

com.dropoff.service.brawndo.client.java

This is the 3rd party dropoff Java client for creating and viewing orders.

- For Javascript documentation go [HERE](#)
- For PHP documentation go [HERE](#)
- For GO documentation go [HERE](#)
- For Ruby documentation go [HERE](#)
- For .NET documentation go [HERE](#)
- For Python documentation go [HERE](#)

Table of Contents

- [Client Info](#)
 - [Configuration](#)
 - [Getting Your Account Info](#)
 - [Enterprise Managed Clients](#)
 - [Getting Pricing Estimates](#)
 - [Placing an Order](#)
 - [Cancelling an Order](#)
 - [Getting a Specific Order](#)
 - [Getting a Page of Order](#)
- [Tips](#)
 - [Creating](#)
 - [Deleting](#)
 - [Reading](#)
- [Webhook Info](#)
 - [Webhook Backoff Algorithm](#)
 - [Webhook Events](#)
 - [Managed Client Events](#)
- [Order Simulation](#)
- [Client Shutdown](#)

- [Java Version](#)

Using the client

Instantiate an instance of `ApiV1` in order to start making calls to brawndo.

```
import com.dropoff.service.brawndo.client.java.api.ApiV1;

public class DropoffExample {
    public static void main(String[] args) {
        ApiV1 brawndo = new ApiV1();
    }
}
```

Configuration

You will then have to configure the brawndo instance with the `configure` function.

```
String url = "https://sandbox-brawndo.dropoff.com/v1";
String host = "sandbox-brawndo.dropoff.com";
String private_key = "fcb60b8680d7b5c67921a852b39067a19d85318ce8abf4c512";
String public_key = "ced2eaf24f1eaf832c1ea92b41386b4a5982cfcdb69b7c7818";

brawndo.initialize(url, host, private_key, public_key);
```

- **api_url** - the url of the brawndo api. This field is required.
- **host** - the api host. This field is required.
- **public_key** - the public key of the user that will be using the client. This field is required.
- **private_key** - the private key of the user that will be using the client.

Getting Your Client Information

If you want to know your client id and name you can access this information via the `info` call.

If you are an enterprise client user, then this call will return all of the accounts that you are allowed to manage with your current account.

```
JsonObject info = brawndo.info();
```

Note that it returns a `JsonObject` instance. The Java brawndo api has a dependency on google-gson library (included in `DropoffApi.jar`). You can get more information on that [here](#).

A response will look like this:

```
{
  success: true
  timestamp: "2017-01-25T16:51:36Z",
  data: {
    client: {
      company_name: "EnterpriseCo Global",
      id: "1111111111110"
    },
    user: {
      first_name: "Algis",
      last_name: "Woss",
      id: "2222222222222"
    },
    managed_clients: {
      level: 0,
      company_name: "EnterpriseCo Global",
      id: "1111111111110"
      children : [
        {
          level: 1,
          company_name: "EnterpriseCo Europe",
          id: "1111111111112"
          children : [
            {
              level: 2,
              company_name: "EnterpriseCo Paris",
              id: "1111111111111"
              children : []
            },
            {
              level: 2,
              company_name: "EnterpriseCo London",
              id: "1111111111113"
              children : []
            },
            {
              level: 2,
              company_name: "EnterpriseCo Milan",
              id: "1111111111114"
              children : []
            }
          ]
        }
      ]
    },
  },
}
```

```

    },
    {
      level: 1,
      company_name: "EnterpriseCo NA",
      id: "11111111111115"
      children : [
        {
          level: 2,
          company_name: "EnterpriseCo Chicago",
          id: "11111111111116"
          children : []
        },
        {
          level: 2,
          company_name: "EnterpriseCo New York",
          id: "11111111111117"
          children : []
        },
        {
          level: 2,
          company_name: "EnterpriseCo Los Angeles",
          id: "11111111111118"
          children : []
        }
      ]
    }
  ]
}
}
}
}
}

```

The main sections in data are user, client, and managed_clients.

The user info shows basic information about the Dropoff user that the used keys represent.

The client info shows basic information about the Dropoff Client that the user belongs to who's keys are being used.

The managed_clients info shows a hierarchical structure of all clients that can be managed by the user who's keys are being used.

Enterprise Managed Clients

In the above info example you see that keys for a user in an enterprise client are being used. It has clients that can be managed as it's descendants.