Documentation

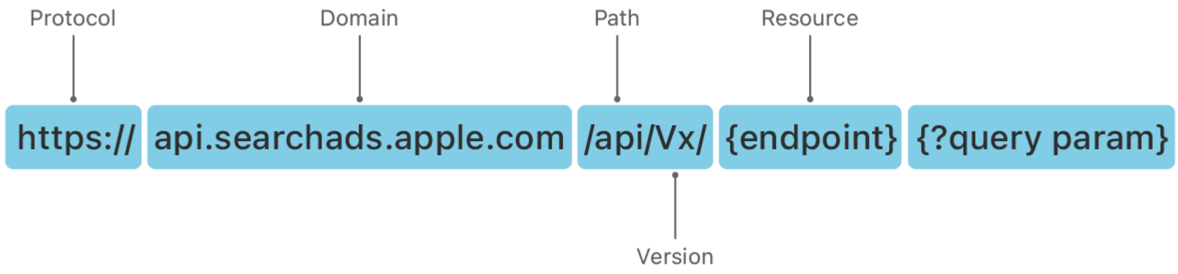
Apple Ads / Using Apple Ads API Functionality

Using Apple Ads API Functionality

Call endpoints using CRUD methods.

Overview

The Apple Ads API uses a REST data model, and you use CRUD (create, read, update, delete) functions to call endpoint resources.



CRUD methods in the Apple Ads API include the following:

POST

Creates new resources, finds resources, and pulls reports.

The API supports the use of selectors with `find` calls to filter resource data. See the `Selector` object.

GET

Reads a specific resource. Some calls use an identifier and accept `limit` and `offset` query parameters to obtain a collection of resources.

See the Use a Partial Fetch section below for query parameter functionality.

PUT

Updates a specific resource or multiple resources by identifier.

DELETE

Deletes a specific resource by identifier.

Format Requests and Responses

The API uses JSON to represent the request and response payloads. Create payloads in JSON using fields, properties, and values that the API provides. Take note of the required fields.

The following is an example POST request to `Create a Campaign`:

```
{
    "orgId": 40669820,
    "adChannelType": "SEARCH",
    "supplySources": ["APPSTORE_SEARCH_RESULTS”],
    "billingEvent": "TAPS",
    "name": "campaign 1",
    "budgetAmount": {
        "amount": "2000",
        "currency": "USD"
     },
    "dailyBudgetAmount": {
        "amount": "300",
        "currency": "USD"
     },
    "adamId": 427916203,
    "countriesOrRegions": ["US","AU"]
}
```

A response resembles the following:

```
{
    "data": {
        "id": 542317095,
        "orgId": 40669820,
        "name": "example campaign",
        "budgetAmount": {
            "amount": "2000",
            "currency": "USD"
        },
        "dailyBudgetAmount": {
            "amount": "300",
            "currency": "USD"
        },
        "adamId": 427916203,
        "paymentModel": "PAYG",
        "locInvoiceDetails": null,
        "budgetOrders": [],
        "displayStatus": "ON_HOLD",
        "adChannelType": "SEARCH",
        "supplySources": ["APPSTORE_SEARCH_RESULTS"],
        "billingEvent": "TAPS",
        "startTime": "2024-04-08T10:33:31.650",
        "endTime": "2024-04-09T10:33:31.650",
        "status": "ENABLED",
        "servingStatus": "NOT_RUNNING",
        "servingStateReasons": [
            "AD_GROUP_MISSING"
        ],
        "modificationTime": "2024-04-08T18:21:35.133",
        "deleted": false,
        "sapinLawResponse": "NOT_ANSWERED",
        "countriesOrRegions": [
            "AU",
            "US"
        ],
        "countryOrRegionServingStateReasons": {}
    },
    "pagination": null,
    "error": null
}
```

You use the `id` that returns in the response in an endpoint URI to target a resource. In the example above, the `id` serves as a `campaignId` throughout the API whenever making calls to that specific campaign resource. The same logic applies when you use the `Create an Ad Group`. The returned `id` serves as an ad group identifier (`adgroupId`) in applicable endpoints.

In other examples, when you use the `Create Targeting Keywords` and `Create Ad Group Negative Keywords` endpoints, the `id` dervies from the `campaignId` and `adgroupId` in the URI. When you use `Create Campaign Negative Keywords`, the `id` derives from the `campaignId` resource.

Use Partial Updates

A partial update is updating a subset of object properties instead of the entire set of object properties. The API supports partial updates for most PUT calls. Use a campaign object envelope in your campaign request payloads. Other objects do not require a JSON envelope.

For example, the following payload is an update to the `name`, `countriesOrRegions` of a campaign:

```
PUT https://api.searchads.apple.com/api/v5/campaigns/<campaignId>
```

```
{
  "campaign": {
    "name": "new campaign name",
    "countriesOrRegions": [
      "US",
      "CA"
    ]
  }
}
```

Use a Partial Fetch

A partial fetch indicates which properties to return in a response when fetching data. Use a partial fetch where a `Selector` is available by using optional field parameters to limit the fields within each record that returns. Use a partial fetch with some GET endpoints by specifying the `field` query string parameter. For example, you can choose to only return the `id` for each campaign, as the following example shows:

```
GET https://api.searchads.apple.com/api/v5/campaigns?fields=id,name,countriesOrRegi
```

```
{
    "data": [
        {
            "id": 542370539,
            "name": "Multicountry campaign example 1",
            "countriesOrRegions": [
                "FR"
            ]
        },
        {
            "id": 542370549,
            "name": "Multicountry campaign example 2",
            "countriesOrRegions": [
                "GB"
            ]
        },
        {
            "id": 542370559,
            "name": "Multicountry campaign example 3",
            "countriesOrRegions": [
                "CA",
                "JP",
                "NZ",
                "AU"
            ]
        }
    ],
    "pagination": {
        "totalResults": 3,
        "startIndex": 0,
        "itemsPerPage": 20
    },
    "error": null
}
```

Use Query Parameters

You use query parameters with the `Search for iOS apps`, `Search for Geolocations`, and `Get a List of Geolocations` endpoints by appending the query parameter to the URI in method calls.

GET calls use query parameters to control `Pagination` and to assign a limit to returned responses. You can combine these query parameters with data to fetch, such as in the following example:

```
GET https://api.searchads.apple.com/api/v5/search/geo?entity=Locality&limit=1000
```

Topics

Responses

Usability and Errors

Additional objects to use for reference.

See Also

Essentials

Implementing OAuth for the Apple Ads API

Manage secure access to Ads accounts.

Calling the Apple Ads API

Pass your access token in the authorization header of HTTP requests.