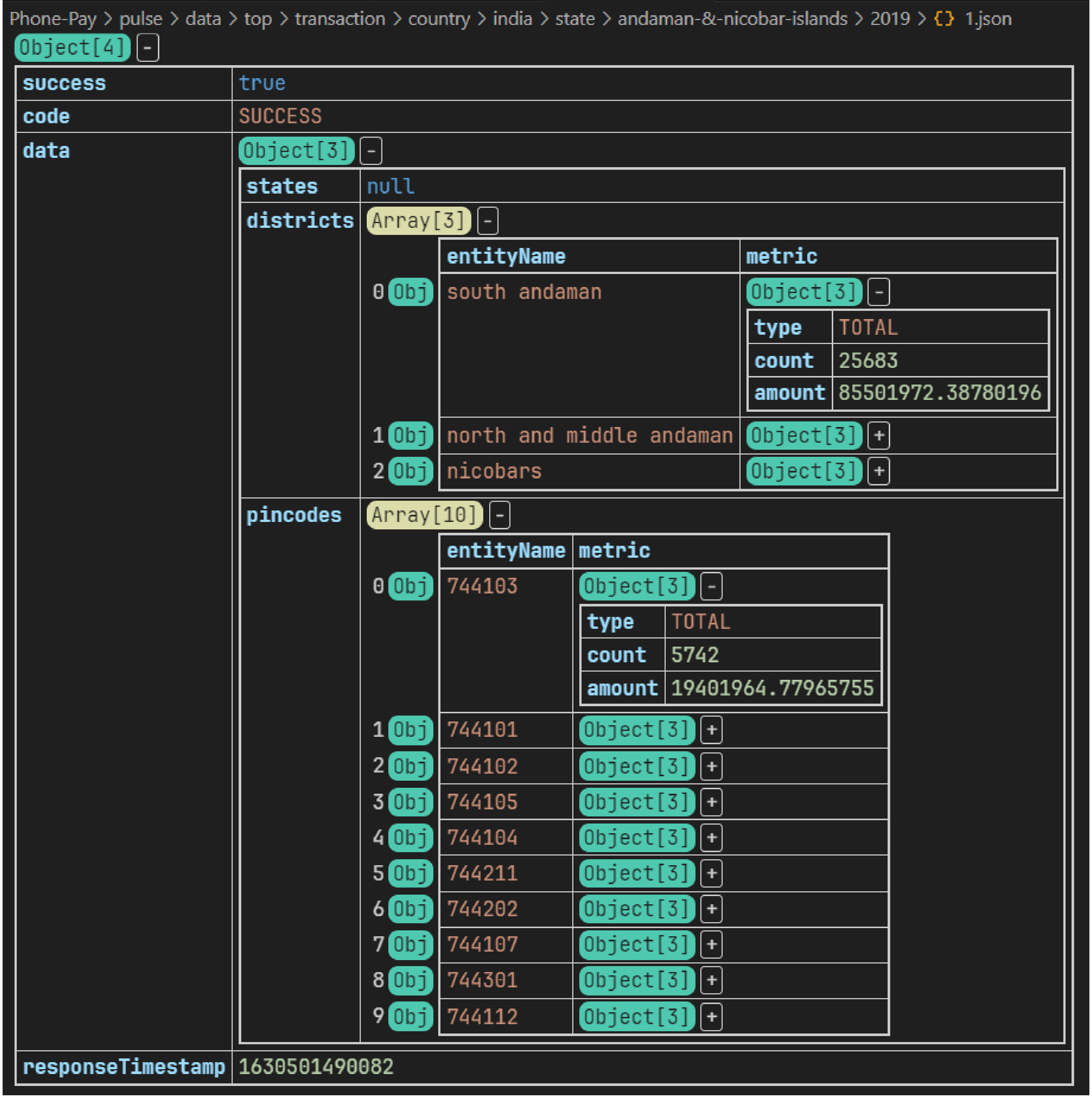
districts Array[10] -

		entityName	metric
0	Obj	bengaluru urban	Object[3] +
1	Obj	pune	Object[3] +
2	Obj	central delhi	Object[3] +
3	Obj	mumbai	Object[3] +
4	Obj	chennai	Object[3] +
5	Obj	hyderabad	Object[3] +
6	Obj	jaipur	Object[3] +
7	Obj	birbhum	Object[3] +
8	Obj	kolkata	Object[3] +
9	Obj	bhopal	Object[3] +

pincodes Array[10] -

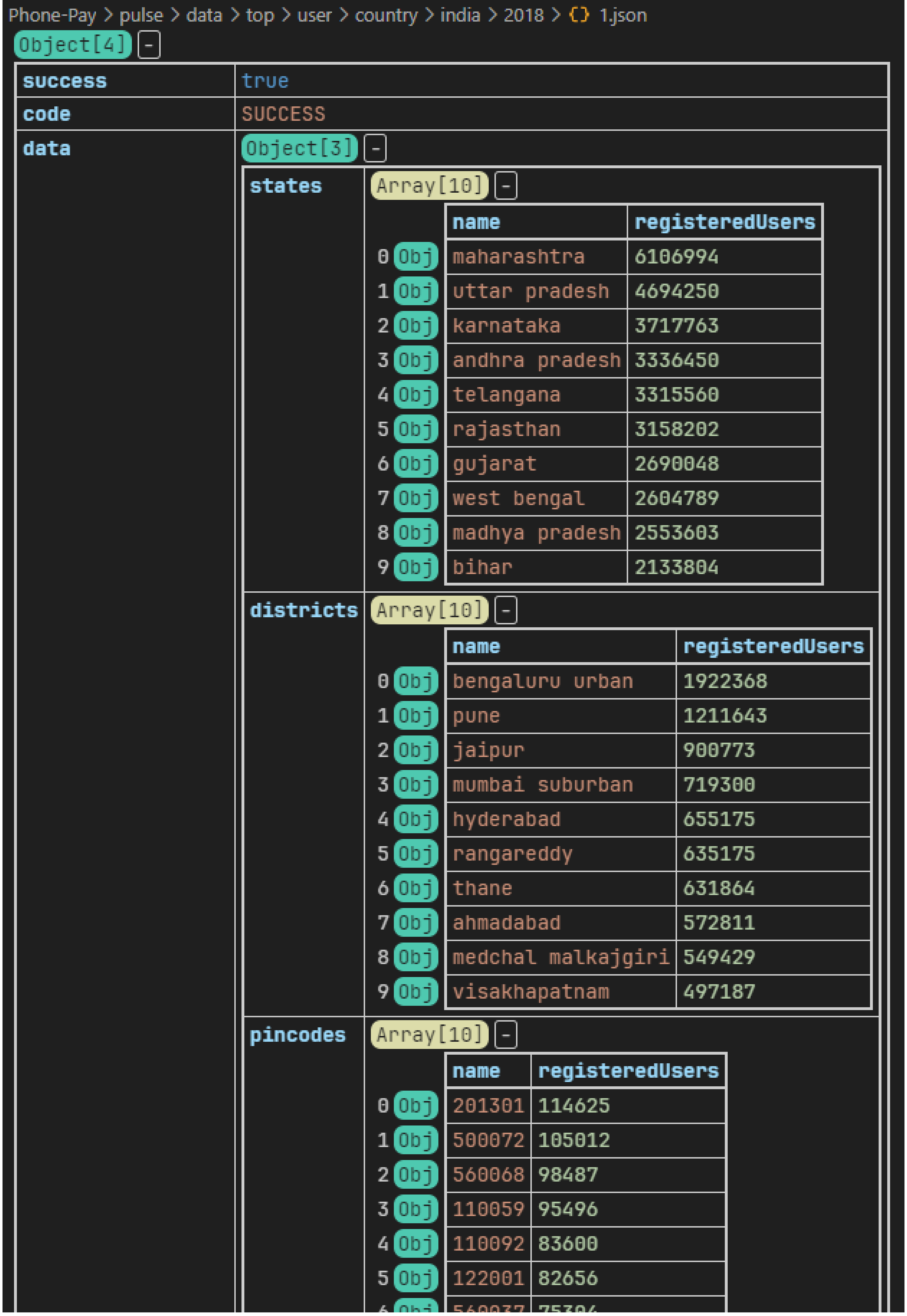
		entityName	metric
0	Obj	560001	Object[3] +
1	Obj	110006	Object[3] +
2	Obj	400008	Object[3] +
3	Obj	600003	Object[3] +
4	Obj	731101	Object[3] +

- **File Data In transaction → country → India → (State [Andhra, Telangana, Delhi, ...])



```
Phone-Pay > pulse > data > top > transaction > country > india > state > andaman-&-nicobar-islands > 2019 > {} 1.json
Object[4] -
success            true
code               SUCCESS
data               Object[3] -
                   states     null
                   districts  Array[3] -
                                     entityName                 metric
                              0 Obj  south andaman              Object[3] -
                                                                type    TOTAL
                                                                count   25683
                                                                amount  85501972.38780196
                              1 Obj  north and middle andaman   Object[3] +
                              2 Obj  nicobars                   Object[3] +
                   pincodes   Array[10] -
                                     entityName  metric
                              0 Obj  744103      Object[3] -
                                                 type    TOTAL
                                                 count   5742
                                                 amount  19401964.77965755
                              1 Obj  744101      Object[3] +
                              2 Obj  744102      Object[3] +
                              3 Obj  744105      Object[3] +
                              4 Obj  744104      Object[3] +
                              5 Obj  744211      Object[3] +
                              6 Obj  744202      Object[3] +
                              7 Obj  744107      Object[3] +
                              8 Obj  744301      Object[3] +
                              9 Obj  744112      Object[3] +
responseTimestamp  1630501490082
```

- **File Data In user → country → India → (Year Folders [2018, 2019, 2020, 2021, 2022, 2023, 2024])**



Phone-Pay > pulse > data > top > user > country > india > 2018 > {} 1.json

Object[4]

success	true
code	SUCCESS
data	Object[3]

states Array[10]

	name	registeredUsers
0	maharashtra	6106994
1	uttar pradesh	4694250
2	karnataka	3717763
3	andhra pradesh	3336450
4	telangana	3315560
5	rajasthan	3158202
6	gujarat	2690048
7	west bengal	2604789
8	madhya pradesh	2553603
9	bihar	2133804

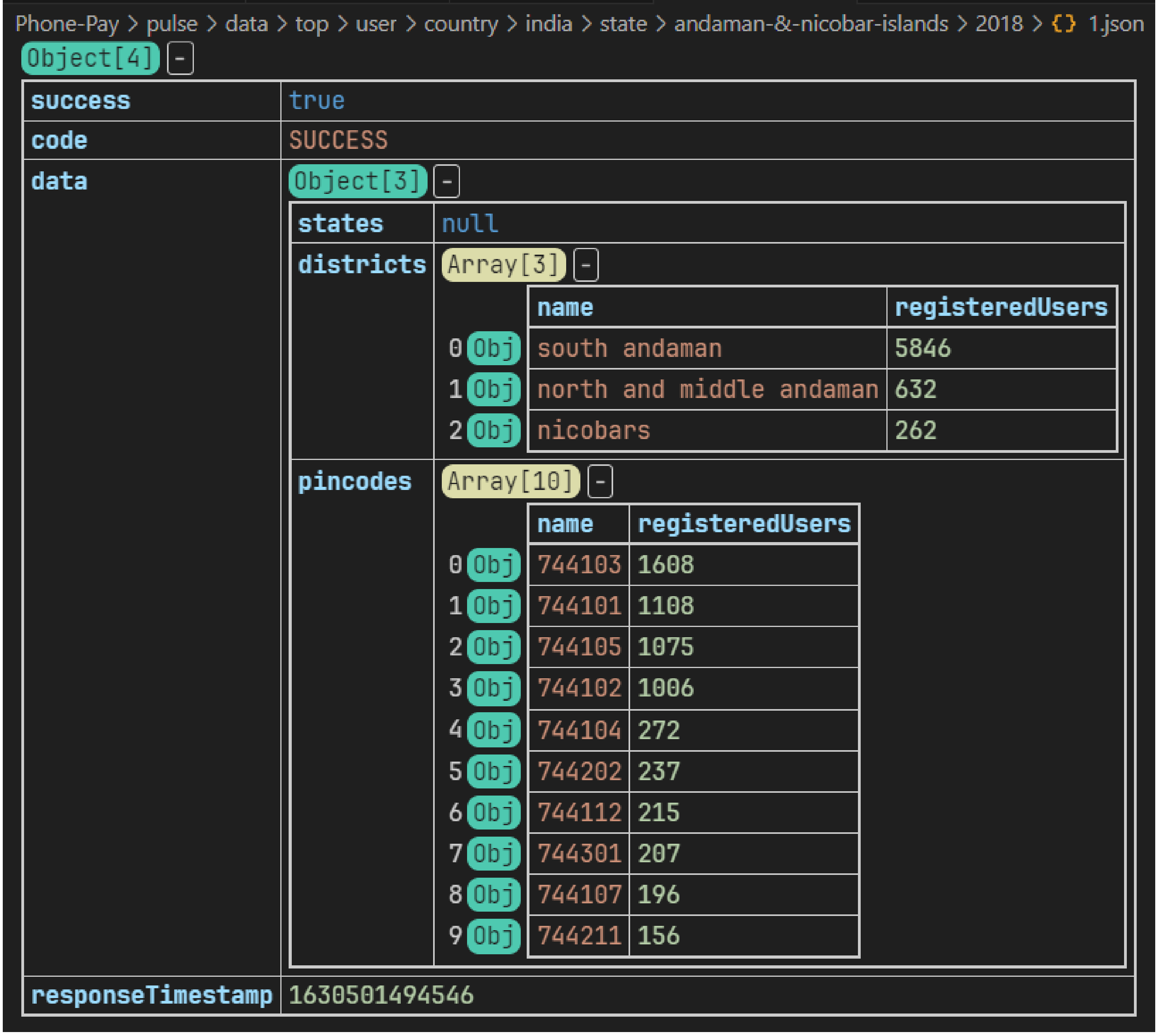
districts Array[10]

	name	registeredUsers
0	bengaluru urban	1922368
1	pune	1211643
2	jaipur	900773
3	mumbai suburban	719300
4	hyderabad	655175
5	rangareddy	635175
6	thane	631864
7	ahmadabad	572811
8	medchal malkajgiri	549429
9	visakhapatnam	497187

pincodes Array[10]

	name	registeredUsers
0	201301	114625
1	500072	105012
2	560068	98487
3	110059	95496
4	110092	83600
5	122001	82656

- **File Data In user → country → India → (State [Andhra, Telangana, Delhi, ...])


Phone-Pay > pulse > data > top > user > country > india > state > andaman-&-nicobar-islands > 2018 > 1.json

Object[4]

success	true
code	SUCCESS
data	Object[3]
responseTimestamp	1630501494546

data: states null

districts Array[3]

	name	registeredUsers
0	south andaman	5846
1	north and middle andaman	632
2	nicobars	262

pincodes Array[10]

	name	registeredUsers
0	744103	1608
1	744101	1108
2	744105	1075
3	744102	1006
4	744104	272
5	744202	237
6	744112	215
7	744301	207
8	744107	196
9	744211	156

Some Suggestion

Some of the tools that could be help full to view the JSON files:

1. JSON GRID VsCode extension
2. JSON CRACK VsCode extension

```
Dataview (inline field
'=========================================================================
=======
 Language                    Files           Lines            Code       Comments
Blanks
=========================================================================
=======
 JSON                         7918            8278            8275              0
```

```
3
--------------------------------------------------------------------------
-------
 Markdown                1          188            0          135
53
 |- JavaScript           1          292          268            0
24
 (Total)                            480          268          135
77
==========================================================================
=======
 Total                7919         8466         8275          135
56
==========================================================================
======='): Error:
-- PARSING FAILED ------------------------------------------------

> 1 |
==========================================================================
======
    | ^
  2 |  Language              Files         Lines          Code     Comments
Blanks
  3 |
==========================================================================
=======

Expected one of the following:

'(', 'null', boolean, date, duration, file link, list ('[1, 2, 3]'),
negated field, number, object ('{ a: 1, b: 2 }'), string, variable
```

```
Dataview (inline field
'=========================================================================
=======
 Language              Files         Lines          Code     Comments
Blanks
==========================================================================
=======
 JSON                  2442          2569          2569            0
0
==========================================================================
=======
 Total                 2442          2569          2569            0
0
==========================================================================
=======') : Error:
-- PARSING FAILED --------------------------------------------------

> 1 |
==========================================================================
======
    | ^
  2 |  Language              Files        Lines         Code     Comments
Blanks
  3 |
==========================================================================
=======

Expected one of the following:

'(', 'null', boolean, date, duration, file link, list ('[1, 2, 3]'),
negated field, number, object ('{ a: 1, b: 2 }'), string, variable
```

```
Dataview (inline field
'=========================================================================
=======
 Language              Files        Lines         Code     Comments
Blanks
==========================================================================
=======
 JSON                   3034         3034         3034            0
0
==========================================================================
=======
 Total                  3034         3034         3034            0
0
==========================================================================
=======') : Error:
-- PARSING FAILED --------------------------------------------------

> 1 |
==========================================================================
======
    | ^
  2 |  Language              Files        Lines         Code     Comments
Blanks
  3 |
==========================================================================
=======

Expected one of the following:
```

```
'(', 'null', boolean, date, duration, file link, list ('[1, 2, 3]'),
negated field, number, object ('{ a: 1, b: 2 }'), string, variable
```

```
Dataview (inline field
'======================================================================
=======
 Language              Files        Lines         Code     Comments
Blanks
======================================================================
=======
 JSON                   2442         2675         2672            0
3
======================================================================
=======
 Total                  2442         2675         2672            0
3
======================================================================
======='): Error:
-- PARSING FAILED --------------------------------------------------

> 1 |
======================================================================
======
    | ^
  2 |  Language              Files        Lines         Code     Comments
Blanks
  3 |
======================================================================
=======

Expected one of the following:

'(', 'null', boolean, date, duration, file link, list ('[1, 2, 3]'),
negated field, number, object ('{ a: 1, b: 2 }'), string, variable
```