

Getting Started with the Spendgo API

Spendgo's suite of APIs enables you to integrate customer data and loyalty into any mobile, in-store POS, kiosks, e-commerce platforms, and more.

The customer data maintained by Spendgo is the source of truth. Hence, the Partner will contact Spendgo when a customer submits a new account creation request (sign up process), when a registered customer needs to sign in, and when a customer updates their account profile information.

Spendgo provides a simple and powerful [REST API](#) to integrate with our customer data and loyalty services. Our API has predictable resource-oriented URLs, returns JSON responses, and uses standard HTTP response codes.

In order to ensure that a high level of quality is maintained, Spendgo provides an environment that can be used during development and integration for testing purposes before rolling out changes to the live environment.

Test Environment

You can use the Spendgo API in a test environment, which does not affect your live data. To test that your integration is working correctly, make test API requests using your test credentials pointed to ***.skuped.com** domain.

Live Environment

Once your integration has been certified by us, your API requests using your production credentials should be pointed to ***.spendgo.com** domain.

Request & Response Protocol

Security

Depending on the use case, these are the following three security mechanisms available when interacting with Spendgo's platform:

- [Signed Requests](#)
Frequently used for server-to-server integrations.
- [OAuth2](#)
When leveraging Spendgo's Identity platform for user-authentication.
- [Token-Based Requests](#)
Frequently used for building mobile apps. A user id and password are exchanged for a token to be used in subsequent API requests.

Request Structure

Every request comprises HTTP Headers, URL endpoint and optionally a JSON Body. All JSON structures follow the rules specified in RFC4627; and encoded as UTF8, as specified in RFC3629, using one to four bytes per Unicode code point.

HTTP Headers

Spendgo APIs use the following headers. Not all APIs require all the header elements in the request to be passed.

- **X-Class-Signature** the message signature. The X-Class-Signature header is generated by creating a sha256 HMAC using the shared secret key (known only to

both parties) on the prehash string:

```
hmac("sha256", (uri + body), Base64Decoded(base64_encoded_secret))
```

where + represents string concatenation

See [Generate Signature Examples](#).

- **X-Class-Key** the account specific identifier provided by Spendgo.
- **Content-Type** identify JSON data format being sent as “application/JSON”
- **Authorization** is the authentication token generally for web or mobile application workflows seeking access to secure customer profile information. The Authorization Key expires after a pre-defined expiry period.

JSON Body

The JSON body is the minified version of the JSON that is to be submitted in the request body. It is omitted if there is no request body (typically for GET requests). NOTE: This same minified version of the JSON body is used to generate the signature as well.

Example cURL command:

```
curl -k -v -X POST -H 'X-Class-Key: test_spendgo_loyalty' -H  
'X-Class-Signature: EMTNgEMeW/E32z/hIFDn1tTVQ017GLC4SiBHNWwRx9w=' -H  
'Content-Type: application/json' -d '{"customer":{"phone":"9999999999"}}'  
https://webservice.skuped.com/loyalty/accounts
```

Response Structure

Spendgo APIs use a combination of HTTP status codes and a JSON Response Body to present the results of the call. Every response body is serialized as JSON, as specified in RFC4627; and encoded as UTF8, as specified in RFC3629, using one to four bytes per Unicode code point.

Error Handling

Spendgo uses conventional HTTP response codes to indicate the success or failure of an API request. In general: Codes in the 2xx range indicate success. Codes in the 4xx range indicate an error that failed given the information provided (e.g., a required parameter was omitted, an order failed, etc.).

Some 4xx errors that could be handled programmatically (e.g., an order failed) include an error code that briefly explains the error reported.

JSON response

id	string	A unique reference representing this instance of the error.
code	integer	Machine readable code representing the reason for the error.
message	string	Optional human-readable message providing more details about the error.
details	string	Optional customer facing error message for e-commerce UI and/or customer email notifications.

HTTP Status Code Summary

Spendgo APIs return HTTP compliant codes as follows:

OK	200	Everything worked as expected.
CREATED	201	Object in the request is created.
ACCEPTED	202	Object in the request is accepted.

BAD REQUEST	400	The request was unacceptable, often due to missing a required parameter.
UNAUTHORIZED	401	No valid API key provided.
FORBIDDEN	403	The API key doesn't have permissions to perform the request or other reward or member information does not match.
NOT FOUND	404	The requested resource doesn't exist.
CONFLICT	409	The request encountered a conflict.
INTERNAL SERVER ERROR	500, 501, 503	Something went wrong on Spendgo's end (these are rare).

In general, the standard followed in mapping the HTTP codes is based on RFC7231 and is summarized below:

- In the 2xx range if a request is successful
- In the 3xx range if a request is redirected
- In the 4xx range if a request fails due to client error
- In the 5xx range if a request fails due to server error

API Reference

Members

Create a member

[#mobile](#) [#web](#)

Creates a member associated with your account without member sign in.

HTTP request

```
POST https://webservice.<ENVIRONMENT>.com/mobile/gen/{account_id}/v1/add
```

JSON body

phone integer	Required member's phone number.
email string	Required member's email address.
password string	Required member's password.
first_name string	Optional member's first name.
last_name string	Optional member's last name.
dob string	Optional member's date of birth formatted as YYYY-MM-DD.
gender string	Optional member's gender. Values are "m" or "f".
marital_status string	Optional member's marital status. Values are "single", "married", "divorced", or "domestic partner".

street	string	Optional member's street address.
city	string	Optional member's city address.
state	string	Optional member's 2-letter state abbreviation address (e.g. CA)
zip	integer	Optional member's zip code address.
favorite_store_id	integer	Optional member's favorite store number
sms_opt_in	boolean	Optional member's text subscription preference for your brand. Once the user has validated their account creation, they will be sent the text double opt-in flow to confirm their text subscription.
email_opt_in	boolean	Optional member's email marketing preference for your brand.
send_welcome_email	boolean	Default value is "false" to not send Spendgo-powered welcome email upon successful creation.
email_validated	boolean	Default value is "false".
addtl_info	array	Optional
name	string	Optional
value	string	Optional

Example request:

```
curl -k -v -X POST -H 'X-Class-Key: test_spendgo_mobile' -H
'X-Class-Signature: rXzoAFj5TaX2cF62a701upistZ5mEj909NNgiaauI3gh' -H
'Content-Type: application/json' -H 'Authorization: Bearer
aBcoAGk2Nag1cF98a957udpotZ3nPs808LLedapdI6xi' -d '{ "phone": "4155555555",
"email": "Jane@example.com", "password": "", "first_name": "Jane",
"last_name": "Smith", "dob": "2000-12-31", "gender": "f",
```

```
"marital_status": "", "street": "1st Avenue", "city": "San Francisco",  
"state": "CA", "zip": 94107, "favorite_store_id": 999999, "sms_opt_in":  
true, "email_opt_in": true, "send_welcome_email": false, "email_validated":  
false, "addtl_info": {"name": "", "value": ""}}'  
https://webservice.skuped.com/mobile/gen/33559/v1/add
```

Returns

Returns the new spendgo member id if request succeeded. When the member's email has not been validated, a 202 status will be returned and the member is sent the email validation flow. Once validated the member is created. If the member already exists, a 409 response will be returned.

JSON response

spendgo_id string Member's Spendgo unique identifier.

username string Member's Spendgo username.

Example response (200):

```
{  
  "spendgo_id": "494979",  
  "username": "janesmith"  
}
```

Create/Retrieve a member

#server-to-server

Creates a new member associated to your account or retrieves the existing member using a phone number and/or email as the member's unique identifiers. A member phone number is the minimum requirement to create/retrieve a member.

When creating a new member and only the phone is posted to identify the account, a starter member (i.e. non-registered member) is created.

When creating a new member and the phone, email and password are provided, a registered member is created. If the email_verify is also set to true in the request, the customer will go through the email verification process.

HTTP request

POST <https://webservice.<ENVIRONMENT>.com/loyalty/accounts>

JSON body

customer	object	Optional.
phone	string	Required member's phone number.
email	string	Optional member's email address.
name	object	Optional.
first	string	Optional member's first name.
last	string	Optional member's last name.
password	string	Optional member's password in clear-text.
dob	string	Optional member's date of birth formatted as YYYY-MM-DD.
gender	string	Optional member's gender. Values are "m" or "f".
marital_status	string	Optional member's marital status. Values are "single", "married", "divorced", or "domestic partner".
address	object	Optional
street	string	Optional member's street address.

city string	Optional member's city address.
state string	Optional member's 2-letter state abbreviation address (e.g. CA).
zip string	Optional member's zip code address.
country string	Optional member's 2-letter country abbreviation (e.g. US).
sms_opt_in boolean	Optional member's text subscription preference for your brand. Once the user has validated their account creation, they will be sent the text double opt-in flow to confirm their text subscription.
email_opt_in boolean	Optional member's email marketing preference for your brand.
verify_email boolean	Default value is "false". Fires email verification autoresponder when set to true.

Example request (with phone only):

```
curl -k -v -X POST -H 'X-Class-Key: test_spendgo_loyalty' -H
'X-Class-Signature: EMTNgEMeW/E32z/hIFDn1tTVQ017GLC4SiBHNWwRx9w=' -H
'Content-Type: application/json' -d '{"customer":{"phone":"9999999999"}}'
https://webservice.skuped.com/loyalty/accounts
```

Example request (with email verification):

```
curl -k -v -X POST -H 'X-Class-Key: test_spendgo_loyalty' -H
'X-Class-Signature: yl6ScIoPUT27tPdJ6FcjA7wiR3oXCqAXSBGR3Cb3hTs=' -H
'Content-Type: application/json' -d
'{"customer":{"phone":"9999999999","email":"joe@example.com","name":{"last":
:"Smith","first":"Joe"},"password":"Password123","dob":"2000-01-01","gender
":"m","marital_status":"married","address":{"city":"San
Francisco","state":"CA","street":"3rd
Street","zip":"94107","country":"US"},"sms_opt_in":false,"email_opt_in":tru
e,"verify_email":true}}' https://webservice.skuped.com/loyalty/accounts
```

Returns

If the member already existed, a **200** success status is returned with their spendgo member id (**id**) and account.

If the member did not already exist, a **201** member created status is returned with the **id** and account status — the new member has been created.

If the member did not already exist and email validation is requested in the API call, a **202** status will be returned and the member is sent the email validation flow. Once validated the member is created in Spendgo.

When the email and phone combination do not match in our system, a **400** bad request response is returned informing you the email and phone already exist and are associated with different accounts.

JSON response

id string	Member's Spendgo unique identifier.
status string	Member's Spendgo account status. Value is "Active".

Example response (200):

```
{
  "id": "6522",
  "status": "Active"
}
```

Example response (202):

```
{
  "status": "WaitingEmailVerification"
}
```

Retrieve a member

[#authenticated](#) [#mobile](#) [#web](#)

Get any member's information after the authentication process is completed.

HTTP request

```
GET https://webservice.<ENVIRONMENT>.com/oauth2/member/v1/get
```

JSON body

None

Example request:

```
curl -k -v -X GET -H 'X-Class-Key: test_spendgo_oauth2' -H  
'X-Class-Signature: rXzoAFj5TaX2cF62a701upistZ5mEj909NNgiaauI3gh' -H  
'Content-Type: application/json' -H 'Authorization: Bearer  
aBcoAGk2Nag1cF98a957udpotZ3nPs808LLedapdI6xi'  
https://webservice.skuped.com/oauth2/member/v1/get
```

Returns

Returns member details after a valid authentication.

JSON response

id	string	A unique identifier for the object.
first_name	string	Member's first name.
last_name	string	Member's last name.
email	string	Member's email address.
mobile_phone	string	Member's phone number.

Example response (200):

```
{  
  "id": "123456",  
  "first_name": "Jane",
```

```
"last_name": "Smith",  
"email": "jane@example.com",  
"mobile_phone": "4155555555"  
}
```

Update a member

[#authenticated](#) [#mobile](#) [#web](#)

Modify the current signed in member and their preferences. At least the member's phone or spendgo_id are required as the unique reference in the request. Both phone and the spendgo_id cannot be updated.

HTTP request

POST https://webservice.<ENVIRONMENT>.com/mobile/gen/{account_id}/v1/update

JSON body

phone string	Required member's phone number.
spendgo_id string	Required member's Spendgo unique identifier.
email string	Optional member's email address.
password string	Required member's password.
first_name string	Optional member's first name.
last_name string	Optional member's last name.
dob string	Optional member's date of birth formatted as YYYY-MM-DD.
gender string	Optional member's gender. Values are "m" or "f".

marital_status	string	Optional member's marital status. Values are "single", "married", "divorced", or "domestic partner".
street	string	Optional member's street address.
city	string	Optional member's city address.
state	string	Optional member's 2-letter state abbreviation address (e.g. CA)
zip	string	Optional member's zip code address.
favorite_store_id	integer	Optional member's favorite store number
sms_opt_in	boolean	Optional member's text subscription preference for your brand. Once the user has validated their account creation, they will be sent the text double opt-in flow to confirm their text subscription.
email_opt_in	boolean	Optional member's email marketing preference for your brand.
research_opt_in	string	Optional
email_validated	boolean	Default value is "false".
additl_info	object	Optional
name	string	Optional
value	string	Optional

Example request:

```
curl -k -v -X GET -H 'X-Class-Key: test_spendgo_mobile' -H
'X-Class-Signature: rXzoAFj5TaX2cF62a701upistZ5mEj909NNgiaauI3gh' -H
'Content-Type: application/json' -H 'Authorization: Bearer
```

```
aBcoAGk2Nag1cF98a957udpotZ3nPs808LLedapdI6xi' -d '{"phone": "4155555555",
"spendgo_id": "494979", "email": "Jane@example.com", "password": "",
"first_name": "Jane", "last_name": "Smith", "dob": "2000-12-31", "gender":
"f", "marital_status": "", "street": "123 Street", "city": "San
Francisco", "state": "CA", "zip": 94107, "favorite_store_id": 999999,
"sms_opt_in": true, "email_opt_in": true, "send_welcome_email": false,
email_validated": false, "additl_info": {"name": "", "value": ""}}'
https://webservice.skuped.com/mobile/gen/33559/v1/add
```

Returns

Returns the new spendgo member id if request succeeded. When the member's email has not been validated, a 202 status will be returned and the member is sent the email validation flow. Once validated the member is created.

JSON response

spendgo_id string Member's Spendgo unique identifier.

username string Member's Spendgo username.

Example response (200):

```
{
  "spendgo_id": "494979",
  "username": "janesmith"
}
```

Retrieve member balance

#server-to-server

Get the member's points, visits, or stamps balance by spendgo member's unique id.

For rewards and offers, use [Retrieve member rewards by store](#).

HTTP request

GET

`https://webservice.<ENVIRONMENT>.com/loyalty/accounts/{spendgo_id}/balance`

- Where the `spendgo_id` is the id returned by the `/loyalty/accounts` call.

JSON body

None

Example request:

```
curl -k -v -X GET -H 'X-Class-Key: test_spendgo_loyalty' -H
'X-Class-Signature: WOHmRE4Cynx+i051135xk2bhRm/LJUfEstFBif2JFXU=' -H
'Content-Type: application/json'
https://webservice.skuped.com/loyalty/accounts/494979/balance
```

Returns

Returns the member's points balance.

JSON response

quantity integer	The number of units in the member's account.
units string	The unit measure of the program. Values are "points", "visits" or "stamps".
label string	The name of the reward.

Example response (200):

```
{
  "quantity": 25,
  "units": "points",
  "label": "$5 Off"
}
```


Retrieve member rewards by store

#server-to-server

Get the member's rewards that are available for redemption at a particular store. The rewards and offers returned may vary depending on the store and the offer rules. Some offers may expire after being returned once if defined in the rules to do so.

The store location identifier — or `store_code` — query parameter is required in the request to return the correct redeemable rewards for a member at the specified store. `store_code` is required to be synchronized in Spendgo.

Often used for brands with multiple store locations who use targeted offers by store location.

HTTP request

GET

```
https://webservice.<ENVIRONMENT>.com/loyalty/accounts/{spendgo_id}/rewards?store={store_code}
```

- Where the `spendgo_id` is the id returned in the `/loyalty/accounts` call.
- The `store_code` is the query parameter to retrieve redeemable rewards at that store location.

JSON body

None

Example request:

```
curl -k -v -X GET -H 'X-Class-Key: test_spendgo_loyalty' -H 'X-Class-Signature: 0tPDNqk+0MIhHIbMNxHbiX6OchbG8lHcGBpQ/Sa4AEE=' -H 'Content-Type: application/json' https://webservice.skuped.com/loyalty/accounts/494979/rewards?store=999999
```

Returns

List of rewards and their quantities available for redemption. If there are no available rewards, the rewards value will be an empty array. For many offers the `reward.value` is to be taken as a projected value, the value is validated in the Order flow and the final value will be returned at that time.

JSON response

rewards array	List of available rewards.
id string	A unique identifier for the object. When prefaced with a “P_” this indicates a Shop With Points program.
currency string	Optional three-letter ISO 4217 code.
offer string	Discount type. Values are “rebate”, “discount” or “gift”.
label string	Member facing reward name.
value decimal	Optional projected reward value.
reference object	Optional POS reference for processing the reward.
type string	Optional. Value is “promo”.
code string	Optional discount code linked to order processing.
quantity integer	The number of rewards that can be used.

Example response (200):

```
{
  "rewards": [
    {
      "id": "2234402",
      "currency": "USD",
      "offer": "rebate",
      "label": "$5 Off",
      "value": 5,
      "reference": {
        "type": "promo",
        "code": "50FF",
      }
    }
  ]
}
```

```
    }
    "quantity": 1
  }
]
}
```

Retrieve member balance and rewards

#mobile #web #authenticated

Get a summary of the member's reward balance (in points) and all their available rewards after the authentication process is completed.

HTTP request

POST
https://webservice.<ENVIRONMENT>.com/mobile/gen/{account_id}/v1/rewardsandoffers

JSON body

None

Example request:

```
curl -k -v -X POST -H 'X-Class-Key: test_spendgo_mobile' -H
'X-Class-Signature: rXzoAFj5TaX2cF62a701upistZ5mEj909NNgiaauI3gh' -H
'Content-Type: application/json' -H 'Authorization: Bearer
aBcoAGk2Nag1cF98a957udpotZ3nPs808LLedapdI6xi' -d '{"spendgo_id: 494979",
"account_id: 33559"}'
https://webservice.skuped.com/mobile/gen/33559/v1/rewardsandoffers
```

Returns

The member's reward balance (in points) and a list of rewards and their quantities available for redemption. If there are no available rewards, the rewards value will be an empty array. For many offers the "reward.value" is to be taken as a projected value, the value is validated in the Order flow and the final value will be returned at that time.

JSON response

point_total	integer	The member's total points balance.
spend_threshold	integer	The point threshold to earn a reward.
rewards_count	integer	Total count of the rewards and offers available to be redeemed.
rewards	array	List of available rewards.
id	string	A unique identifier for the object. When prefaced with a "P_" this indicates a Shop With Points program.
currency	string	Optional three-letter ISO 4217 code.
offer	string	Discount type. Values are "rebate", "discount" or "gift".
label	string	Member facing reward name.
value	integer	Optional projected reward value.
reference	array	Optional POS reference for processing the reward.
type	string	Optional. Value is "promo".
code	string	Optional discount code linked to order processing.
quantity	integer	The number of rewards that can be used.

Example response (200):

```
{
  "point_total": 50,
  "spend_threshold": 100,
  "rewards_count": 1,
```

```
    "rewards": [
      {
        "id": "1640",
        "offer": "rebate",
        "label": "$5 Off",
        "value": 5,
        "reference": {
          "type": "promo",
          "code": "50FF",
        },
        "quantity": 1
      }
    ]
  }
```

Sign in member

[#mobile](#) [#web](#) [#authenticated](#)

Logs a member into their account after the authentication process is completed.

HTTP request

POST https://webservice.<ENVIRONMENT>.com/mobile/gen/{account_id}/v1/signin

JSON body

value	string	Required member's email or phone number.
password	integer	The point threshold to earn a reward.
device_id	integer	Optional unique device ID to identify the device log in source.
os_name	string	Optional device operating system name.
os_version	string	Optional device operating system version.

manufacturer array Optional device manufacturer details.

timestamp string Optional UTC timestamp.

transaction_num string Optional.

expires string Optional UTC timestamp.

total integer Optional.

points integer Optional.

Example request:

```
curl -k -v -X POST -H 'X-Class-Key: test_spendgo_mobile'
-H 'X-Class-Signature: rXzoAFj5TaX2cF62a701upistZ5mEj909NNgiaauI3gh' -H
'Content-Type: application/json' -H 'Authorization: Bearer
aBcoAGk2Nag1cF98a957udpotZ3nPs808LLedapdI6xi' -d
'{"value":555555555,"password":"","device-id":"2fc4b5912826ad1","os_name":
"Android","os_version":"10.0","manufacturer":{"timestamp":"2020-01-01","tra
nsaction_num":"m","expires":2020-01-01T00:00:00,"total":10,"points":10}}'
https://webservice.skuped.com/mobile/gen/33559/v1/signin
```

Returns

Returns a token and `spendgo_id` on a 200 successful request. When the member is not found, a 404 response is returned. When the user id and/or password are invalid, a 401 response is returned.

JSON response

auth_token string

spendgo_id string Member's Spendgo unique identifier.

Example response (200):

```
{
```

```
"auth_token": "12a701upistZ5mEj9Z3nPs808LNgiaauI3ghLe3nPs",  
"spendgo_id": "494979"  
}
```

Sign out member

[#mobile](#) [#web](#) [#authenticated](#)

Logs a member out of their account after having been signed in.

HTTP request

```
POST  
https://webservice.<ENVIRONMENT>.com/mobile/gen/{account_id}/v1/signoff
```

JSON body

None

Example request:

```
curl -k -v -X POST -H 'X-Class-Key: test_spendgo_mobile'  
-H 'X-Class-Signature: rXzoAFj5TaX2cF62a701upistZ5mEj909NNgiaauI3gh' -H  
'Content-Type: application/json' -H 'Authorization: Bearer  
aBcoAGk2Nag1cF98a957udpotZ3nPs808LLedapdI6xi'  
https://webservice.skuped.com/mobile/gen/33559/v1/signoff
```

Returns

Returns an empty 200 successful sign out response.

JSON response

None

Example response (200):

```
{}
```

Reset member password

#mobile #web #authenticated

Triggers a reset password flow which sends an email with a reset password link to the member's account email address.

HTTP request

POST

`https://webservice.<ENVIRONMENT>.com/mobile/gen/{account_id}/v1/forgotpassword`

JSON body

email string Required member's email address.

program_name string Optional reward's program name for the account.

resetpassword_link string Optional custom URL for reset password site. The resetpassword_link with arguments - token is used in the email sent to the member. For example if the link is "https://www.mydomain.com/forgotpassword" then a HTTP get method with "https://www.mydomain.com/forgotpassword?token=xxxx" will be used in the email. The token argument passed can be used to call the resetpassword REST call.

Example request:

```
curl -k -v -X POST -H 'X-Class-Key: test_spendgo_mobile' -H  
'X-Class-Signature: rXzoAFj5TaX2cF62a701upistZ5mEj909NNgiaauI3gh' -H  
'Content-Type: application/json' -H 'Authorization: Bearer  
aBcoAGk2Nag1cF98a957udpotZ3nPs808LLedapdI6xi' -d  
'{"email": "janesmith@example.com"}'  
https://webservice.skuped.com/mobile/gen/33559/v1/forgotpassword
```

Returns

Returns an empty 200 response when successful. When the email is invalid or does not exist within Spendgo, a 400 bad request response is returned.

JSON response

None

Example response (200):

```
{}
```

Events

Orders

#server-to-server

Posts a member's order at the various stages of their ordering workflow. Each stage in the ordering workflow has a corresponding order status to be inserted in the request.

These are some of the common ordering workflows:

- [Add Transaction Using Signed Requests](#)
- [Add Transaction With User-Authentication \(OAuth2\)](#)

HTTP request

POST

`https://webservice.<ENVIRONMENT>.com/loyalty/accounts/{spendgo_id}/orders`

- Where the `spendgo_id` is the id returned in the `/loyalty/accounts` call.
- When the `spendgo_id` is "guest" (i.e. `.../loyalty/accounts/guest/orders`) this is equivalent to an order made by a non-member or guest.

JSON body

id `string` Optional order id generated by the commerce system. Required at or after the "placed" order status.

source `string` Required order source. Values are "web" for desktop, "mobileweb" for mobile web, "mobileapp" for mobile apps, or "instore" for in-store orders.

transaction string Optional when “refunded” status, this is an external reference sent from Spendgo at the “billed” or “completed” status.

status string Required order status. Values are “checkout”, “placed”, “billed”, “completed”, “voided”, and “refunded”.

type string Optional order type. Values are “regular” or “catering”.

rendition string Optional. Values are “onsite”, “pickup”, or “delivery”.

currency string Optional three-letter ISO 4217 code.

placed string Optional date-time as defined in RFC3339. This is when the order was created, but does not indicate when the order was paid or when the customer receives the order. For example the order is placed on 3/1/2020 @ 10am to be picked up on 3/3/2020 @ noon.

wanted string Optional date-time as defined in RFC3339. This represents the time when the customer receives the order. For example, the order is placed on 3/1/2018 @ 10am to be picked up on 3/3/2018 @ noon.

vendor string Required store code.

brand string Required brand’s unique account_id in Spendgo.

subtotal decimal Required order subtotal — order cost before applying tax, tip, delivery, and discount.

tax decimal Optional order tax amount.

tip decimal Optional order tip amount.

delivery decimal Optional delivery cost.

discount decimal Optional discount amount.

total	decimal	Optional order total — the sum of subtotal, delivery, tip, tax, and discounts.
address	array	Optional customer's delivery address.
street	string	Optional street address.
cross	string	Optional cross street.
city	string	Optional city.
code	string	Optional zip code.
region	string	Optional region (a two-letter ISO 3166-2 code).
country	string	Optional country (a three-letter ISO 3166-1 code).
payments	array	Optional payment tender(s). Required for subsequent order states after "checkout".
tender	string	Optional payment method. Values are "cash", "check", "credit", "debit", "prepaid", "transfer", or "value".
issuer	string	Optional payment vendor. Values are "amex", "diners", "discover", "jcb", "mastercard", "paypal", or "visa".
suffix	string	Optional payment last four digits.
amount	integer	Optional amount allocated to this payment method.
basket	array	Required order details.
uuid	string	Optional ordering transaction identifier.

rewards	array	Optional. Corresponds to the reward discounts on the order.
provider	string	Optional. When redeeming a reward, the required value is "Spendgo".
id	string	Optional. When redeeming a reward, the required value is the corresponding id returned in the Retrieve member rewards by store request.
level	string	Optional. When redeeming a reward, the value is "basket".
product	string	Optional product item SKU.
discount	decimal	Optional. This is required when redeeming a reward. When the order status is "checkout", the value is the reward.value returned in the Retrieve member rewards by store request. The response will include the validated discount value — you will use the validated value in subsequent order calls.
posEntries	array	Required order items.
posItem	array	Required item details.
product	string	Optional item product identifier.
modifiers	array	Optional modifications or add-ons to an item.
product	string	Optional modifier product identifier.
modifiers	array	Optional modifications or add-ons to a modifier.
label	string	Optional item modifier name. Required when a modifier is present.

cost decimal	Optional item modifier cost value. Required when a modifier is present.
categories array	Optional modifier categories.
quantity integer	Optional quantity of modifier added to item. Required when a modifier is present.
label string	Required item name.
cost decimal	Required item cost value.
categories string	Optional item categories.
quantity integer	Required item quantity added to order.

Example request:

```
curl -k -v -X POST -H 'X-Class-Key: test_spendgo_loyalty' -H
'X-Class-Signature: HG4+uMDAjbPESYM0ISfHdPzALQQ7Fd2W141MVJATmbY=' -H
'Content-Type: application/json' -d
'{"id":"988654417","source":"web","transaction":null,"status":"checkout","t
ype":"regular","rendition":"delivery","currency":null,"placed":"2020-01-01T
12:00:00Z","wanted":"2020-01-01T13:00:00Z","vendor":"999999","brand":"33559
","subtotal":10,"tax":0.1,"tip":0,"delivery":0,"discount":5,"total":5.1,"ad
dress": [{"street":"123 Street","cross":"","city":"San Francisco",
"region":"CA","code":94107,"country":"USA"}],
"payments":{"tender":"credit","issuer":"visa","suffix":"1234","amount":5.1}
,"basket":{"uuid":"472b564f-ae72-49ac-91cb-f177e21216f8",
"rewards":[{"provider":"Spendgo","id":"2234402","level":"basket","product":
"0","discount":5}], "posEntries":[{"posItem":{"product":"1082","cost":5,"cat
egories":null,"label":"Turkey Sandwich"},"quantity":2}]}'
https://webservice.skuped.com/loyalty/accounts/494979/orders
```

Returns

When the order status is checkout, the response returns the final reward discount value that has been validated against the items within the order submitted. In most cases you will use this final value as the reward value to be discounted from the order. Only a [Shop With](#)

[Points](#) reward — identified by a reward.id prefixed with a “P_” — can be equal to or less than the returned value because the member can redeem all or some of the value.

When the order status is billed, the successful response returns the transaction.id and status.

JSON response

transaction array
id string Optional order identifier created in status “billed” order.
discount decimal The validated reward discount value.
status string The status of the order that was passed in the request, will be returned in the response.
message string Optional. If there is an error applying a reward to the order in the checkout stage, this will contain a customer-facing error message.

Example response (200):

```
{
  "transaction": {
    "discount": 5.00,
    "status": "checkout",
  }
}
```