Discollect Developer Portal

Table of Contents

The Discollect Developer Portal provides access to data collected from all user transactions on **www.discollect.net**. Selecting and customizing particular API end-points allows for filtering data by location, time, user, or item category.

> All data is returned in JSON format.

Getting an API key

Before you can be assigned an API key, you must first sign up as a new user on Discollect. Once you have signed up, sign in and navigate to your Dashboard page. At the very bottom of the Dashboard page, clicking on '**Access Dev Portal**' will redirect you to the portal where you will be prompted for an email address.

Once you have entered this, an API key will be provided. You will also now have access to the Discollect Developer Visuals Center.

> It is important that you keep note of your API key for future use.

End-Points

An end-point is a specific internet address (or URL) that can be used to connect to and access certain information on a server. To use the given end-points for Discollect, an HTTP GET request must be made to retrieve the required information.
(For more on how to make HTTP GET requests, read here).

Every Discollect end-point you use should be preceded by the following:

```
http://www.discollect.net/
```

This means that for a given end-point of:

```
...api/YOUR_USER_ID_HERE/example
```

if your user-ID is 123, the entire end-point you must use is actually:

```
http://www.discollect.net/api/123/example
```

> The current limit is **200** API requests per day.

All items for a specific user

Fill in your own user-ID (you must request it from our help center) to receive an array of all the items you have offered in the past on Discollect.

GET request:

```
.../api/YOUR_API_KEY_HERE/user?userId=YOUR_USER_ID
```

Example response:

```
{
  "userListingData": [
    {
      "id": 33,
      "title": "Phillips Toaster",
      "createdAt": "2016-08-17T22:45:40.000Z",
      "category": "appliances",
      "zipcode": 19525
    },
    {
      "id": 32,
      "title": "Paisley duvet cover",
      "createdAt": "2016-08-17T22:29:59.000Z",
      "category": "fashion",
      "zipcode": 12345
    } ...
```

All your items sorted by category

Fill in your user-ID and receive all items sorted by categories.

GET request:

```
.../api/YOUR_API_KEY_HERE/user/category?userId= YOUR_USER_ID
```

Example response:

```
{
  "appliances": [
    {
      "id": 33,
      "title": "Phillips Toaster",
      "category": "appliances",
      "createdAt": "2016-08-17T22:45:40.000Z",
      "zipcode": 19525
    },
    {
      "id": 31,
      "title": "Older Grapes",
      "category": "appliances",
      "createdAt": "2016-08-17T18:33:04.000Z",
      "zipcode": 94102
    }
  ],
  "fashion": [] ...
```

Specific item by ID number

Receive an object with item data and accompanying click data.

GET request:

```
.../api/YOUR_API_KEY_HERE/listing?listingId=0
```

Example response:

```
{
  "listingData": [
    {
      "id": 0,
      "title": "Vacuum cleaner",
      "category": "electronics",
      "createdAt": "2016-08-16T22:09:58.000Z",
      "zipcode": 94102
    }
  ],
  "clickData": [
    {
      "userId": 8,
      "createdAt": "2016-08-18T23:05:31.000Z"
    },
    ...
```

All clicks by all users

Receive list of all clicks on all items for all users.

> This is a very large amount of data; it may take a while to retrieve.

GET request:

```
.../api/YOUR_API_KEY_HERE/all/clicks
```

Example response:

```
[
  {
    "userId": 8,
    "createdAt": "2016-08-18T23:04:35.000Z",
    "Listing": {
      "id": 32,
      "title": "Yankees cap",
      "zipcode": 12345,
      "takerId": null,
      "giverId": 2,
      "status": 0,
      "picReference": null,
      "category": "fashion",
      "description": "Handmade, a little worn on the back.",
      "condition": 2,
      "giverRating": 4.5,
      "takerRating": 0,
      "createdAt": "2016-08-17T22:29:59.000Z",
      "updatedAt": "2016-08-17T22:29:59.000Z"
    }
  } ...
```

All items by state

Receive list of all items, ordered by state.

> Use the **capitalized** two-letter abbreviation. For example: CA for California.

GET request:

```
.../api/YOUR_API_KEY_HERE/all/state?state= STATE_ABBREVIATION
```

Example response:

```
[
  {
    "id": 33,
    "title": "Phillips Toaster",
    "category": "appliances",
    "createdAt": "2016-08-17T22:45:40.000Z",
    "zipcode": 19525
  }, ...
```

One category’s clicks during a specific timeframe

Receive array of numbers of clicks on a category over a given period, divided into regular time-frames.

> Possible time-frames: ‘hour’, ‘day’, ‘month’, ‘year’

GET request:

```
...api/YOUR_API_KEY_HERE/category/time?cat=appliances&past=day
```

> In this example, the returned array is the past twenty four hours separated into twelve 2-hour segments.

Example response:

```
[
  0,
  2,
  225,
  1234,
  678,
  567,
  345,
  12,
  678,
  1456,
  90,
  2
]
```

All clicks by category within specific timeframe

Receive list of all items from selected categories (up to six) over specific time-frame.

> Currently available categories: ‘appliances’, ‘fashion’, ‘furniture’, ‘books’, ‘electronics’ and ‘tools’.

> Possible time-frames: ‘hour’, ‘day’, ‘month’, ‘year’

GET request:

```
...api/YOUR_API_KEY_HERE/time/category?cat=books&cat=furniture&past=day
```

Example response:

```
{
  "data": [
    139,
    227,
    66,
    244,
    71,
    32
  ],
  "labels": [
    "fashion",
    "appliances",
    "electronics",
    "furniture",
    "tools",
    "books"
  ],
  "label": "Clicks by Category per day"
}
```

> When filling out end-point details, use **lowercase** for all categories and times.

Developer Visuals Center

Discollect also offers a Visuals Center. Here, developers can make use of a variety of charts and selections to see displays of the Discollect data.

Additional Resources

API KEY

An API (Application Program Interface) is a set of definitions, protocols and tools for building software. It provides the building blocks for software which a developer can then make use of. In order to have access to Discollect data, a user must register and then request an API key. This key – a sequence of random letters and numbers – must then be inserted into every request to allow access to the data.

JSON

JSON is an abbreviation for JavaScript Object Notation. It is text-only (easily readable) and allows for transmission of information in key-value pairs. It is the most common data format used for asynchronous browser-server communication.

HELP-CENTER

To obtain your user ID number, please send a request to discollect123@gmail.com.