

# OpenCart API Doc

## Overview

To use the OpenCart API, enable it in the **Admin** panel:

1. Navigate to **System** → **Users** → **API**.
  2. You will see a predefined user named “**Default.**”
  3. Edit the user, generate an API key, and set **Status = Enabled**.
  4. Add your IP address to the list of allowed IPs.
- 
- 

## 1. Authentication

**Endpoint:** `api/login`

**Method :** POST

**Body**(`x-www-form-urlencoded`):

Key	Description
<b>username</b>	API username configured in admin
<b>key</b>	API key generated for the API user

**Response:**

```
{
  "success": "Success: API session successfully started!",
  "api_token": "your_generated_token"
}
```

---

## 2. Currency

**Endpoint:** `api/currency`

**Method :** POST

**Body**(`x-www-form-urlencoded`):

Key	Description
<b>currency</b>	Currency code (e.g., USD, EUR)

#### Response:

```
{
  "success": "Success: Your currency has been changed!"
}
```

---

## 3. Cart

### Add Product

**Endpoint:** `api/cart/add`

**Method :** POST

**Body**(`x-www-form-urlencoded`):

Key	Description
<b>product_id</b>	The ID of the product to add
<b>quantity</b>	Quantity (optional, default = 1)

#### Response:

```
{
  "success": "Success: You have modified your shopping cart!"
}
```

---

### Edit Quantity

**Endpoint:** `api/cart/edit`

**Method :** POST

**Body**(`x-www-form-urlencoded`):

Key	Description
<b>key</b>	The cart ID for the product got from <b>api/cart/products</b>
<b>quantity</b>	New quantity value

#### Response:

```
{
  "success": "Success: You have modified your shopping cart!"
}
```

---

## Remove Product

**Endpoint:** `api/cart/remove`

**Method :** POST

**Body**(`x-www-form-urlencoded`):

Key	Description
key	The cart ID for the product got from <b>api/cart/products</b>

**Response:**

```
{
  "success": "Success: You have modified your shopping cart!"
}
```

---

## View Cart

**Endpoint:** `api/cart/products`

**Method :** GET

**Response Example:**

```
{
  "products": [
    {
      "cart_id": "15",
      "product_id": "43",
      "name": "MacBook",
      "model": "Product 16",
      "option": [],
      "quantity": "1",
      "stock": true,
      "shipping": "0",
      "price": "$602.00",
      "total": "$602.00",
    }
  ]
}
```

```

        "reward": 600
      }
    ],
    "vouchers": [],
    "totals": [
      {
        "title": "Sub-Total",
        "text": "$500.00"
      },
      {
        "title": "Eco Tax (-2.00)",
        "text": "$2.00"
      },
      {
        "title": "VAT (20%)",
        "text": "$100.00"
      },
      {
        "title": "Total",
        "text": "$602.00"
      }
    ]
  ]
}

```

## 4. Coupon

**Endpoint:** api/coupon

**Method :** POST

**Body**(x-www-form-urlencoded):

Key	Description
coupon	Coupon code to apply (You got from Admin)

**Response:**

```

{
  "success": "Success: Your coupon discount has been applied!"
}

```

## 5. Customer

**Endpoint:** `api/customer`

**Method :** POST

**Body**(`x-www-form-urlencoded`):

Key	Description
<b>firstname</b>	Customer first name
<b>lastname</b>	Customer last name
<b>email</b>	Customer email address
<b>telephone</b>	Contact number

**Response:**

```
{
  "success": "Success: Customer details have been set!"
}
```

---

## 6. Voucher

### Apply Existing Voucher

**Endpoint:** `api/voucher`

**Method :** POST

**Body**(`x-www-form-urlencoded`):

Key	Description
<b>voucher</b>	Existing voucher code to apply (Get the code from admin panel)

**Response:**

```
{
  "success": "Success: Your gift certificate discount has been applied!"
}
```

---

### Add New Voucher to your cart

**Endpoint:** `api/voucher/add`

**Method :** POST

**Body**(x-www-form-urlencoded):

Key	Description
from_name	Sender name
from_email	Sender email
to_name	Recipient name
to_email	Recipient email
amount	Voucher amount

**Response:**

```
{
  "success": "Success: You have modified your shopping cart!"
}
```

---

## 7. Shipping

### Set Shipping Address

**Endpoint:** api/shipping/address

**Method :** POST

**Body**(x-www-form-urlencoded):

Key	Description
firstname	First name
lastname	Last name
address_1	Address line 1
city	City name
postcode	Postal code
country_id	Country ID
zone_id	Zone/State ID

**Response:**

```
{
  "success": "Success: Shipping address has been set!"
}
```

---

### Get Shipping Methods

**Endpoint:** api/shipping/methods

**Method :** POST

**Response Example:**

```
{
  "shipping_methods": {
    "flat": {
      "title": "Flat Rate",
      "quote": {
        "flat": {
          "code": "flat.flat",
          "title": "Flat Shipping Rate",
          "cost": "5.00",
          "tax_class_id": "9",
          "text": "$8.00"
        }
      },
      "sort_order": "1",
      "error": false
    }
  }
}
```

---

## Set Shipping Method

**Endpoint:** api/shipping/method

**Method :** POST

**Body**(x-www-form-urlencoded):

Key	Description
shipping_method	Shipping method code (e.g., flat.flat)

**Response:**

```
{
  "success": "Success: Shipping method has been set!"
}
```

---

## 8. Payment

### Set Payment Address

**Endpoint:** api/payment/address

**Method :** POST

**Body**(x-www-form-urlencoded):

Key	Description
firstname	First name
lastname	Last name
address_1	Address line 1
city	City name
country_id	Country ID
postcode	Post code
zone_id	Zone/State ID

**Response:**

```
{
  "success": "Success: Payment address has been set!"
}
```

---

## Get Payment Methods

**Endpoint:** api/payment/methods

**Method :** POST

**Response Example:**

```
{
  "cod": {
    "code": "cod",
    "title": "Cash On Delivery",
    "terms": "",
    "sort_order": ""
  }
}
```

---

## Set Payment Method

**Endpoint:** api/payment/method

**Method :** POST

**Body**(x-www-form-urlencoded):

Key	Description
-----	-------------



**payment\_method**

Payment method code (e.g., cod)

**Response:**

```
{
  "success": "Success: Payment method has been set!"
}
```

---

## 9. Order

### Create Order

**Endpoint:** `api/order/add`

**Method :** POST

**Response:**

```
{
  "success": "Success: You have modified orders!",
  "order_id": 1
}
```

---

### Get Order Info

**Endpoints:**

- `api/order/info`
- `/api/order/edit`
- `/api/order/delete`
- `/api/order/history`

Information are missing from official Documentation

---