

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

Coupon

List coupons

Retrieve a list of all coupons.

POST<https://api.coupontools.com/v3/coupon/list>

Authentication: [API Keys](#)

Parameters

Optional **only_active** true | false

If set to true, only campaigns marked as 'active' will be returned

Optional **folder** Folder ID

If defined, the response will only contain coupons in the specified folder

Optional **tags** String of coupon tags

If defined, the response will only contain coupons containing that tag. Comma separated for each new tag

Optional **category** Category ID (cat_XXX)

If defined, the response will only contain coupons of that category

Example

```
1 const https = require("https");           Node ✓ 
2 const client_id = "XXX"; // Your API client ID (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   only_active: true
6 });
7 const options = {
8   hostname: "api.coupontools.com",
9   port: 443,
10  path: "/v3/coupon/list",
```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

17      },
18  };
19 const req = https.request(options, (res) => {
20   res.on("data", (d) => {
21     process.stdout.write(d);
22   });
23 });
24 req.write(data);
25 req.end();

```

Response

```
{
  "status": {
    "status": "OK"
  },
  "amount_of_campaigns": "1",
  "coupon_info": [
    {
      "ID": "cam_123456",
      "code": "tn91r7",
      "url": "https://digicpn.com/p/tn91r7",
      "status": "active", (example | active | archived | terminated)
      "subaccount": "sub_12345",
      "timezone": "America/New_York",
      "expirydate": "2018-01-01 04:59:00", (UTC)
      "name": "API EXAMPLE CAMPAIGN",
      "title": "Coupon title",
      "subtitle": "Coupon subtitle",
      "description": "Coupon description content goes here",
      "terms": "Coupon full terms and conditions goes here",
      "terms_short": "Coupon short terms and conditions goes here",
      "poweredby": "MyName",
      "poweredbylink": "http://www.mywebsite.com",
      "coupon_tags": "Coupon,tags,here",
      "coupon_category": "1",
      "coupon_category_name": "Other",
      "logo_url": "https://example.com/logo.png",
      "coupon_image": "https://example.com/coupon_image.jpg",
      "banner1": "https://example.com/banner.jpg",
      "locations": [
        {
          "name": "New York"
        }
      ]
    }
  ]
}
```

Possible errors

515: Folder not found

Overview

Coupon

Directory

Loyalty

Account

Others

Authentication: [API Keys](#)

Parameters

Required **campaign** Coupon ID (cam_XXXXXX)

Can be found in the list of all coupons

Optional **show_usage_stats** true | false

Defaults to false, will add the total number of custom validation codes and number of claimed custom codes

Example

```
1 const https = require("https");           Node ▾ □
2 const client_id = "XXX"; // Your API client ID (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   campaign: "cam_123456"
6 });
7 const options = {
8   hostname: "api.coupontools.com",
9   port: 443,
10  path: "/v3/coupon/info",
11  method: "POST",
12  headers: {
13    "X-Client-Id": client_id,
14    "X-Client-Secret": client_secret,
15    "Content-Type": "application/json",
16    "Content-Length": data.length
17  },
18 };
19 const req = https.request(options, (res) => {
20   console.log(`statusCode: ${res.statusCode}`);
21   res.on("data", (d) => {
22     process.stdout.write(d);
23   });
24 });
25 req.write(data);
26 req.end();
```

Response

```
{
  "status": {
```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

    "status": "example",
    "subaccount": "",
    "name": "[NO NAME]",
    "title": "10% OFF",
    "subtitle": "On all pizzas",
    "description": "",
    "expirydate": "2018-08-12 11:19:00",
    "timezone": "UTC",
    "terms": "",
    "terms_short": "Example coupon;",
    "poweredby": "Your Company",
    "poweredbylink": "http://www.yourwebsite.com",
    "coupon_tags": "",
    "coupon_category": "1",
    "coupon_category_name": "",
    "logo_url": "https://hosting4images.com/upload/logo/302",
    "banner1": "https://www.hosting4images.com/preview/imag",
    "banner2": "https://www.hosting4images.com/preview/imag",
    "banner3": "https://www.hosting4images.com/preview/imag",
    "banner4": "https://www.hosting4images.com/preview/imag",
    "banner5": "https://www.hosting4images.com/preview/imag",
    "coupon_image": "https://hosting4images.com/upload/cust"
  
```

Possible errors

200: Missing parameter: campaign

301: Invalid character in parameter: campaign

502: Campaign ID not found

Coupon session

List coupon sessions

Retrieve a list of all sessions from a coupon.

| | |
|------|---|
| POST | https://api.coupontools.com/v3/campaign/data |
|------|---|

Only sessions that are opened

| | |
|------|---|
| POST | https://api.coupontools.com/v3/campaign/opens |
|------|---|

Only sessions that are claimed

| | |
|------|---|
| POST | https://api.coupontools.com/v3/campaign/claims |
|------|---|

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

Parameters

Required **campaign** Coupon ID (cam_XXXXXX)

Can be found in the list of all coupons

Optional **page** Numeric

Page ID. If there are 100 or more records these will be split in multiple pages.

Optional **start_date** Date (format: 'yyyy-MM-dd hh-mm' | 'yyyy-MM-dd')

Start date of the sessions in UTC. Inclusive.

Optional **end_date** Date (format: 'yyyy-MM-dd hh-mm' | 'yyyy-MM-dd')

End date of the sessions in UTC. Inclusive.

Example

```

1 const https = require("https");           Node    ✓   □
2 const client_id = "XXX"; // Your API client id (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   campaign: "cam_123456"
6 });
7 const options = {
8   hostname: "api.coupontools.com",
9   port: 443,
10  path: "/v3/campaign/data",
11  method: "POST",
12  headers: {
13    "X-Client-Id": client_id,
14    "X-Client-Secret": client_secret,
15    "Content-Type": "application/json",
16    "Content-Length": data.length
17  },
18 };
19 const req = https.request(options, (res) => {
20   console.log(`statusCode: ${res.statusCode}`);
21   res.on("data", (d) => {
22     process.stdout.write(d);
23   });
24 });
25

```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

    "status": {
        "status": "OK"
    },
    "campaign": "cam_12345",
    "action": "retrieve_all_data",
    "data": [
        {
            "session": "kkfiznr8gp8630h82es8d",
            "sequenceid": "19878460",
            "status": {
                "created_date_utc": "2019-03-25 10:43:50",
                "opened": "0",
                "open_date_utc": "",
                "played": "0",
                "played_date_utc": "",
                "claimed": "1",
                "claim_date_utc": "2019-03-25 10:44:05",
                "claimtype": "",
                "saved_to_passbook": "0",
                "saved_to_passbook_date_utc": "",
                "saved_to_androidpay": "0",
                "saved_to_androidpay_date_utc": "",
                "validated": "0",
                "validation_date_utc": "",
                "own_validation_code": "ding_C0dHepKvqNrNC8OU",
                "validation_value": "",
                "redeem_location_ID": "",
                "voided": "0",
                "unvoided_date_utc": "2019-03-25 10:44:05",
                "locked": "0"
            }
        }
    ]
}

```

Possible errors

200: Missing parameter: campaign

301: Invalid character in parameter: campaign

301: Invalid character in parameter: page

303: Parameter end_date is not a valid date

303: Parameter end_date must be in the following format:
yyyy-MM-dd hh:mm or yyyy-MM-dd

303: Parameter start_date is not a valid date

303: Parameter start_date must be in the following format:
yyyy-MM-dd hh:mm or yyyy-MM-dd

502: Campaign ID not found

Get coupon session

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

Authentication: [API Keys](#)

Parameters

Required **couponsession** String

Unique session code (example: ac6s7u3g8a25ja7x9qg87). Can be found in the URL, webhooks, statistics or through API call

Example

```

1 const https = require("https");           Node ↻
2 const client_id = "XXX"; // Your API client id (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   couponsession: "ac6s7u3g8a25ja7x9qg87"
6 });
7 const options = {
8   hostname: "api.coupontools.com",
9   port: 443,
10  path: "/v3/couponsession/data",
11  method: "POST",
12  headers: {
13    "X-Client-Id": client_id,
14    "X-Client-Secret": client_secret,
15    "Content-Type": "application/json",
16    "Content-Length": data.length
17  },
18 };
19 const req = https.request(options, (res) => {
20   console.log(`statusCode: ${res.statusCode}`);
21   res.on("data", (d) => {
22     process.stdout.write(d);
23   });
24 });
25 req.write(data);
26 req.end();

```

Response

```
{
  "status": {
    "status": "OK"
  },
  "campaign": "cam_12345",
  "action": "retrieve_all_session_data",
  "session": "kkfiznr8Yp8630h82es8d",
  "session_url": "https://digicpn.com/p/v6n7gn/kkfiznr8Yp8630h82es8d"
}
```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

    "open_date_utc": "",
    "played": "0",
    "played_date_utc": "",
    "claimed": "1",
    "claim_date_utc": "2019-03-25 10:44:05",
    "claimtype": "",
    "saved_to_passbook": "0",
    "saved_to_passbook_date_utc": "",
    "saved_to_androidpay": "0",
    "saved_to_androidpay_date_utc": "",
    "validated": "0",
    "validation_date_utc": "",
    "own_validation_code": "ding_C0dHepKvqNrNC80U",
    "validation_value": "",
    "redeem_location_ID": "",
    "voided": "0"
  
```

Possible errors

200: Missing parameter: couponsession

300: Invalid character in parameter: couponsession

513: Coupon session not found

Update session

Update data from 1 coupon session. For example, set the session from claimed to validated status.

POST

<https://api.coupontools.com/v3/couponsession/update>

Authentication: [API Keys](#)

Parameters

Required **couponsession** String

Unique session code (example: ac6s7u3g8a25ja7x9qg87). Can be found in the URL, webhooks, statistics or through API call

Optional **operation** void | unvoid | lock | unlock | claim | validate |
unvalidate | activate

void: sets coupon session to voided status
 unvoid: unsets the voided status
 lock: sets coupon session to locked status
 unlock: unsets the locked status

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

Optional **customfield1 - customfield7** String

Custom field 1 through 7. Max. 50 characters

Optional **customid** Custom ID

Max. 50 characters (az-AZ-09)

Optional **money_spent** Decimal number

Money spent. (Format: 0.000) Do not use special characters

Example

```

1 const https = require("https");
2 const client_id = "XXX"; // Your API client id (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   couponsession: "ac6s7u3g8a25ja7x9qg87",
6   operation: "validate"
7 });
8 const options = {
9   hostname: "api.coupontools.com",
10  port: 443,
11  path: "/v3/couponsession/update",
12  method: "POST",
13  headers: {
14    "X-Client-Id": client_id,
15    "X-Client-Secret": client_secret,
16    "Content-Type": "application/json",
17    "Content-Length": data.length
18  },
19 };
20 const req = https.request(options, (res) => {
21   console.log(`statusCode: ${res.statusCode}`);
22   res.on("data", (d) => {
23     process.stdout.write(d);
24   });
25 });
26 req.write(data);
27 req.end();

```

Response

```
{
  "status":{
    "status": "OK"
  }
}
```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

        "operation": "validate_successful"
    }

```

Possible errors

- 200:** Missing parameter: couponsession
- 300:** Invalid character in parameter: couponsession
- 300:** Invalid character in parameter: customid
- 302:** Invalid value in parameter: operation
- 351:** Session or coupon has expired
- 352:** Session already was claimed before
- 352:** Session could not be locked because it is not claimed
- 353:** Session was already unvalidated
- 354:** This coupon is locked
- 354:** Session already was locked before
- 355:** Session could not be locked because it is voided
- 355:** Session could not be unlocked because it is voided
- 355:** Session was already voided
- 355:** This coupon is voided
- 356:** Session was already unlocked
- 357:** Session wasn't voided
- 358:** This session was already validated
- 358:** Session could not be locked because it is already validated
- 512:** Coupon session not found

Update sessions

Update data from multiple coupon sessions. For example, set the sessions from claimed to validated status. Must define at least start sequence and/or end sequence. The API will update all sessions after, before, or between start and end sequence.

| | |
|------|---|
| POST | https://api.coupontools.com/v3/batch/update |
|------|---|

Authentication: [API Keys](#)

Parameters

Required **campaign** Coupon ID (cam_XXXXXX)

Can be found in the list of all coupons

Overview

Coupon

Directory

Loyalty

Account

Others

Required **endsequence** The sequence ID of the last session

Returned in multiplesingleuse/create. Only 1 of "startsequence" or "endsequence" is required

Optional **operation** void | unvoid | lock | unlock | claim | activate

void: sets coupon session to voided status

unvoid: unsets the voided status

lock: sets coupon session to locked status

unlock: unsets the locked status

claim: sets coupon session to claimed status.

activate: will unvoid and start the session, if the coupon has a flexible expiry date. the expiration time will be calculated.

Optional **customfield1** -
customfield7

Add data to custom field 1 through 7

Max. 50 characters

Example

```
1 const https = require("https");           Node ✓
2 const client_id = "XXX"; // Your API client ID (required)
3 const client_secret = "YYY"; // Your API client secret (required)
4 const data = JSON.stringify({
5   campaign: "cam_123456",
6   startsequence: "1000000",
7   operation: "validate"
8 });
9 const options = {
10   hostname: "api.couptoools.com",
11   port: 443,
12   path: "/v3/batch/update",
13   method: "POST",
14   headers: {
15     "X-Client-Id": client_id,
16     "X-Client-Secret": client_secret,
17     "Content-Type": "application/json",
18     "Content-Length": data.length
19   },
20 };
21 const req = https.request(options, (res) => {
22   console.log(`statusCode: ${res.statusCode}`);
23   res.on("data", (d) => {
24     process.stdout.write(d);
25   });
26 });
27
28 req.end(data);
```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```
{
  "status": {
    "status": "OK"
  },
  "action_date": "2019-04-10 11:35:36",
  "action_quantity": 3
}
```

Possible errors

- 200:** no operation or data provided
- 200:** startsequence or endsequence not provided
- 301:** start/endsequence must be numeric
- 302:** status invalid, valid statuses are: X
- 351:** Some session(s) are expired
- 352:** Some session(s) were already claimed
- 354:** Some session(s) were already locked
- 355:** Some session(s) could not be claimed because they're voided
- 355:** Some session(s) were already voided
- 356:** Some session(s) weren't locked
- 357:** Some session(s) weren't voided
- 358:** Some session(s) were already validated
- 359:** Some session(s) weren't claimed
- 360:** Startdate not reached
- 502:** Provided start and end are not of the provided campaign
- 502:** Provided start and end are not of the same campaign
- 513:** End ID comes before start ID
- 513:** End ID not found
- 513:** End ID not found or not part of the campaign
- 513:** Start ID not found
- 513:** Start ID not found or not part of the campaign

Search session

Search coupon session based on your custom validation code, phone number or email.

| | |
|------|---|
| POST | https://api.coupontools.com/v3/couponsession/search |
|------|---|

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

Parameters

Optional **campaign** Coupon ID (cam_XXXXXX)

Only required when searching in specific coupon. Can be found in the list of all coupons.

Optional **code** String

Custom validation code. Minimum 5 characters

Optional **phone** Phone number

Minimum 5 characters

Optional **email** Email address

Minimum 5 characters

Example

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

  ...
7 const options = {
8   hostname: "api.coupontools.com",
9   port: 443,
10  path: "/v3/couponsession/search",
11  method: "POST",
12  headers: {
13    "X-Client-Id": client_id,
14    "X-Client-Secret": client_secret,
15    "Content-Type": "application/json",
16    "Content-Length": data.length
17  },
18};
19 const req = https.request(options, (res) => {
20   console.log(`statusCode: ${res.statusCode}`);
21   res.on("data", (d) => {
22     process.stdout.write(d);
23   });
24 });
25 req.write(data);
26 req.end();

```

Response

```
{
  "status": {
    "status": "OK"
  },
  "results_found": "1",
  "searchresults": [
    {
      "campaign": "cam_12345",
      "session": "rg4u1ka3snccb5hrxyhiw",
      "single_use_url": "https://digicpn.com/p/v6n7gn/rg4u",
      "custom_validation_code": "1020402010203040053",
      "customid": "",
      "coupon": {
        "ID": "cam_1010951",
        "code": "v6n7gn",
        "url": "https://digicpn.com/p/v6n7gn",
        "status": "test",
        "coupon_tags": ""
      },
      "data": {
        "status": {
          "created_date_utc": "2019-03-25 10:43:50",
          "opened": "0",
          "open_date_utc": ""
        }
      }
    }
  ]
}
```

Overview

Coupon

Directory

Loyalty

Account

Others

Possible errors

200: No filter defined**200:** No parameter campaign**300:** Search field 'code' must be at least 5 characters**300:** Search field 'email' must be at least 5 characters**300:** Search field 'phone' must be at least 5 characters**502:** campaign not found**512:** No results found

Delete session

Delete a coupon session.

POST<https://api.coupontools.com/v3/singleuse/remove>

Authentication: [API Keys](#)

Parameters

Required **campaign** Coupon ID (cam_XXXXXX)

Can be found in the list of all coupons

Required **couponsession** String

Unique session code (example: ac6s7u3g8a25ja7x9qg87). Can be found in the URL, webhooks, statistics or through API call

Example

```

1 const https = require("https");
2 const client_id = "XXX"; // Your API client id (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   campaign: "cam_123456",
6   couponsession: "ac6s7u3g8a25ja7x9qg87"
7 });
8 const options = {
9   hostname: "api4coupons.com",
10  port: 443,
11  path: "/v3/singleuse/remove",
12  method: "POST",
13  headers: {

```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

20 const req = https.request(options, (res) => {
21   console.log(`statusCode: ${res.statusCode}`);
22   res.on("data", (d) => {
23     process.stdout.write(d);
24   });
25 });
26 req.write(data);
27 req.end();

```

Response

```
{
  "status": {
    "status": "OK"
  },
  "campaign": "cam_12345",
  "action": "single_use_code_removed",
  "single_use_code": "ac6s7u3g8a25ja7x9qg87"
}
```

Possible errors

200: Missing parameter: campaign

301: Invalid character in parameter: campaign

502: Campaign ID not found

513: Coupon session not found

Generate session barcode

Generate a barcode for a session.

POST

<https://api.coupontools.com/v3/couponsession/barcode>

Authentication: [API Keys](#)

Parameters

Required **couponsession** String

Unique session code (example: ac6s7u3g8a25ja7x9qg87). Can be found in the URL, webhooks, statistics or through API calls

Required **type** code39 | code128b | pdf417 | aztec | qrcode

Type of the barcode.

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

1 const data = {
2   "couponsession": "ac6s7u3g8a25ja7x9qg87",
3   "type": "code39"
4 };
5
6 const options = {
7   "hostname": "api.coupontools.com",
8   "port": 443,
9   "path": "/v3/couponsession/barcode",
10  "method": "POST",
11  "headers": {
12    "X-Client-Id": clientId,
13    "X-Client-Secret": clientSecret,
14    "Content-Type": "application/json",
15    "Content-Length": data.length
16  }
17};
18
19
20 const request = https.request(options, (response) => {
21   console.log(`statusCode: ${response.statusCode}`);
22   response.on("data", (d) => {
23     process.stdout.write(d);
24   });
25 });
26 request.write(data);
27 request.end();

```

Possible errors

400: Missing parameter: couponsession

400: Couponsession {couponsession} does not exist

400: Missing parameter: type

400: Unsupported type: {type}

Single-use

Create single-use URL

Create a single-use URL for a coupon. Prefill data like first name, last name, custom validation code, ...

POST

<https://api.coupontools.com/v3/singleuse/create>

Authentication: [API Keys](#)

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

Optional **firstname** First name

Optional **lastname** Last name

Optional **email** Email

Optional **phone** Phone

Optional **customid** String

Custom ID. Max. 50 characters (az-AZ-09)

Optional **customfield1 - customfield7** String

Custom field 1 through 7 Max. 50 characters

Optional **customvalcode** String

Custom validation code Max. 100 characters (az-AZ-09). Cannot be combined with 'autovalcode'

Optional **autovalcode** 6digits | 8digits | 10digits

Automatically generates unique validation code. Cannot be combined with 'customvalcode'

Optional **duplicate_check** email | phone | customid

Defaults to false. If enabled, we will verify if the same user input already received a single-use url for that coupon based on the selected field (email, phone or customid), and return the duplicated session instead of creating a new session

Optional **utm_campaign | utm_source | utm_medium | utm_term | utm_content** String

Google Analytics tags. Will appear in exported statistics

Optional **expiry_date** Date (format: 'yyyy-MM-dd HH:mm:ss')

This expiration date is for this single use session and will override the campaign expiration date. The existing timezone from the

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   campaign: "cam_123456"
6 });
7 const options = {
8   hostname: "api.coupontools.com",
9   port: 443,
10  path: "/v3/singleuse/create",
11  method: "POST",
12  headers: {
13    "X-Client-Id": client_id,
14    "X-Client-Secret": client_secret,
15    "Content-Type": "application/json",
16    "Content-Length": data.length
17  },
18 };
19 const req = https.request(options, (res) => {
20   console.log(`statusCode: ${res.statusCode}`);
21   res.on("data", (d) => {
22     process.stdout.write(d);
23   });
24 });
25 req.write(data);
26 req.end();

```

Response

```
{
  "status": {
    "status": "OK"
  },
  "campaign": "cam_12345",
  "action": "single_use_code_created",
  "single_use_code": "c43403t2fv7qz27in745p",
  "single_use_url": "https://digicpn.com/p/abc123/c43403t2fv7qz",
  "widget_validation_code": "c43403",
  "custom_validation_code": "102030405060708",
  "duplicate_found": true (only returned when checking for dupl
}
```

Possible errors

200: Missing parameter: campaign

301: Invalid character in parameter: campaign

502: Campaign ID not found

COUPON

[List coupons](#)

[Get coupon](#)

[Create multiple single-use URLs](#)

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)[Update session](#)**Authentication:** [API Keys](#)[Update sessions](#)**Parameters**[Search session](#)Required **campaign** Coupon ID (cam_XXXXXX)[Delete session](#)

Can be found in the list of all coupons

[Generate session barcode](#)**SINGLE-USE**[Create single-use URL](#)Optional **amount** Number

Amount of single-use URLs to generate. Defaults to 10

[Create multiple single-use URLs](#)Optional **status** normal | voided | activated

Defaults to normal

DISTRIBUTIONOptional **automated_codes** true | false

If true, requires the parameters below. Defaults to false

[Send coupon by email](#)Optional **code_prefix** String

Prefix of custom validation code

NOTIFICATION[Send push notification](#)Optional **code_length** Length of code excluding prefix

Number between 10 and 25 inclusive

Optional **code_type** numeric | alphanumeric

The type of the custom validation code

Example

```

1 const https = require("https");
2 const client_id = "XXX"; // Your API client id (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   campaign: "cam_123456",
6   amount: "10"
7 });
8 const options = {
9   hostname: "api.coupontools.com",
10  port: 443,
11  path: "/v3/multiplesingleuse/create",
12  method: "POST",
13  headers: {

```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

20 const req = https.request(options, (res) => {
21   console.log(`statusCode: ${res.statusCode}`);
22   res.on("data", (d) => {
23     process.stdout.write(d);
24   });
25 });
26 req.write(data);
27 req.end();

```

Response

```
{
  "status": {
    "status": "OK"
  },
  "campaign": "cam_12345",
  "action": "single_use_code_created",
  "single_use_code": "c43403t2fv7qz27in745p",
  "single_use_url": "https://digicpn.com/p/abc123/c43403t2fv7qz",
  "widget_validation_code": "c43403",
  "custom_validation_code": "102030405060708",
  "duplicate_found": true (only returned when checking for dupl
}
```

Possible errors

200: Missing parameter: campaign

301: Invalid character in parameter: amount

301: code length must be between 3 and 30 inclusive

301: Invalid character in parameter: campaign

302: code_type must be "numeric" or "alphanumeric"

302: Parameter "status" has invalid value. Possible values are "normal", "voided" and "active"

502: Campaign ID not found

601: low on unique generated validation codes

Distribution

Send coupon by email

You need to have an email marketing platform connected to be able to send out marketing emails (check out: integrations -> email marketing).

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

Parameters

Required **campaign** Coupon ID (cam_XXXXXX)

Can be found in the list of all coupons

Required **email** Email

Email address receiving coupon

Optional **couponsession** String

Unique session code (example: ac6s7u3g8a25ja7x9qg87). If not defined, we'll generate a new coupon session

Optional **subject** String

Subject of email

Optional **firstname** String

Personalize email with a first name. The email body must contain [FIRSTNAME]

Optional **lastname** String

Personalize email with a last name. The email body must contain [LASTNAME]

Optional **check_unique** true | false

Check if email address already received coupon. When not unique, email is not send

Example

```
1 const https = require("https");           Node    ▾   □
2 const client_id = "XXX"; // Your API client id (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5     campaign: "cam_123456",
6     email: "john@doe.com"
7 });
8 const options = {
9     hostname: "api.coupontools.com",
10    port: 443,
11    path: "/v3/send/email",
```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

18      },
19  };
20 const req = https.request(options, (res) => {
21   console.log(`statusCode: ${res.statusCode}`);
22   res.on("data", (d) => {
23     process.stdout.write(d);
24   });
25 });
26 req.write(data);
27 req.end();

```

Response

```
{
  "status": {
    "status": "OK"
  },
  "campaign": "cam_123456",
  "subject": "Subject here",
  "receiver": "info@receiver.com",
  "single_use_code": "0ist0ywa4t9z7rj0g8mf7",
  "single_use_url": "https://digicpn.com/p/abc123/0ist0ywa4t9z7",
  "action": "email_sent"
}
```

Possible errors

104: No Email integration has been setup in your Coupontools account. Go to 'integrations' and 'Email marketing' to setup an Email provider

200: Missing parameter: campaign

301: Invalid character in parameter: campaign

305: Email address is not valid

502: Campaign ID not found

604: Email not sent to 'john@doe.com'. Reason: X

Send coupon by text message

You need to have an SMS marketing platform connected to be able to send out SMS messages (check out: integrations -> SMS marketing).

E.164 phone number format:

First of all, in the international E.164 notation, a leading '0' is

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

characters should be numeric.

Example: +447911123456

POST<https://api.coupontools.com/v3/send/sms>

Authentication: [API Keys](#)

Parameters

Required **campaign** Coupon ID (cam_XXXXXX)

Can be found in the list of all coupons

Required **phone** Phone number

Phone number receiving coupon. International E.164 notation

Optional **couponsession** String

Unique session code (example: ac6s7u3g8a25ja7x9qg87). If not defined, we'll generate a new coupon session

Optional **body** String

SMS body text. The unique coupon URL will be added to each message

Example

```

1 const https = require("https");           Node   ▾
2 const client_id = "XXX"; // Your API client ID (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   campaign: "cam_123456",
6   phone: "+123456789"
7 });
8 const options = {
9   hostname: "api.coupontools.com",
10  port: 443,
11  path: "/v3/send/sms",
12  method: "POST",
13  headers: {
14    "X-Client-Id": client_id,
15    "X-Client-Secret": client_secret,
16    "Content-Type": "application/json",

```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

```

23         process.stdout.write(d);
24     });
25 });
26 req.write(data);
27 req.end();

```

Response

```
{
  "status": {
    "status": "OK"
  },
  "campaign": "cam_123456",
  "message": "Here comes the body text https://digicpn.com/p/ab",
  "phone": "441234567890",
  "single_use_code": "d8ysxk1wqy6ajzh65atx7",
  "single_use_url": "https://digicpn.com/p/abc123/d8ysxk1wqy6aj",
  "action": "sms_sent"
}
```

Possible errors

103: No SMS integration has been setup in your Coupontools account. Go to 'integrations' and 'SMS marketing' to setup an SMS provider

200: Missing parameter: campaign

301: Invalid character in parameter: campaign

301: Phone number not valid

502: Campaign ID not found

513: Statcode not found

602: SMS not sent to 'X'. Reason: Y

605: SMS not sent to 'X'. Reason: This number is on the blacklist or opt-out list.

Notification

Send push notification

Send push notification to all users of a coupon. This requires the coupon to be added in the mobile wallet.

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

Parameters

Required **campaign** Coupon ID (cam_XXXXXX)

Can be found in the list of all coupons

Required **message** String

Push notification message

Example

```
1 const https = require("https");           Node ✓ 
2 const client_id = "XXX"; // Your API client id (required)
3 const client_secret = "YYY"; // Your API client secret (req
4 const data = JSON.stringify({
5   campaign: "cam_123456",
6   message: "Example"
7 });
8 const options = {
9   hostname: "api.coupontools.com",
10  port: 443,
11  path: "/v3/coupon/pushnotification/list",
12  method: "POST",
13  headers: {
14    "X-Client-Id": client_id,
15    "X-Client-Secret": client_secret,
16    "Content-Type": "application/json",
17    "Content-Length": data.length
18  },
19 };
20 const req = https.request(options, (res) => {
21   console.log(`statusCode: ${res.statusCode}`);
22   res.on("data", (d) => {
23     process.stdout.write(d);
24   });
25 });
26 req.write(data);
27 req.end();
```

Response

```
{
  "status": {
    "status": "OK"
  }
}
```

[Overview](#)[Coupon](#)[Directory](#)[Loyalty](#)[Account](#)[Others](#)

507: Error sending push notification

507: 0 users have saved the campaign