

NFT Market v0.0.1

Scroll down for code samples, example requests and responses. Select a language for code samples from the tabs above or the mobile navigation menu.

NFT Market API

Base URLs:

- <http://localhost/api/master/v0>

Authentication

- API Key (BaseApiKey)
 - Parameter Name: **apikey**, in: header.

base

base api define

get_base_ping

Code samples

```
const headers = {
  'Accept': 'application/json'
};

fetch('http://localhost/api/master/v0/base/ping',
{
  method: 'GET',

  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});
```

GET /base/ping

伺服器作動檢查

檢查伺服器是否有回應，並取得伺服器時間戳。

Parameters

Name	In	Type	Required	Description
now	query	integer	false	本地時間戳

Example responses

200 Response

0

Responses

Status	Meaning	Description	Schema
200	OK	OK	integer
default	Default	Error response	BaseError

This operation does not require authentication

cart

NFT 購物/上架/匯入車 相關操作

cartShoppingLoad

Code samples

```
const inputBody = `{
  "userId": "string"
}`;
const headers = {
  'Content-Type':'application/json',
  'Accept':'application/json',
  'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/cart/shopping/load',
{
  method: 'POST',
  body: inputBody,
  headers: headers
}
```

```
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

POST /cart/shopping/load
```

取得購物車資訊

取得購物車資訊

Body parameter

```
{
  "userId": "string"
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	user check
» userId	body	string	true	none

Example responses

200 Response

```
{
  "hash": "string",
  "userId": "string",
  "list": [
    {
      "chainId": 0,
      "contract": "string",
      "collection": "string",
      "id": 0,
      "nftName": "string",
      "shop": "string",
      "currency": "string",
      "price": 0,
      "enable": true
    }
  ],
  "cost": [
    {
      "id": 0,
      "name": "string",
      "value": 0
    }
  ]
}
```

```

    "type": "string",
    "currency": "string",
    "amount": 0
  }
],
"wallet": [
  {
    "currency": "string",
    "balance": 0,
    "enable": true,
    "willPay": 0
  }
],
"payment": [
  {
    "currency": "string",
    "amount": 0,
    "payBy": [
      {
        "currency": "string",
        "used": 0,
        "rate": 0,
        "amount": 0
      }
    ]
  }
]
}

```

Responses

Status	Meaning	Description	Schema
200	OK	OK	CartShoppingModel
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

cartShoppingSave

Code samples

```
const inputBody = `{
  "userId": "string",
  "list": [
    {
      "chainId": 0,
      "contract": "string",
      "id": 0,
      "shop": "string"
    }
  ],
  "wallet": [
    {
      "currency": "string",
      "enable": true,
      "rate": 0
    }
  ]
}`;
const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json',
  'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/cart/shopping/save',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

POST /cart/shopping/save
```

更新購物車資訊

更新購物車資訊

Body parameter

```
{  
  "userId": "string",  
  "list": [  
    {  
      "chainId": 0,  
      "contract": "string",  
      "id": 0,  
      "shop": "string"  
    }  
  ],  
  "wallet": [  
    {  
      "currency": "string",  
      "enable": true,  
      "rate": 0  
    }  
  ]  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	user check
» userId	body	string	true	none
» list	body	[object]	false	none
»» chainId	body	integer	false	none
»» contract	body	string	false	none
»» id	body	integer	false	none
»» shop	body	string	false	none
» wallet	body	[object]	false	none
»» currency	body	string	false	none
»» enable	body	boolean	false	none
»» rate	body	number	false	none

Example responses

200 Response

```
{  
  "hash": "string",  
  "userId": "string",  
  "list": [  
    {  
      "chainId": 0,  
      "contract": "string",  
      "collection": "string",  
      "id": 0,  
      "nftName": "string",  
      "shop": "string",  
      "currency": "string",  
      "price": 0,  
      "enable": true  
    }  
,  
  ],  
  "cost": [  
    {  
      "type": "string",  
      "currency": "string",  
      "amount": 0  
    }  
,  
  ],  
  "wallet": [  
    {  
      "currency": "string",  
      "balance": 0,  
      "enable": true,  
      "willPay": 0  
    }  
,  
  ],  
  "payment": [  
    {  
      "currency": "string",  
      "amount": 0,  
      "payBy": [  
        {  
          "currency": "string",  
          "used": 0,  
          "rate": 0,  
          "amount": 0  
        }  
      ]  
    }  
,  
  ]  
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	CartShoppingModel
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» hash	string	false	none	none
» userId	string	false	none	none
» list	string	false	none	none
» wallet	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

cartSellingLoad

Code samples

```
const inputBody = `{
  "userId": "string"
}`;
const headers = {
  'Content-Type':'application/json',
  'Accept':'application/json',
  'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/cart/selling/load',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
```

```
    console.log(body);
});
```

POST /cart/selling/load

取得上架車資訊

取得上架車資訊

Body parameter

```
{
  "userId": "string"
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	user check
» userId	body	string	true	none

Example responses

200 Response

```
{
  "hash": "string",
  "userId": "string",
  "list": [
    {
      "chainId": 0,
      "contract": "string",
      "collection": "string",
      "id": 0,
      "nftName": "string",
      "shops": [
        {
          "shop": "string",
          "currency": "string",
          "price": 0,
          "enable": true
        }
      ]
    }
  ],
  "cost": [
    {
      "cost": 0
    }
  ]
}
```

```

    "type": "string",
    "currency": "string",
    "amount": 0
  }
],
"wallet": [
  {
    "currency": "string",
    "balance": 0,
    "enable": true,
    "willPay": 0
  }
],
"payment": [
  {
    "currency": "string",
    "amount": 0,
    "payBy": [
      {
        "currency": "string",
        "used": 0,
        "rate": 0,
        "amount": 0
      }
    ]
  }
]
}

```

Responses

Status	Meaning	Description	Schema
200	OK	OK	CartSellingModel
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

cartSellingSave

Code samples

```
const inputBody = `{
  "userId": "string",
  "list": [
    {
      "chainId": 0,
      "contract": "string",
      "id": 0,
      "shop": "string"
    }
  ],
  "wallet": [
    {
      "currency": "string",
      "enable": true,
      "rate": 0
    }
  ]
}`;
const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json',
  'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/cart/selling/save',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

POST /cart/selling/save
```

更新上架車資訊

更新上架車資訊

Body parameter

```
{  
  "userId": "string",  
  "list": [  
    {  
      "chainId": 0,  
      "contract": "string",  
      "id": 0,  
      "shop": "string"  
    }  
  ],  
  "wallet": [  
    {  
      "currency": "string",  
      "enable": true,  
      "rate": 0  
    }  
  ]  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	user check
» userId	body	string	true	none
» list	body	[object]	false	none
»» chainId	body	integer	false	none
»» contract	body	string	false	none
»» id	body	integer	false	none
»» shop	body	string	false	none
» wallet	body	[object]	false	none
»» currency	body	string	false	none
»» enable	body	boolean	false	none
»» rate	body	number	false	none

Example responses

200 Response

```
{
  "hash": "string",
  "userId": "string",
  "list": [
    {
      "chainId": 0,
      "contract": "string",
      "collection": "string",
      "id": 0,
      "nftName": "string",
      "shops": [
        {
          "shop": "string",
          "currency": "string",
          "price": 0,
          "enable": true
        }
      ]
    }
  ],
  "cost": [
    {
      "type": "string",
      "currency": "string",
      "amount": 0
    }
  ],
  "wallet": [
    {
      "currency": "string",
      "balance": 0,
      "enable": true,
      "willPay": 0
    }
  ],
  "payment": [
    {
      "currency": "string",
      "amount": 0,
      "payBy": [
        {
          "currency": "string",
          "used": 0,
          "rate": 0,
          "amount": 0
        }
      ]
    }
  ]
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	CartSellingModel
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» hash	string	false	none	none
» userId	string	false	none	none
» list	string	false	none	none
» wallet	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

cartExportCheck

Code samples

```
const inputBody = '{
  "userId": "string",
  "chainId": 0,
  "list": [
    {
      "contract": "string",
      "id": 0
    }
  ],
  "receiver": "string",
  "sign": "string",
  "wallet": [
    {
      "currency": "string",
      "enable": true
    }
]
```

```
]
};

const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json',
  'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/cart/export/check',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});


```

POST /cart/export/check

檢查匯出車內品項及計算匯出手續費

檢查匯出車內品項及計算匯出手續費

Body parameter

```
{
  "userId": "string",
  "chainId": 0,
  "list": [
    {
      "contract": "string",
      "id": 0
    }
  ],
  "receiver": "string",
  "sign": "string",
  "wallet": [
    {
      "currency": "string",
      "enable": true
    }
  ]
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	user check
» userId	body	string	true	none
» chainId	body	integer	false	none
» list	body	[object]	false	none
»» contract	body	string	false	contract address
»» id	body	integer	false	nft id
» receiver	body	string	false	收件人錢包帳號
» sign	body	string	false	eip712 簽名證明為錢包持有人
» wallet	body	[object]	false	none
»» currency	body	string	false	none
»» enable	body	boolean	false	none

Example responses

200 Response

```
{
  "userId": "string",
  "chainId": 0,
  "list": [
    {
      "contract": "string",
      "id": 0,
      "transfer": "string"
    }
  ],
  "cost": [
    {
      "type": "string",
      "currency": "string",
      "amount": 0
    }
  ],
  "wallet": [
    {
      "currency": "string",
      "balance": 0,
      "enable": true,
      "willPay": 0
    }
  ]
}
```

```

],
"payment": [
{
  "currency": "string",
  "amount": 0,
  "payBy": [
    {
      "currency": "string",
      "used": 0,
      "rate": 0,
      "amount": 0
    }
  ]
}
]
}

```

Responses

Status	Meaning	Description	Schema
200	OK	OK	Inline
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **200**

Name	Type	Required	Restrictions	Description
» userId	string	true	none	none
» chainId	integer	false	none	none
» list	[object]	false	none	none
»» contract	string	false	none	contract address
»» id	integer	false	none	nft id
»» transfer	string	false	none	目標匯入帳號地址
» cost	[object]	false	none	none
»» type	string	false	none	gas / price / fee
»» currency	string	false	none	none

Name	Type	Required	Restrictions	Description
»» amount	number	false	none	none
» wallet	[object]	false	none	none
»» currency	string	false	none	none
»» balance	number	false	none	none
»» enable	boolean	false	none	none
»» willPay	number	false	none	none
» payment	[object]	false	none	none
»» currency	string	false	none	none
»» amount	number	false	none	none
»» payBy	[object]	false	none	none
»»» currency	string	false	none	none
»»» used	number	false	none	none
»»» rate	number	false	none	none
»»» amount	number	false	none	none

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» chainId	string	false	none	none
» userId	string	false	none	none
» list	string	false	none	none
» sender	string	false	none	none
» sign	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

cartImportRegister

Code samples

```

const inputBody = `{
  "userId": "string",
  "chainId": 0,
  "list": [
    {
      "contract": "string",
      "id": 0
    }
  ],
  "sender": "string",
  "sign": "string"
}`;
const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json',
  'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/cart/import/register',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

```

POST /cart/import/register

註冊NFT匯入並取得匯入錢包帳號

註冊NFT匯入並取得匯入錢包帳號

Body parameter

```
{
  "userId": "string",
  "chainId": 0,
  "list": [
    {
      "contract": "string",
      "id": 0
    }
  ],
  "sender": "string",
  "sign": "string"
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	user check
» userId	body	string	true	none
» chainId	body	integer	false	none
» list	body	[object]	false	none
»» contract	body	string	false	contract address
»» id	body	integer	false	nft id
» sender	body	string	false	匯入錢包帳號
» sign	body	string	false	eip712 簽名證明為錢包持有人

Example responses

200 Response

```
{  
  "userId": "string",  
  "chainId": 0,  
  "list": [  
    {  
      "contract": "string",  
      "id": 0,  
      "transfer": "string"  
    }  
  ]  
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	Inline
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code 200

Name	Type	Required	Restrictions	Description
» userId	string	true	none	none
» chainId	integer	false	none	none
» list	[object]	false	none	none
»» contract	string	false	none	contract address
»» id	integer	false	none	nft id
»» transfer	string	false	none	目標匯入帳號地址

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» chainId	string	false	none	none
» userId	string	false	none	none
» list	string	false	none	none
» sender	string	false	none	none
» sign	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

catagory

NFT 集合查詢

catagoryQuery

Code samples

```
const inputBody = `{
  "keyword": "string",
  "catagory": "string",
  "offset": 0,
  "limit": 0
}`;
const headers = {
  'Content-Type': 'application/json',
}
```

```

'Accept': 'application/json'
};

fetch('http://localhost/api/master/v0/catagory/query',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

```

POST /catagory/query

查詢/搜尋NFT集合清單.

空查詢時各分類導入固定數量,
 關鍵字回傳固定使用名稱排序.
 關鍵字查詢使用 Moralis.
 分類索引使用系統資料庫.

Body parameter

```
{
  "keyword": "string",
  "catagory": "string",
  "offset": 0,
  "limit": 0
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	none
» keyword	body	string	false	關鍵字查詢
» catagory	body	string	false	分類查詢
» offset	body	integer	false	起始位置
» limit	body	integer	false	單頁數量

Example responses

200 Response

```
{
  "total": 0,
  "offset": 0,
  "list": [
    {
      "nftId": "string",
      "imgUrl": "string",
      "name": "string",
      "catagory": "catagory.hot",
      "chainId": 0,
      "contract": "string"
    }
  ]
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	Inline
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **200**

NFT 集合回傳物件

Name	Type	Required	Restrictions	Description
» total	integer	false	none	查詢結果總數
» offset	integer	false	none	回傳起始位置
» list	[object]	false	none	none
»» nftId	string	false	none	資料庫 NFT ID
»» imgUrl	string	false	none	none
»» name	string	false	none	none
»» catagory	string	false	none	none
»» chainId	integer	false	none	none
»» contract	string	false	none	NFT 合約地址

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» category	string	false	none	none
» keyword	string	false	none	none
» offset	string	false	none	none

This operation does not require authentication

collection

NFT 系列查詢

collectionQuery

Code samples

```
const inputBody = '{\n    "chainId": 0,\n    "contract": "string",\n    "offset": 0,\n    "limit": 0,\n    "filter": {\n        "idStart": 0,\n        "idEnd": 0,\n        "rarityStart": 0,\n        "rarityEnd": 0,\n        "buyNow": true,\n        "shops": [\n            "string"\n        ],\n        "attributes": [\n            {\n                "key": "string",\n                "value": "string"\n            }\n        ]\n    },\n    "sort": {\n        "price": "string",\n        "rarity": "string",\n        "id": "string"\n    }\n}'
```

```
};

const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json'
};

fetch('http://localhost/api/master/v0/collection/query',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});


```

POST /collection/query

查詢/搜尋NFT系列清單.

查詢NFT集合

Body parameter

```
{
  "chainId": 0,
  "contract": "string",
  "offset": 0,
  "limit": 0,
  "filter": {
    "idStart": 0,
    "idEnd": 0,
    "rarityStart": 0,
    "rarityEnd": 0,
    "buyNow": true,
    "shops": [
      "string"
    ],
    "attributes": [
      {
        "key": "string",
        "value": "string"
      }
    ]
  },
  "sort": {
    "price": "string",
    "rarity": "string",
    "id": "string"
  }
}
```

```
    }  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	none
» chainId	body	integer	false	公鏈ID
» contract	body	string	false	NFT合約地址
» offset	body	integer	false	起始位置
» limit	body	integer	false	單頁數量
» filter	body	object	false	none
»» idStart	body	integer	false	none
»» idEnd	body	integer	false	none
»» rarityStart	body	integer	false	none
»» rarityEnd	body	integer	false	none
»» buyNow	body	boolean	false	none
»» shops	body	[string]	false	none
»» attributes	body	[object]	false	none
»»» key	body	string	false	none
»»» value	body	string	false	none
» sort	body	object	false	none
»» price	body	string	false	asc
»» rarity	body	string	false	asc
»» id	body	string	false	asc

Example responses

200 Response

```
{  
  "total": 0,  
  "offset": 0,  
  "rarity": [
```

```

{
  "key": "string",
  "values": [
    "string"
  ]
}
],
"idMin": 0,
"idMax": 0,
"rarityMin": 0,
"rarityMax": 0,
"totalSupply": 0,
"floorPrice": 0,
"chainId": 0,
"contract": "string",
"name": "string",
"symbol": "string",
"description": "string",
"chainOn": 0,
"nftId": "string",
"catagories": [
  "string"
],
"list": [
  {
    "id": 0,
    "rarity": 0,
    "attrubutes": [
      {
        "key": "string",
        "value": "string"
      }
    ],
    "imgUrl": "string",
    "name": "string",
    "shops": [
      {
        "buyNow": true,
        "currency": "string",
        "price": 0
      }
    ]
  }
]
}

```

Responses

Status	Meaning	Description	Schema
--------	---------	-------------	--------

Status	Meaning	Description	Schema
200	OK	OK	Inline
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **200**

NFT 系列回傳物件

Name	Type	Required	Restrictions	Description
» total	integer	false	none	查詢結果總數
» offset	integer	false	none	回傳起始位置
» rarity	[object]	false	none	稀有度資料
»» key	string	false	none	none
»» values	[string]	false	none	none
» idMin	integer	false	none	none
» idMax	integer	false	none	none
» rarityMin	integer	false	none	none
» rarityMax	integer	false	none	none
» totalSupply	integer	false	none	none
» floorPrice	number	false	none	none
» chainId	integer	false	none	none
» contract	string	false	none	none
» name	string	false	none	none
» symbol	string	false	none	none
» description	string	false	none	none
» chainOn	integer	false	none	部署上鏈時間
» nftId	string	false	none	資料庫 NFT ID
» catagories	[string]	false	none	none

Name	Type	Required	Restrictions	Description
» list	[object]	false	none	none
»» id	integer	false	none	none
»» rarity	integer	false	none	none
»» attributes	[object]	false	none	none
»»» key	string	false	none	none
»»» value	string	false	none	none
»» imgUrl	string	false	none	none
»» name	string	false	none	none
»» shops	[object]	false	none	none
»»» buyNow	boolean	false	none	none
»»» currency	string	false	none	none
»»» price	number	false	none	none

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» chainId	string	false	none	none
» contract	string	false	none	none
» offset	string	false	none	none
» limit	string	false	none	none
» filter	string	false	none	none
» sort	string	false	none	none

This operation does not require authentication

collectionMember

Code samples

```
const inputBody = `{
  "userId": "string",
```

```

"offset": 0,
"limit": 0,
"filter": {
  "chainId": 0,
  "contract": "string"
}
};

const headers = {
  'Content-Type':'application/json',
  'Accept':'application/json'
};

fetch('http://localhost/api/master/v0/collection/member',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

```

POST /collection/member

查詢我的NFT清單.

查詢我的NFT集合

Body parameter

```
{
  "userId": "string",
  "offset": 0,
  "limit": 0,
  "filter": {
    "chainId": 0,
    "contract": "string"
  }
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	none
» userId	body	string	true	member ID

Name	In	Type	Required	Description
» offset	body	integer	false	起始位置
» limit	body	integer	false	單頁數量
» filter	body	object	false	none
»» chainId	body	integer	false	none
»» contract	body	string	false	none

Example responses

200 Response

```
{
  "userId": "string",
  "chains": [
    {
      "chainId": 0,
      "contract": "string",
      "name": "string"
    }
  ],
  "total": 0,
  "offset": 0,
  "list": [
    {
      "id": 0,
      "rarity": 0,
      "attrubutes": [
        {
          "key": "string",
          "value": "string"
        }
      ],
      "imgUrl": "string",
      "name": "string",
      "shops": [
        {
          "currency": "string",
          "price": 0
        }
      ],
      "chainId": 0,
      "contract": "string",
      "collection": "string",
      "symbol": "string"
    }
  ]
}
```

```
]  
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	Inline
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **200**

NFT 系列回傳物件

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none
» chains	[object]	false	none	公鏈及NFT合約地址
»» chainId	integer	false	none	none
»» contract	string	false	none	none
»» name	string	false	none	none
» total	integer	false	none	查詢結果總數
» offset	integer	false	none	回傳起始位置
» list	[object]	false	none	none
»» id	integer	false	none	none
»» rarity	integer	false	none	none
»» attributes	[object]	false	none	none
»»» key	string	false	none	none
»»» value	string	false	none	none
»» imgUrl	string	false	none	none
»» name	string	false	none	none
»» shops	[object]	false	none	none

Name	Type	Required	Restrictions	Description
»»» currency	string	false	none	none
»»» price	number	false	none	none
»» chainId	integer	false	none	none
»» contract	string	false	none	none
»» collection	string	false	none	none
»» symbol	string	false	none	none

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none
» offset	string	false	none	none
» filter	string	false	none	none

This operation does not require authentication

collectionSelling

Code samples

```
const inputBody = `{
  "userId": "string",
  "offset": 0,
  "limit": 0,
  "filter": {
    "chainId": 0,
    "contract": "string",
    "shops": "string"
  }
}`;
const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json'
};

fetch('http://localhost/api/master/v0/collection/selling',
{
  method: 'POST',
  body: inputBody
});
```

```
body: inputBody,
headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});
});
```

POST /collection/selling

查詢已上架NFT清單.

查詢已上架NFT集合

Body parameter

```
{
  "userId": "string",
  "offset": 0,
  "limit": 0,
  "filter": {
    "chainId": 0,
    "contract": "string",
    "shops": "string"
  }
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	none
» userId	body	string	true	member ID
» offset	body	integer	false	起始位置
» limit	body	integer	false	單頁數量
» filter	body	object	false	none
»» chainId	body	integer	false	none
»» contract	body	string	false	none
»» shops	body	string	false	none

Example responses

200 Response

```
{
  "userId": "string",
  "chains": [
    {
      "chainId": 0,
      "contract": "string",
      "name": "string"
    }
  ],
  "shops": [
    "string"
  ],
  "total": 0,
  "offset": 0,
  "list": [
    {
      "id": 0,
      "rarity": 0,
      "imgUrl": "string",
      "name": "string",
      "shop": "string",
      "chainId": 0,
      "contract": "string",
      "collection": "string",
      "symbol": "string",
      "invoiceId": "string"
    }
  ]
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	Inline
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **200**

NFT 系列回傳物件

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none

Name	Type	Required	Restrictions	Description
» chains	[object]	false	none	公鏈及NFT合約地址
»» chainId	integer	false	none	none
»» contract	string	false	none	none
»» name	string	false	none	none
» shops	[string]	false	none	none
» total	integer	false	none	查詢結果總數
» offset	integer	false	none	回傳起始位置
» list	[object]	false	none	none
»» id	integer	false	none	none
»» rarity	integer	false	none	none
»» imgUrl	string	false	none	none
»» name	string	false	none	none
»» shop	string	false	none	none
»» chainId	integer	false	none	none
»» contract	string	false	none	none
»» collection	string	false	none	none
»» symbol	string	false	none	none
»» invoiceId	string	false	none	none

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none
» offset	string	false	none	none
» filter	string	false	none	none

This operation does not require authentication

guest

未登入使用者操作界面

guestRegisterByPhone

Code samples

```
const inputBody = '  
  "email": "string",  
  "emailVerify": "string",  
  "phone": "string",  
  "phoneVerify": "string",  
  "password": "string"  
}';  
const headers = {  
  'Content-Type': 'application/json',  
  'Accept': 'application/json'  
};  
  
fetch('http://localhost/api/master/v0/guest/register/phone',  
{  
  method: 'POST',  
  body: inputBody,  
  headers: headers  
})  
.then(function(res) {  
  return res.json();  
}).then(function(body) {  
  console.log(body);  
});
```

POST /guest/register/phone

註冊新帳號 *email* (驗證手機)

Register a new user by phone.

Body parameter

```
{  
  "email": "string",  
  "emailVerify": "string",  
  "phone": "string",  
  "phoneVerify": "string",  
  "password": "string"  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	Register informations
» email	body	string	true	none
» emailVerify	body	string	true	none
» phone	body	string	true	none
» phoneVerify	body	string	true	none
» password	body	string	true	none

Example responses

200 Response

```
{  
  "userId": "string",  
  "token": "string"  
}
```

Responses

Status	Meaning	Description	Schema
200	OK	Auth token.	Inline
403	Forbidden	註冊資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code 200

Login token

Name	Type	Required	Restrictions	Description
» userId	string	true	none	none
» token	string	true	none	none

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» email	string	false	none	none
» emailVerify	string	false	none	none
» phone	string	false	none	none
» phoneVerify	string	false	none	none
» password	string	false	none	none

This operation does not require authentication

guestLoginByPassword

Code samples

```
const inputBody = '{\n    "account": "string",\n    "password": "string"\n}';\nconst headers = {\n    'Content-Type': 'application/json',\n    'Accept': 'application/json'\n};\n\nfetch('http://localhost/api/master/v0/guest/login/password', {\n    method: 'POST',\n    body: inputBody,\n    headers: headers\n})\n.then(function(res) {\n    return res.json();\n}).then(function(body) {\n    console.log(body);\n});
```

POST /guest/login/password

使用 *email/password* 登入系統

Login by email and password.

Body parameter

```
{
  "account": "string",
  "password": "string"
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	Login fields
» account	body	string	true	email
» password	body	string	true	none

Example responses

200 Response

```
{
  "userId": "string",
  "token": "string"
}
```

Responses

Status	Meaning	Description	Schema
200	OK	Auth token.	Inline
401	Unauthorized	登入資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **200**

Login token

Name	Type	Required	Restrictions	Description
» userId	string	true	none	none
» token	string	true	none	none

Status Code **401**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» account	string	false	none	none
» password	string	false	none	none

This operation does not require authentication

guestVerifyEmail

Code samples

```
const inputBody = '  
  "email": "string",  
  "lang": "en-US"  
}';  
const headers = {  
  'Content-Type': 'application/json',  
  'Accept': 'application/json'  
};  
  
fetch('http://localhost/api/master/v0/guest/verify/email',  
{  
  method: 'POST',  
  body: inputBody,  
  headers: headers  
})  
.then(function(res) {  
  return res.json();  
}).then(function(body) {  
  console.log(body);  
});
```

POST /guest/verify/email

寄發電子郵件檢查碼信件

寄發電子郵件檢查碼信件

Body parameter

```
{  
  "email": "string",  
  "lang": "en-US"  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	none
» email	body	string	true	none
» lang	body	string	false	none

Example responses

200 Response

[1653502991840](#)

Responses

Status	Meaning	Description	Schema
200	OK	OK	integer
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» email	string	false	none	none
» time	integer	false	none	下次可執行時間

This operation does not require authentication

guestVerifyPhone

Code samples

```
const inputBody = '{\n  "phone": 886987654321,\n  "lang": "en-US"
```

```

};

const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json'
};

fetch('http://localhost/api/master/v0/guest/verify/phone',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

```

POST /guest/verify/phone

寄發手機檢查碼簡訊

寄發手機檢查碼簡訊

Body parameter

```
{
  "phone": 886987654321,
  "lang": "en-US"
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	none
» phone	body	string	true	none
» lang	body	string	false	none

Example responses

200 Response

1653502991840

Responses

Status	Meaning	Description	Schema
200	OK	OK	integer
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» phone	string	false	none	none
» time	integer	false	none	下次可執行時間

This operation does not require authentication

guestResetPassword

Code samples

```
const inputBody = `{
  "email": "string",
  "emailVerify": "string",
  "phoneVerify": "string",
  "password": "string"
}`;
const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json'
};

fetch('http://localhost/api/master/v0/guest/reset/password',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});
```

POST /guest/reset/password

重設密碼(未登入)

Reset member password

Body parameter

```
{  
  "email": "string",  
  "emailVerify": "string",  
  "phoneVerify": "string",  
  "password": "string"  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	Reset informations
» email	body	string	true	none
» emailVerify	body	string	true	none
» phoneVerify	body	string	true	none
» password	body	string	true	none

Example responses

403 Response

```
{  
  "email": "api.guest.register.error.email.format",  
  "emailVerify": "api.guest.register.error.emailVerify.notMatch",  
  "phoneVerify": "api.guest.register.error.phoneVerify.overtime",  
  "password": "api.guest.register.error.password.minLength8"  
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK --> renew token salt(logout all devices) --> 導向登入頁	None
403	Forbidden	註冊資訊錯誤	None
default	Default	Error response	None

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» email	string	false	none	none
» emailVerify	string	false	none	none
» phoneVerify	string	false	none	none
» password	string	false	none	none

This operation does not require authentication

guestVerifyAccountPhone

Code samples

```
const inputBody = '{\n  "email": "string",\n  "lang": "en-US"\n};\nconst headers = {\n  'Content-Type': 'application/json',\n  'Accept': 'application/json'\n};\n\nfetch('http://localhost/api/master/v0/guest/verify/account/phone', {\n  method: 'POST',\n  body: inputBody,\n  headers: headers\n})\n.then(function(res) {\n  return res.json();\n}).then(function(body) {\n  console.log(body);\n});
```

POST /guest/verify/account/phone

依帳號關聯手機號碼寄發檢查碼簡訊

依帳號關聯手機號碼寄發檢查碼簡訊

Body parameter

```
{  
  "email": "string",  
  "lang": "en-US"  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	none
» email	body	string	true	none
» lang	body	string	false	none

Example responses

200 Response

1653502991840

Responses

Status	Meaning	Description	Schema
200	OK	OK	integer
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» email	string	false	none	none
» time	integer	false	none	下次可執行時間

This operation does not require authentication

invoice

購物/上架/提領 訂單處理相關操作

invoiceList

Code samples

```
const inputBody = '  
  "userId": "string",  
  "offset": 0,  
  "limit": 0,  
  "filter": {  
    "invoiceType": "string",  
    "status": "string",  
    "createAfter": 0,  
    "createBefore": 0  
  },  
  "sort": [  
    {  
      "field": "string",  
      "mode": "asc"  

```

查詢交易歷史紀錄

查詢交易歷史紀錄

Body parameter

```
{  
  "userId": "string",  
  "offset": 0,  
  "limit": 0,  
  "filter": {  
    "invoiceType": "string",  
    "status": "string",  
    "createAfter": 0,  
    "createBefore": 0  
  },  
  "sort": [  
    {  
      "field": "string",  
      "mode": "asc"  
    }  
  ]  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	user check
» userId	body	string	true	none
» offset	body	integer	false	none
» limit	body	integer	false	none
» filter	body	object	false	none
»» invoiceType	body	string	false	shopping / selling / export / mint
»» status	body	string	false	none
»» createAfter	body	integer	false	none
»» createBefore	body	integer	false	none
» sort	body	[object]	false	none
»» field	body	string	false	createOn
»» mode	body	string	false	none

Enumerated Values

Parameter	Value
»» mode	asc
»» mode	des

Example responses

200 Response

```
{
  "userId": "string",
  "total": 0,
  "offset": 0,
  "list": [
    {
      "invoiceType": "string",
      "invoiceId": "string",
      "createOn": 0,
      "status": "string",
      "payment": [
        {
          "currency": "string",
          "amount": 0
        }
      ],
      "income": [
        {
          "currency": "string",
          "amount": 0
        }
      ]
    }
  ]
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	Inline
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **200**

invoice 列表

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none
» total	integer	false	none	查詢結果總數
» offset	integer	false	none	回傳起始位置
» list	[object]	false	none	none
»» invoiceType	string	false	none	none
»» invoiceId	string	false	none	none
»» createOn	integer	false	none	none
»» status	string	false	none	none
»» payment	[object]	false	none	none
»»» currency	string	false	none	none
»»» amount	number	false	none	none
»» income	[object]	false	none	none
»»» currency	string	false	none	none
»»» amount	number	false	none	none

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

invoiceCheck

Code samples

```
const inputBody = '{\n  "userId": "string",\n  "invoiceType": "string",\n  "invoiceId": "string",\n}
```

```

"lang": "string"
};

const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json',
  'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/invoice/check',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

```

POST /invoice/check

查詢訂單資訊

查詢訂單資訊

Body parameter

```
{
  "userId": "string",
  "invoiceType": "string",
  "invoiceId": "string",
  "lang": "string"
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	user check
» userId	body	string	true	none
» invoiceType	body	string	false	shopping / selling / export / mint
» invoiceId	body	string	false	none
» lang	body	string	false	none

Example responses

200 Response

```
{  
    "userId": "string",  
    "invoiceType": "string",  
    "invoiceId": "string",  
    "status": "success",  
    "createOn": 0,  
    "jobs": [  
        {  
            "jobType": "string",  
            "jobId": "string",  
            "data": {},  
            "status": "success",  
            "finishOn": 0,  
            "description": "string"  
        }  
    ]  
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	InvoiceStatus
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code 403

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

invoiceShoppingCreate

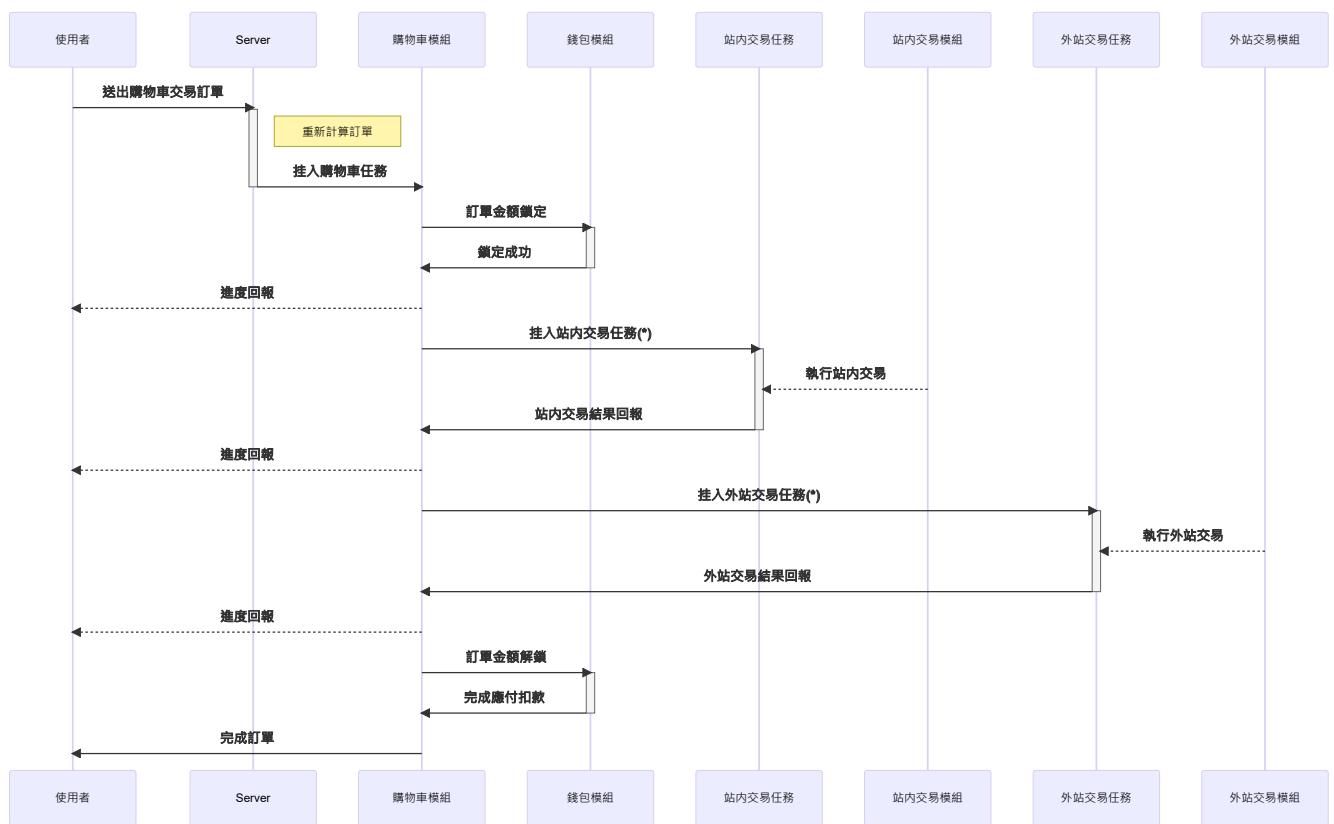
Code samples

```
const inputBody = '  
  "userId": "string",  
  "cartHash": "string",  
  "lang": "string"  
}';  
const headers = {  
  'Content-Type': 'application/json',  
  'Accept': 'application/json',  
  'apikey': 'API_KEY'  
};  
  
fetch('http://localhost/api/master/v0/invoice/shopping/create',  
{  
  method: 'POST',  
  body: inputBody,  
  headers: headers  
})  
.then(function(res) {  
  return res.json();  
}).then(function(body) {  
  console.log(body);  
});
```

POST /invoice/shopping/create

建立購物車交易訂單(流程圖)

建立購物車交易訂單



Body parameter

```
{
  "userId": "string",
  "cartHash": "string",
  "lang": "string"
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	shopping order
» userId	body	string	true	none
» cartHash	body	string	false	none
» lang	body	string	false	none

Example responses

200 Response

```
{
  "userId": "string",
  "invoiceType": "string",
  "invoiceId": "string",
  "status": "success",
  "createOn": 0,
  "jobs": [
    {
      "jobType": "string",
      "jobId": "string",
      "data": {},
      "status": "success",
      "finishOn": 0,
      "description": "string"
    }
  ]
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	InvoiceStatus
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none
» cartHash	string	false	none	none
» lang	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

invoiceSellingCreate

Code samples

```
const inputBody = `{
  "userId": "string",
  "cartHash": "string",
  "lang": "string"
}`;
const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json',
  'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/invoice/selling/create',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});
```

POST /invoice/selling/create

建立上架車交易訂單

建立上架車交易訂單

Body parameter

```
{  
  "userId": "string",  
  "cartHash": "string",  
  "lang": "string"  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	shopping order
» userId	body	string	true	none
» cartHash	body	string	false	none
» lang	body	string	false	none

Example responses

200 Response

```
{  
  "userId": "string",  
  "invoiceType": "string",  
  "invoiceId": "string",  
  "status": "success",  
  "createOn": 0,  
  "jobs": [  
    {  
      "jobType": "string",  
      "jobId": "string",  
      "data": {},  
      "status": "success",  
      "finishOn": 0,  
      "description": "string"  
    }  
  ]  
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	InvoiceStatus
403	Forbidden	資訊錯誤	Inline

Status	Meaning	Description	Schema
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none
» cartHash	string	false	none	none
» lang	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

invoiceSellingCancel

Code samples

```
const inputBody = '{
  "userId": "string",
  "invoiceId": "string",
  "chainId": 0,
  "contract": "string",
  "id": "string",
  "shop": "string",
  "payable": {
    "cost": [
      {
        "type": "string",
        "currency": "string",
        "amount": 0
      }
    ],
    "wallet": [
      {
        "currency": "string",
        "enable": true
      }
    ],
    "payment": [
      {

```

```

    "currency": "string",
    "amount": 0,
    "payBy": [
      {
        "currency": "string",
        "used": 0,
        "rate": 0,
        "amount": 0
      }
    ]
  }
}
};

const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json',
  'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/invoice/selling/cancel',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}

POST /invoice/selling/cancel

```

單一NFT下架

單一NFT下架

Body parameter

```
{
  "userId": "string",
  "invoiceId": "string",
  "chainId": 0,
  "contract": "string",
  "id": "string",
  "shop": "string",
  "payable": {
    "cost": [
      {
        "type": "string",
        "currency": "string",
        "amount": 0
      }
    ]
  }
}
```

```

],
"wallet": [
{
  "currency": "string",
  "enable": true
}
],
"payment": [
{
  "currency": "string",
  "amount": 0,
  "payBy": [
    {
      "currency": "string",
      "used": 0,
      "rate": 0,
      "amount": 0
    }
  ]
}
]
}
}

```

Parameters

Name	In	Type	Required	Description
body	body	object	true	cancel selling order
» userId	body	string	true	none
» invoiceId	body	string	false	none
» chainId	body	number	false	none
» contract	body	string	false	none
» id	body	string	false	none
» shop	body	string	false	none
» payable	body	object	false	none
»» cost	body	[object]	false	none
»»» type	body	string	false	gas / price / fee
»»» currency	body	string	false	none
»»» amount	body	number	false	none

Name	In	Type	Required	Description
»» wallet	body	[object]	false	none
»»» currency	body	string	false	none
»»» enable	body	boolean	false	none
»» payment	body	[object]	false	none
»»» currency	body	string	false	none
»»» amount	body	number	false	none
»»» payBy	body	[object]	false	none
»»»» currency	body	string	false	none
»»»» used	body	number	false	none
»»»» rate	body	number	false	none
»»»» amount	body	number	false	none

Example responses

200 Response

```
{
  "userId": "string",
  "invoiceType": "string",
  "invoiceId": "string",
  "status": "success",
  "createOn": 0,
  "jobs": [
    {
      "jobType": "string",
      "jobId": "string",
      "data": {},
      "status": "success",
      "finishOn": 0,
      "description": "string"
    }
  ]
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	InvoiceStatus

Status	Meaning	Description	Schema
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none
» cartHash	string	false	none	none
» lang	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

invoiceSellingCancelCost

Code samples

```
const inputBody = `{
  "userId": "string",
  "invoiceId": "string",
  "chainId": 0,
  "contract": "string",
  "id": "string",
  "shop": "string",
  "wallet": [
    {
      "currency": "string",
      "enable": true
    }
  ]
}`;
const headers = {
  'Content-Type': 'application/json',
  'Accept': 'application/json',
  'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/invoice/selling/cancel/cost',
```

```

{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

```

POST /invoice/selling/cancel/cost

查詢單一NFT下架費用

查詢單一NFT下架費用

Body parameter

```

{
  "userId": "string",
  "invoiceId": "string",
  "chainId": 0,
  "contract": "string",
  "id": "string",
  "shop": "string",
  "wallet": [
    {
      "currency": "string",
      "enable": true
    }
  ]
}

```

Parameters

Name	In	Type	Required	Description
body	body	object	true	none
» userId	body	string	true	none
» invoiceId	body	string	true	none
» chainId	body	number	true	none
» contract	body	string	true	none
» id	body	string	true	none
» shop	body	string	true	none

Name	In	Type	Required	Description
» wallet	body	[object]	false	none
»» currency	body	string	false	none
»» enable	body	boolean	false	none

Example responses

200 Response

```
{
  "cost": [
    {
      "type": "string",
      "currency": "string",
      "amount": 0
    }
  ],
  "wallet": [
    {
      "currency": "string",
      "enable": true,
      "balance": 0
    }
  ],
  "payment": [
    {
      "currency": "string",
      "amount": 0,
      "payBy": [
        {
          "currency": "string",
          "used": 0,
          "rate": 0,
          "amount": 0
        }
      ]
    }
  ]
}
```

Responses

Status	Meaning	Description	Schema
200	OK	OK	Inline

Status	Meaning	Description	Schema
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **200**

Name	Type	Required	Restrictions	Description
» cost	[object]	false	none	none
»» type	string	false	none	gas / price / fee
»» currency	string	false	none	none
»» amount	number	false	none	none
» wallet	[object]	false	none	none
»» currency	string	false	none	none
»» enable	boolean	false	none	none
»» balance	number	false	none	none
» payment	[object]	false	none	none
»» currency	string	false	none	none
»» amount	number	false	none	none
»» payBy	[object]	false	none	none
»»» currency	string	false	none	none
»»» used	number	false	none	none
»»» rate	number	false	none	none
»»» amount	number	false	none	none

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» userId	string	false	none	none

Name	Type	Required	Restrictions	Description
» cartHash	string	false	none	none
» lang	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

invoiceExportCreate

Code samples

```
const inputBody = `{
  "userId": "string",
  "chainId": 0,
  "list": [
    {
      "contract": "string",
      "id": 0
    }
  ],
  "receiver": "string",
  "sign": "string"
}`;
const headers = {
  'Content-Type':'application/json',
  'Accept':'application/json',
  'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/invoice/export/create',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
});

POST /invoice/export/create
```

建立NFT匯出訂單

建立NFT匯出訂單

Body parameter

```
{  
  "userId": "string",  
  "chainId": 0,  
  "list": [  
    {  
      "contract": "string",  
      "id": 0  
    }  
  ],  
  "receiver": "string",  
  "sign": "string"  
}
```

Parameters

Name	In	Type	Required	Description
body	body	object	true	user check
» userId	body	string	true	none
» chainId	body	integer	false	none
» list	body	[object]	false	none
»» contract	body	string	false	contract address
»» id	body	integer	false	nft id
» receiver	body	string	false	匯出錢包賬號
» sign	body	string	false	eip712 簽名證明為錢包持有人(可選)

Example responses

200 Response

```
{  
  "userId": "string",  
  "invoiceType": "string",  
  "invoiceId": "string",  
  "status": "success",  
  "createOn": 0,  
  "jobs": [  
    {  
      "jobType": "string",  
      "jobId": "string",  
      "data": {}  
    }  
  ]  
}
```

```

    "status": "success",
    "finishOn": 0,
    "description": "string"
}
]
}

```

Responses

Status	Meaning	Description	Schema
200	OK	OK	InvoiceStatus
403	Forbidden	資訊錯誤	Inline
default	Default	Error response	BaseError

Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

Name	Type	Required	Restrictions	Description
» chainId	string	false	none	none
» userId	string	false	none	none
» list	string	false	none	none
» importer	string	false	none	none
» sign	string	false	none	none

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

invoiceMintCreate

Code samples

```

const inputBody = '{
  "userId": "string",
  "mintId": "string",
  "quantity": 0,
  "payable": {
    "cost": [

```

```
{
  "type": "string",
  "currency": "string",
  "amount": 0
},
],
"wallet": [
  {
    "currency": "string",
    "enable": true
  }
],
"payment": [
  {
    "currency": "string",
    "amount": 0,
    "payBy": [
      {
        "currency": "string",
        "used": 0,
        "rate": 0,
        "amount": 0
      }
    ]
  }
]
}
};

const headers = {
  'Content-Type':'application/json',
  'Accept':'application/json',
  'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/invoice/mint/create',
{
  method: 'POST',
  body: inputBody,
  headers: headers
})
.then(function(res) {
  return res.json();
}).then(function(body) {
  console.log(body);
```
POST /invoice/mint/create
```

建立NFT鑄造訂單

建立NFT鑄造訂單

## Body parameter

```
{
 "userId": "string",
 "mintId": "string",
 "quantity": 0,
 "payable": [
 "cost": [
 {
 "type": "string",
 "currency": "string",
 "amount": 0
 }
],
 "wallet": [
 {
 "currency": "string",
 "enable": true
 }
],
 "payment": [
 {
 "currency": "string",
 "amount": 0,
 "payBy": [
 {
 "currency": "string",
 "used": 0,
 "rate": 0,
 "amount": 0
 }
]
 }
]
]
}
```

## Parameters

| Name       | In   | Type    | Required | Description            |
|------------|------|---------|----------|------------------------|
| body       | body | object  | true     | user check             |
| » userId   | body | string  | true     | none                   |
| » mintId   | body | string  | true     | 資料庫內 mint 按鈕 widget ID |
| » quantity | body | integer | false    | 鑄造數量                   |
| » payable  | body | object  | false    | none                   |

| Name          | In   | Type     | Required | Description       |
|---------------|------|----------|----------|-------------------|
| »» cost       | body | [object] | false    | none              |
| »»» type      | body | string   | false    | gas / price / fee |
| »»» currency  | body | string   | false    | none              |
| »»» amount    | body | number   | false    | none              |
| »» wallet     | body | [object] | false    | none              |
| »»» currency  | body | string   | false    | none              |
| »»» enable    | body | boolean  | false    | none              |
| »» payment    | body | [object] | false    | none              |
| »»» currency  | body | string   | false    | none              |
| »»» amount    | body | number   | false    | none              |
| »»» payBy     | body | [object] | false    | none              |
| »»»» currency | body | string   | false    | none              |
| »»»» used     | body | number   | false    | none              |
| »»»» rate     | body | number   | false    | none              |
| »»»» amount   | body | number   | false    | none              |

## Example responses

### 200 Response

```
{
 "userId": "string",
 "invoiceType": "string",
 "invoiceId": "string",
 "status": "success",
 "createOn": 0,
 "jobs": [
 {
 "jobType": "string",
 "jobId": "string",
 "data": {},
 "status": "success",
 "finishOn": 0,
 "description": "string"
 }
]
}
```

# Responses

| Status  | Meaning                   | Description    | Schema                        |
|---------|---------------------------|----------------|-------------------------------|
| 200     | <a href="#">OK</a>        | OK             | <a href="#">InvoiceStatus</a> |
| 403     | <a href="#">Forbidden</a> | 資訊錯誤           | Inline                        |
| default | Default                   | Error response | <a href="#">BaseError</a>     |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name           | Type   | Required | Restrictions | Description |
|----------------|--------|----------|--------------|-------------|
| » userId       | string | false    | none         | none        |
| » collectionId | string | false    | none         | none        |
| » mintId       | string | false    | none         | none        |
| » cost         | string | false    | none         | none        |
| » wallet       | string | false    | none         | none        |
| » payment      | string | false    | none         | none        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## member

已登入使用者操作界面

## memberInfo

Code samples

```
const inputBody = `{
 "userId": "string"
}`;
const headers = {
 'Content-Type': 'application/json',
}
```

```
'Accept': 'application/json',
'apikey': 'API_KEY'

};

fetch('http://localhost/api/master/v0/member/info',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});


```

POST /member/info

### 取得會員資訊

check and renew auth token

#### Body parameter

```
{
 "userId": "string"
}
```

## Parameters

| Name     | In   | Type   | Required | Description |
|----------|------|--------|----------|-------------|
| body     | body | object | true     | user id     |
| » userId | body | string | true     | none        |

#### Example responses

#### 200 Response

```
{
 "userId": "string",
 "email": "string",
 "level": 0
}
```

## Responses

| Status  | Meaning                      | Description    | Schema                    |
|---------|------------------------------|----------------|---------------------------|
| 200     | <a href="#">OK</a>           | 會員基本資料         | Inline                    |
| 401     | <a href="#">Unauthorized</a> | 登入資訊錯誤         | None                      |
| default | Default                      | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code **200**

| Name     | Type    | Required | Restrictions | Description |
|----------|---------|----------|--------------|-------------|
| » userId | string  | false    | none         | none        |
| » email  | string  | false    | none         | none        |
| » level  | integer | false    | none         | none        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## memberVerifyPhone

### Code samples

```
const inputBody = `{
 "userId": "string",
 "lang": "en-US"
}`;
const headers = {
 'Content-Type':'application/json',
 'Accept':'application/json',
 'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/member/verify/phone',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});
```

POST /member/verify/phone

送出手機驗證碼 (登入中)

send SMS to member phone for verify code

#### Body parameter

```
{
 "userId": "string",
 "lang": "en-US"
}
```

## Parameters

| Name     | In   | Type   | Required | Description |
|----------|------|--------|----------|-------------|
| body     | body | object | true     | user id     |
| » userId | body | string | true     | none        |
| » lang   | body | string | false    | none        |

#### Example responses

#### 200 Response

1653502991840

## Responses

| Status  | Meaning                      | Description    | Schema                    |
|---------|------------------------------|----------------|---------------------------|
| 200     | <a href="#">OK</a>           | OK             | integer                   |
| 401     | <a href="#">Unauthorized</a> | 登入資訊錯誤         | None                      |
| 403     | <a href="#">Forbidden</a>    | 資訊錯誤           | Inline                    |
| default | Default                      | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name | Type | Required | Restrictions | Description |
|------|------|----------|--------------|-------------|
|      |      |          |              |             |

| Name   | Type    | Required | Restrictions | Description      |
|--------|---------|----------|--------------|------------------|
| » time | integer | false    | none         | 此錯誤代表尚未到達可再次送出時間 |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## memberVerifyEmail

### Code samples

```
const inputBody = `{
 "userId": "string",
 "lang": "en-US"
}`;
const headers = {
 'Content-Type':'application/json',
 'Accept':'application/json',
 'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/member/verify/email',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});
```

POST /member/verify/email

寄發電子郵件檢查碼信件 (登入中)

寄發電子郵件檢查碼信件

### Body parameter

```
{
 "userId": "string",
 "lang": "en-US"
}
```

## Parameters

| Name     | In   | Type   | Required | Description |
|----------|------|--------|----------|-------------|
| body     | body | object | true     | none        |
| » userId | body | string | true     | none        |
| » lang   | body | string | false    | none        |

Example responses

200 Response

1653502991840

## Responses

| Status  | Meaning                   | Description    | Schema                    |
|---------|---------------------------|----------------|---------------------------|
| 200     | <a href="#">OK</a>        | OK             | integer                   |
| 403     | <a href="#">Forbidden</a> | 資訊錯誤           | Inline                    |
| default | Default                   | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name   | Type    | Required | Restrictions | Description |
|--------|---------|----------|--------------|-------------|
| » time | integer | false    | none         | 下次可執行時間     |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## memberResetPhone

Code samples

```
const inputBody = '{
 "userId": "string",
 "password": "string",
 "emailVerify": "string",
 "phone": "string",
 "phoneVerify": "string"
}'
```

```

};

const headers = {
 'Content-Type': 'application/json',
 'Accept': 'application/json',
 'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/member/reset/phone',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});

```

POST /member/reset/phone

**重設連結手機號碼**

重設連結手機號碼

#### Body parameter

```
{
 "userId": "string",
 "password": "string",
 "emailVerify": "string",
 "phone": "string",
 "phoneVerify": "string"
}
```

## Parameters

| Name          | In   | Type   | Required | Description                  |
|---------------|------|--------|----------|------------------------------|
| body          | body | object | true     | user id                      |
| » userId      | body | string | true     | none                         |
| » password    | body | string | true     | none                         |
| » emailVerify | body | string | true     | email 驗證碼                    |
| » phone       | body | string | true     | 新手機號碼                        |
| » phoneVerify | body | string | true     | 使用 guestVerifyPhone 送出之手機驗證碼 |

Example responses

403 Response

```
{
 "emailVerify": "api.member.reset.phone.error.emailVerify.notMatch",
 "phone": "api.member.reset.phone.error.phone.exist",
 "phoneVerify": "api.member.reset.phone.error.phoneVerify.notMatch",
 "password": "api.member.reset.phone.error.password.notMatch"
}
```

## Responses

| Status  | Meaning                      | Description                                 | Schema                    |
|---------|------------------------------|---------------------------------------------|---------------------------|
| 200     | <a href="#">OK</a>           | OK --> renew token salt(logout all devices) | None                      |
| 401     | <a href="#">Unauthorized</a> | 登入資訊錯誤                                      | None                      |
| 403     | <a href="#">Forbidden</a>    | 資訊錯誤                                        | Inline                    |
| default | Default                      | Error response                              | <a href="#">BaseError</a> |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name          | Type   | Required | Restrictions | Description |
|---------------|--------|----------|--------------|-------------|
| » emailVerify | string | false    | none         | none        |
| » phone       | string | false    | none         | none        |
| » phoneVerify | string | false    | none         | none        |
| » password    | string | false    | none         | none        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## memberResetPassword

Code samples

```

const inputBody = `{
 "userId": "string",
 "emailVerify": "string",
 "phoneVerify": "string",
 "password": "string"
}`;
const headers = {
 'Content-Type': 'application/json',
 'Accept': 'application/json',
 'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/member/reset/password',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});

```

POST /member/reset/password

## 重設登入密碼

Reset member password

### Body parameter

```
{
 "userId": "string",
 "emailVerify": "string",
 "phoneVerify": "string",
 "password": "string"
}
```

## Parameters

| Name          | In   | Type   | Required | Description        |
|---------------|------|--------|----------|--------------------|
| body          | body | object | true     | Reset informations |
| » userId      | body | string | true     | none               |
| » emailVerify | body | string | true     | none               |
| » phoneVerify | body | string | true     | none               |

| Name       | In   | Type   | Required | Description |
|------------|------|--------|----------|-------------|
| » password | body | string | true     | none        |

Example responses

403 Response

```
{
 "email": "api.member.reset.password.error.email.format",
 "emailVerify": "api.member.reset.password.error.emailVerify.notMatch",
 "phoneVerify": "api.member.reset.password.error.phoneVerify.overtime",
 "password": "api.member.reset.password.error.password.minLength8"
}
```

## Responses

| Status  | Meaning                   | Description                                 | Schema                    |
|---------|---------------------------|---------------------------------------------|---------------------------|
| 200     | <a href="#">OK</a>        | OK --> renew token salt(logout all devices) | None                      |
| 403     | <a href="#">Forbidden</a> | 註冊資訊錯誤                                      | Inline                    |
| default | Default                   | Error response                              | <a href="#">BaseError</a> |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name          | Type   | Required | Restrictions | Description |
|---------------|--------|----------|--------------|-------------|
| » email       | string | false    | none         | none        |
| » emailVerify | string | false    | none         | none        |
| » phoneVerify | string | false    | none         | none        |
| » password    | string | false    | none         | none        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

**mint**

NFT 代客鑄造設定界面

# mintList

## Code samples

```
const inputBody = {
 "userId": "string",
 "offset": 0,
 "limit": 0,
 "filter": {
 "status": "string",
 "createAfter": 0,
 "createBefore": 0,
 "chainId": 0,
 "category": "string",
 "mints": [
 "string"
]
 },
 "sort": [
 {
 "field": "string",
 "mode": "asc"
 }
]
};

const headers = {
 'Content-Type': 'application/json',
 'Accept': 'application/json'
};

fetch('http://localhost/api/master/v0/mint/list',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});

POST /mint/list
```

列出代鑄專案清單

mint list

## Body parameter

```
{
 "userId": "string",
 "offset": 0,
 "limit": 0,
 "filter": {
 "status": "string",
 "createAfter": 0,
 "createBefore": 0,
 "chainId": 0,
 "category": "string",
 "mints": [
 "string"
]
 },
 "sort": [
 {
 "field": "string",
 "mode": "asc"
 }
]
}
```

## Parameters

| Name            | In   | Type     | Required | Description |
|-----------------|------|----------|----------|-------------|
| body            | body | object   | true     | user id     |
| » userId        | body | string   | true     | none        |
| » offset        | body | integer  | false    | none        |
| » limit         | body | integer  | false    | none        |
| » filter        | body | object   | false    | none        |
| »» status       | body | string   | false    | none        |
| »» createAfter  | body | integer  | false    | none        |
| »» createBefore | body | integer  | false    | none        |
| »» chainId      | body | integer  | false    | none        |
| »» category     | body | string   | false    | none        |
| »» mints        | body | [string] | false    | none        |
| » sort          | body | [object] | false    | none        |
| »» field        | body | string   | false    | none        |

| Name    | In   | Type   | Required | Description |
|---------|------|--------|----------|-------------|
| »» mode | body | string | false    | none        |

## Enumerated Values

| Parameter | Value |
|-----------|-------|
| »» mode   | asc   |
| »» mode   | des   |

Example responses

200 Response

```
{
 "total": 0,
 "offset": 0,
 "list": [
 {
 "id": "string",
 "hash": "string",
 "createOn": 0,
 "banner": "string",
 "image": "string",
 "status": "string",
 "title": "string",
 "mintOn": 0,
 "currency": "string",
 "price": 0,
 "totalMints": 0,
 "totalSupply": 0,
 "totalMinted": 0
 }
]
}
```

## Responses

| Status  | Meaning                   | Description    | Schema                    |
|---------|---------------------------|----------------|---------------------------|
| 200     | <a href="#">OK</a>        | OK             | Inline                    |
| 403     | <a href="#">Forbidden</a> | 資訊錯誤           | Inline                    |
| default | Default                   | Error response | <a href="#">BaseError</a> |

# Response Schema

Status Code 200

NFT 系列回傳物件

| Name           | Type                                 | Required | Restrictions | Description  |
|----------------|--------------------------------------|----------|--------------|--------------|
| » total        | integer                              | false    | none         | 查詢結果總數       |
| » offset       | integer                              | false    | none         | 回傳起始位置       |
| » list         | [ <a href="#">MintSummaryModel</a> ] | false    | none         | none         |
| »» id          | string                               | false    | none         | none         |
| »» hash        | string                               | false    | none         | none         |
| »» createOn    | integer                              | false    | none         | none         |
| »» banner      | string                               | false    | none         | 大圖連結         |
| »» image       | string                               | false    | none         | 小圖/NFT示意圖連結  |
| »» status      | string                               | false    | none         | none         |
| »» title       | string                               | false    | none         | none         |
| »» mintOn      | integer                              | false    | none         | none         |
| »» currency    | string                               | false    | none         | none         |
| »» price       | number                               | false    | none         | none         |
| »» totalMints  | integer                              | false    | none         | 站內 mint 總次數  |
| »» totalSupply | integer                              | false    | none         | 合約總供應量       |
| »» totalMinted | integer                              | false    | none         | 合約已 mint 總數量 |

Status Code 403

回傳對應欄位的錯誤訊息/代號

| Name     | Type   | Required | Restrictions | Description |
|----------|--------|----------|--------------|-------------|
| » filter | string | false    | none         | ...         |
| » sort   | string | false    | none         | ...         |

This operation does not require authentication

# mintRead

## Code samples

```
const inputBody = '{\n "userId": "string",\n "mintId": "string"\n}';\nconst headers = {\n 'Content-Type':'application/json',\n 'Accept':'application/json'\n};\n\nfetch('http://localhost/api/master/v0/mint/read',\n{\n method: 'POST',\n body: inputBody,\n headers: headers\n})\n.then(function(res) {\n return res.json();\n}).then(function(body) {\n console.log(body);\n});
```

POST /mint/read

取得專案資訊

mint list

## Body parameter

```
{\n "userId": "string",\n "mintId": "string"\n}
```

## Parameters

| Name     | In   | Type   | Required | Description |
|----------|------|--------|----------|-------------|
| body     | body | object | true     | user id     |
| » userId | body | string | true     | none        |
| » mintId | body | string | false    | none        |

## Example responses

## 200 Response

```
{
 "id": "string",
 "hash": "string",
 "createOn": 0,
 "isLock": true,
 "isPublish": true,
 "editable": true,
 "mintOn": 0,
 "banner": "string",
 "image": "string",
 "description": "string",
 "homepage": "string",
 "communities": [
 {
 "platform": "string",
 "url": "string"
 }
],
 "status": "string",
 "title": "string",
 "currency": "string",
 "price": 0,
 "collection": "string",
 "chainId": 0,
 "contract": "string",
 "function": "string",
 "argument": {
 "fix": 0,
 "min": 0,
 "max": 0
 },
 "expected": 0
}
```

## Responses

| Status  | Meaning                   | Description    | Schema                          |
|---------|---------------------------|----------------|---------------------------------|
| 200     | <a href="#">OK</a>        | OK             | <a href="#">MintDetailModel</a> |
| 403     | <a href="#">Forbidden</a> | 資訊錯誤           | Inline                          |
| default | Default                   | Error response | <a href="#">BaseError</a>       |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name     | Type   | Required | Restrictions | Description |
|----------|--------|----------|--------------|-------------|
| » userId | string | false    | none         | ...         |
| » mintId | string | false    | none         | ...         |

This operation does not require authentication

## mintCreate

### Code samples

```
const inputBody = '
 "userId": "string",
 "detail": {
 "id": "string",
 "hash": "string",
 "createOn": 0,
 "isLock": true,
 "isPublish": true,
 "editable": true,
 "mintOn": 0,
 "banner": "string",
 "image": "string",
 "description": "string",
 "homepage": "string",
 "communities": [
 {
 "platform": "string",
 "url": "string"
 }
],
 "status": "string",
 "title": "string",
 "currency": "string",
 "price": 0,
 "collection": "string",
 "chainId": 0,
 "contract": "string",
 "function": "string",
 "argument": {
 "fix": 0,
 "min": 0,
 "max": 0
 },
 "expected": 0
 }
```

```
};

const headers = {
 'Content-Type': 'application/json',
 'Accept': 'application/json',
 'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/mint/create',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}) then(function(body) {

 POST /mint/create
```

## 建立專案

mint create

### Body parameter

```
{
 "userId": "string",
 "detail": {
 "id": "string",
 "hash": "string",
 "createOn": 0,
 "isLock": true,
 "isPublish": true,
 "editable": true,
 "mintOn": 0,
 "banner": "string",
 "image": "string",
 "description": "string",
 "homepage": "string",
 "communities": [
 {
 "platform": "string",
 "url": "string"
 }
],
 "status": "string",
 "title": "string",
 "currency": "string",
 "price": 0,
 "collection": "string",
 "chainId": 0,
```

```

"contract": "string",
"function": "string",
"argument": {
 "fix": 0,
 "min": 0,
 "max": 0
},
"expected": 0
}
}

```

## Parameters

| Name         | In   | Type                            | Required | Desc            |
|--------------|------|---------------------------------|----------|-----------------|
| body         | body | object                          | true     | user id         |
| » userId     | body | string                          | true     | none            |
| » detail     | body | <a href="#">MintDetailModel</a> | false    | none            |
| »» id        | body | string                          | false    | none            |
| »» hash      | body | string                          | false    | none            |
| »» createOn  | body | integer                         | false    | none            |
| »» isLock    | body | boolean                         | false    | 是否已鎖定           |
| »» isPublish | body | boolean                         | false    | 是否已上架(可查詢)      |
| »» editable  | body | boolean                         | false    | 是否可編輯, 伺服器視狀態給定 |
| »» mintOn    | body | integer                         | false    | 合約可執行鑄造時間       |
| »» banner    | body | string                          | false    | 大圖連結            |
| »» image     | body | string                          | false    | 小圖/NFT示意圖連結     |

|                |      |          |       |                              |
|----------------|------|----------|-------|------------------------------|
| »» description | body | string   | false | 項目及鑄造說明文字                    |
| »» homepage    | body | string   | false | 官方網站連結                       |
| »» communities | body | [object] | false | none                         |
| »»» platform   | body | string   | false | discord / facebook / twitter |
| »»» url        | body | string   | false | none                         |
| »» status      | body | string   | false | none                         |

| »» title      | Name | In   | Type    | Required | none             | Desc |
|---------------|------|------|---------|----------|------------------|------|
| »» currency   |      | body | string  | false    | none             |      |
| »» price      |      | body | number  | false    | 單價               |      |
| »» collection |      | body | string  | false    | NFT 集合名稱         |      |
| »» chainId    |      | body | integer | false    | none             |      |
| »» contract   |      | body | string  | false    | NFT 合約地址         |      |
| »» function   |      | body | string  | false    | 呼叫合約函數名稱         |      |
| »» argument   |      | body | object  | false    | 呼叫合約參數(鑄造數量)     |      |
| »»» fix       |      | body | integer | false    | none             |      |
| »»» min       |      | body | integer | false    | none             |      |
| »»» max       |      | body | integer | false    | none             |      |
| »» expected   |      | body | integer | false    | 預期收到NFT數量(不參與計算) |      |

## Example responses

### 200 Response

```
{
 "id": "string",
 "hash": "string",
 "createOn": 0,
 "isLock": true,
 "isPublish": true,
 "editable": true,
 "mintOn": 0,
 "banner": "string",
 "image": "string",
 "description": "string",
 "homepage": "string",
 "communities": [
 {
 "platform": "string",
 "url": "string"
 }
],
 "status": "string",
 "title": "string",
 "currency": "string",
 "price": 0,
 "collection": "string",
 "chainId": 0,
 "mintOn": 0
}
```

```

"contract": "string",
"function": "string",
"argument": {
 "fix": 0,
 "min": 0,
 "max": 0
},
"expected": 0
}

```

## Responses

| Status  | Meaning                      | Description    | Schema                          |
|---------|------------------------------|----------------|---------------------------------|
| 200     | <a href="#">OK</a>           | OK             | <a href="#">MintDetailModel</a> |
| 401     | <a href="#">Unauthorized</a> | 登入資訊錯誤         | None                            |
| 403     | <a href="#">Forbidden</a>    | 資訊錯誤           | Inline                          |
| default | Default                      | Error response | <a href="#">BaseError</a>       |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name     | Type   | Required | Restrictions | Description |
|----------|--------|----------|--------------|-------------|
| » hash   | string | false    | none         | none        |
| » detail | string | false    | none         | none        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## mintUpdate

### Code samples

```

const inputBody = '{
 "userId": "string",
 "hash": "string",
 "detail": {
 "id": "string",
 "hash": "string",
 "createOn": 0,
 }
}

```

```
"isLock": true,
"isPublish": true,
"editable": true,
"mintOn": 0,
"banner": "string",
"image": "string",
"description": "string",
"homepage": "string",
"communities": [
 {
 "platform": "string",
 "url": "string"
 }
],
"status": "string",
"title": "string",
"currency": "string",
"price": 0,
"collection": "string",
"chainId": 0,
"contract": "string",
"function": "string",
"argument": {
 "fix": 0,
 "min": 0,
 "max": 0
},
"expected": 0
}
}';

const headers = {
 'Content-Type': 'application/json',
 'Accept': 'application/json',
 'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/mint/update',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();

POST /mint/update
```

更新專案

mint update

## Body parameter

```
{
 "userId": "string",
 "hash": "string",
 "detail": {
 "id": "string",
 "hash": "string",
 "createOn": 0,
 "isLock": true,
 "isPublish": true,
 "editable": true,
 "mintOn": 0,
 "banner": "string",
 "image": "string",
 "description": "string",
 "homepage": "string",
 "communities": [
 {
 "platform": "string",
 "url": "string"
 }
],
 "status": "string",
 "title": "string",
 "currency": "string",
 "price": 0,
 "collection": "string",
 "chainId": 0,
 "contract": "string",
 "function": "string",
 "argument": {
 "fix": 0,
 "min": 0,
 "max": 0
 },
 "expected": 0
 }
}
```

## Parameters

| Name     | In   | Type            | Required | Desc    |
|----------|------|-----------------|----------|---------|
| body     | body | object          | true     | user id |
| » userId | body | string          | true     | none    |
| » hash   | body | string          | false    | none    |
| » detail | body | MintDetailModel | false    | none    |

| Name        | In   | Type    | Required | none  | Desc |
|-------------|------|---------|----------|-------|------|
| »» id       | body | string  | false    | none  |      |
| »» hash     | body | string  | false    | none  |      |
| »» createOn | body | integer | false    | none  |      |
| »» isLock   | body | boolean | false    | 是否已鎖定 |      |

|                |      |          |       |                              |
|----------------|------|----------|-------|------------------------------|
| »» isPublish   | body | boolean  | false | 是否已上架(可查詢)                   |
| »» editable    | body | boolean  | false | 是否可編輯, 伺服器視狀態給定              |
| »» mintOn      | body | integer  | false | 合約可執行鑄造時間                    |
| »» banner      | body | string   | false | 大圖連結                         |
| »» image       | body | string   | false | 小圖/NFT示意圖連結                  |
| »» description | body | string   | false | 項目及鑄造說明文字                    |
| »» homepage    | body | string   | false | 官方網站連結                       |
| »» communities | body | [object] | false | none                         |
| »»» platform   | body | string   | false | discord / facebook / twitter |
| »»» url        | body | string   | false | none                         |
| »» status      | body | string   | false | none                         |
| »» title       | body | string   | false | none                         |
| »» currency    | body | string   | false | none                         |
| »» price       | body | number   | false | 單價                           |
| »» collection  | body | string   | false | NFT 集合名稱                     |
| »» chainId     | body | integer  | false | none                         |
| »» contract    | body | string   | false | NFT 合約地址                     |
| »» function    | body | string   | false | 呼叫合約函數名稱                     |
| »» argument    | body | object   | false | 呼叫合約參數(鑄造數量)                 |
| »»» fix        | body | integer  | false | none                         |
| »»» min        | body | integer  | false | none                         |
| »»» max        | body | integer  | false | none                         |

| »» expected<br>Name | body<br>In | integer<br>Type | raise<br>Required | 預期收到JSON數量(不參與計算)<br>Desc |
|---------------------|------------|-----------------|-------------------|---------------------------|
|---------------------|------------|-----------------|-------------------|---------------------------|

## Example responses

### 200 Response

```
{
 "id": "string",
 "hash": "string",
 "createOn": 0,
 "isLock": true,
 "isPublish": true,
 "editable": true,
 "mintOn": 0,
 "banner": "string",
 "image": "string",
 "description": "string",
 "homepage": "string",
 "communities": [
 {
 "platform": "string",
 "url": "string"
 }
],
 "status": "string",
 "title": "string",
 "currency": "string",
 "price": 0,
 "collection": "string",
 "chainId": 0,
 "contract": "string",
 "function": "string",
 "argument": {
 "fix": 0,
 "min": 0,
 "max": 0
 },
 "expected": 0
}
```

## Responses

| Status | Meaning                      | Description | Schema                          |
|--------|------------------------------|-------------|---------------------------------|
| 200    | <a href="#">OK</a>           | OK          | <a href="#">MintDetailModel</a> |
| 401    | <a href="#">Unauthorized</a> | 登入資訊錯誤      | None                            |
| 403    | <a href="#">Forbidden</a>    | 資訊錯誤        | Inline                          |

| Status  | Meaning | Description    | Schema                    |
|---------|---------|----------------|---------------------------|
| default | Default | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name     | Type   | Required | Restrictions | Description |
|----------|--------|----------|--------------|-------------|
| » hash   | string | false    | none         | none        |
| » detail | string | false    | none         | none        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## mintDelete

### Code samples

```
const inputBody = `{
 "userId": "string",
 "mintId": "string",
 "mintHash": "string"
}`;
const headers = {
 'Content-Type':'application/json',
 'Accept':'application/json',
 'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/mint/delete',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});

POST /mint/delete
```

## 刪除專案

mint delete

### Body parameter

```
{
 "userId": "string",
 "mintId": "string",
 "mintHash": "string"
}
```

## Parameters

| Name       | In   | Type   | Required | Description |
|------------|------|--------|----------|-------------|
| body       | body | object | true     | user id     |
| » userId   | body | string | true     | none        |
| » mintId   | body | string | false    | none        |
| » mintHash | body | string | false    | none        |

### Example responses

#### 403 Response

```
{
 "mintId": "string",
 "mintHash": "string"
}
```

## Responses

| Status  | Meaning                      | Description    | Schema                    |
|---------|------------------------------|----------------|---------------------------|
| 200     | <a href="#">OK</a>           | OK             | None                      |
| 401     | <a href="#">Unauthorized</a> | 登入資訊錯誤         | None                      |
| 403     | <a href="#">Forbidden</a>    | 資訊錯誤           | Inline                    |
| default | Default                      | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name       | Type   | Required | Restrictions | Description |
|------------|--------|----------|--------------|-------------|
| » mintId   | string | false    | none         | none        |
| » mintHash | string | false    | none         | none        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## mintAction

### Code samples

```
const inputBody = `{
 "userId": "string",
 "mintId": "string",
 "hash": "string",
 "action": "string"
}`;
const headers = {
 'Content-Type':'application/json',
 'Accept':'application/json',
 'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/mint/action',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});

POST /mint/action
```

設定專案狀態 啓用/停用/上架/下架/送審/測試

mint action

### Body parameter

```
{
 "userId": "string",
```

```

 "mintId": "string",
 "hash": "string",
 "action": "string"
}

```

## Parameters

| Name     | In   | Type   | Required | Description                 |
|----------|------|--------|----------|-----------------------------|
| body     | body | object | true     | user id                     |
| » userId | body | string | true     | none                        |
| » mintId | body | string | false    | none                        |
| » hash   | body | string | false    | none                        |
| » action | body | string | false    | 啓用 / 停用 / 上架 / 下架 / 送審 / 測試 |

### Example responses

#### 200 Response

```
{
 "id": "string",
 "hash": "string",
 "createOn": 0,
 "isLock": true,
 "isPublish": true,
 "editable": true,
 "mintOn": 0,
 "banner": "string",
 "image": "string",
 "description": "string",
 "homepage": "string",
 "communities": [
 {
 "platform": "string",
 "url": "string"
 }
],
 "status": "string",
 "title": "string",
 "currency": "string",
 "price": 0,
 "collection": "string",
 "chainId": 0,
 "contract": "string",
 "function": "string",
 "argument": {
 "key": "string",
 "value": "string"
 }
}
```

```

 "fix": 0,
 "min": 0,
 "max": 0
},
"expected": 0
}

```

## Responses

| Status  | Meaning                      | Description    | Schema                          |
|---------|------------------------------|----------------|---------------------------------|
| 200     | <a href="#">OK</a>           | OK             | <a href="#">MintDetailModel</a> |
| 401     | <a href="#">Unauthorized</a> | 登入資訊錯誤         | None                            |
| 403     | <a href="#">Forbidden</a>    | 資訊錯誤           | Inline                          |
| default | Default                      | Error response | <a href="#">BaseError</a>       |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息／代號

| Name     | Type   | Required | Restrictions | Description |
|----------|--------|----------|--------------|-------------|
| » mintId | string | false    | none         | none        |
| » hash   | string | false    | none         | none        |
| » action | string | false    | none         | none        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## wallet

使用者錢包操作

### walletBalance

Code samples

```

const inputBody = `{
 "userId": "string"
}`;

```

```

const headers = {
 'Content-Type': 'application/json',
 'Accept': 'application/json',
 'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/wallet/balance',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});

```

POST /wallet/balance

取得錢包可用餘額

wallet balance

#### Body parameter

```
{
 "userId": "string"
}
```

## Parameters

| Name     | In   | Type   | Required | Description |
|----------|------|--------|----------|-------------|
| body     | body | object | true     | user id     |
| » userId | body | string | true     | none        |

#### Example responses

##### 200 Response

```
[
 {
 "currency": "ETH",
 "symbol": "\u262e",
 "decimal": 4,
 "balance": 0,
 "lock": 0
 }
]
```

```
 }
]
```

## Responses

| Status  | Meaning                      | Description    | Schema                    |
|---------|------------------------------|----------------|---------------------------|
| 200     | <a href="#">OK</a>           | 錢包餘額           | Inline                    |
| 401     | <a href="#">Unauthorized</a> | 登入資訊錯誤         | None                      |
| default | Default                      | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code **200**

| Name       | Type    | Required | Restrictions | Description |
|------------|---------|----------|--------------|-------------|
| » currency | string  | false    | none         | 虛擬貨幣名稱      |
| » symbol   | string  | false    | none         | 貨幣標識符       |
| » decimal  | integer | false    | none         | 小數位數        |
| » balance  | number  | false    | none         | 可動用餘額       |
| » lock     | number  | false    | none         | 鎖倉金額        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## walletHistory

### Code samples

```
const inputBody = '{
 "userId": "string",
 "offset": 0,
 "limit": 20
}';
const headers = {
 'Content-Type':'application/json',
 'Accept':'application/json',
 'apikey':'API_KEY'

```

```
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});
```

POST /wallet/history

## 分頁拉取用戶存提幣記錄

## 分頁拉取用戶存提幣記錄

## Body parameter

```
{
 "userId": "string",
 "offset": 0,
 "limit": 20
}
```

## Parameters

| Name     | In   | Type    | Required | Description      |
|----------|------|---------|----------|------------------|
| body     | body | object  | true     | none             |
| » userId | body | string  | true     | none             |
| » offset | body | integer | false    | 起始資料位置索引(0 base) |
| » limit  | body | integer | false    | 每頁個數             |

## Example responses

## 200 Response

```

 "decimal": 4,
 "from": "string",
 "into": "string",
 "chainId": 1,
 "txns": "string",
 "direction": "deposit",
 "amount": 0,
 "status": "success"
}
]
}

```

## Responses

| Status  | Meaning                      | Description    | Schema                    |
|---------|------------------------------|----------------|---------------------------|
| 200     | <a href="#">OK</a>           | 錢包存提紀錄         | Inline                    |
| 401     | <a href="#">Unauthorized</a> | 登入資訊錯誤         | None                      |
| default | Default                      | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code 200

| Name        | Type     | Required | Restrictions | Description                                    |
|-------------|----------|----------|--------------|------------------------------------------------|
| » total     | integer  | false    | none         | none                                           |
| » offset    | integer  | false    | none         | none                                           |
| » list      | [object] | false    | none         | none                                           |
| »» id       | string   | false    | none         | none                                           |
| »» time     | number   | false    | none         | 建立時間                                           |
| »» currency | string   | false    | none         | 虛擬貨幣名稱                                         |
| »» symbol   | string   | false    | none         | 貨幣標識符                                          |
| »» decimal  | integer  | false    | none         | 顯示小數位數                                         |
| »» from     | string   | false    | none         | 出金賬號                                           |
| »» into     | string   | false    | none         | 入金賬號                                           |
| »» chainId  | string   | false    | none         | <a href="#">交易公鏈 ID</a> 1 for Ethereum Mainnet |
| »» txns     | string   | false    | none         | 鏈上交易序號                                         |

| <b>Name</b>  | <b>Type</b> | <b>Required</b> | <b>Restrictions</b> | <b>none</b> | <b>Description</b> |
|--------------|-------------|-----------------|---------------------|-------------|--------------------|
| »» direction | string      | false           | none                | none        | 交易數量               |
| »» amount    | number      | false           | none                | none        |                    |
| »» status    | string      | false           | none                | none        |                    |

## Enumerated Values

| <b>Property</b> | <b>Value</b> |
|-----------------|--------------|
| direction       | diposit      |
| direction       | withdraw     |
| status          | success      |
| status          | cancel       |
| status          | processing   |
| status          | waiting      |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## walletDepositInfo

### Code samples

```
const inputBody = `{
 "userId": "string",
 "currency": "ETH"
}`;
const headers = {
 'Content-Type':'application/json',
 'Accept':'application/json',
 'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/wallet/diposit/info',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {
```

```
 console.log(body);
});
```

POST /wallet/diposit/info

取得虛擬貨幣入金帳號

取得虛擬貨幣入金帳號

#### Body parameter

```
{
 "userId": "string",
 "currency": "ETH"
}
```

## Parameters

| Name       | In   | Type   | Required | Description |
|------------|------|--------|----------|-------------|
| body       | body | object | true     | user id     |
| » userId   | body | string | true     | none        |
| » currency | body | string | true     | 虛擬貨幣名稱      |

#### Example responses

#### 200 Response

```
[
 {
 "currency": "ETH",
 "token": "0xdAC17F958D2ee523a2206206994597C13D831xxx",
 "chainId": 1,
 "address": "string",
 "memo": "string",
 "effect": 1653631833378
 }
]
```

## Responses

| Status | Meaning                      | Description | Schema |
|--------|------------------------------|-------------|--------|
| 200    | <a href="#">OK</a>           | 入金帳號        | Inline |
| 401    | <a href="#">Unauthorized</a> | 登入資訊錯誤      | None   |

| Status  | Meaning | Description    | Schema                    |
|---------|---------|----------------|---------------------------|
| default | Default | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code **200**

| Name       | Type   | Required | Restrictions | Description                                    |
|------------|--------|----------|--------------|------------------------------------------------|
| » currency | string | false    | none         | 虛擬貨幣名稱                                         |
| » token    | string | false    | none         | 虛擬貨幣合約地址(ex4:USDT on ether)                    |
| » chainId  | string | true     | none         | <a href="#">匯款公鍊 ID</a> 1 for Ethereum Mainnet |
| » address  | string | true     | none         | 匯入帳號/空為目前無可用帳號                                 |
| » memo     | string | false    | none         | 匯款注記(BNB Mainnet)                              |
| » effect   | number | false    | none         | 有效期限(timestamp)/0為無限期有效(                       |

To perform this operation, you must be authenticated by means of one of the following methods: `BaseApiKey`

## walletWithdrawInfo

### Code samples

```

const inputBody = `{
 "userId": "string",
 "currency": "ETH"
}`;
const headers = {
 'Content-Type':'application/json',
 'Accept':'application/json',
 'apikey': 'API_KEY'
};

fetch('http://localhost/api/master/v0/wallet/withdraw/info',
{
 method: 'POST',
 body: inputBody,
 headers: headers
})
.then(function(res) {
 return res.json();
}).then(function(body) {

```

```
 console.log(body);
});
```

POST /wallet/withdraw/info

取得虛擬貨幣出金方式

取得虛擬貨幣出金方式

#### Body parameter

```
{
 "userId": "string",
 "currency": "ETH"
}
```

## Parameters

| Name       | In   | Type   | Required | Description |
|------------|------|--------|----------|-------------|
| body       | body | object | true     | user id     |
| » userId   | body | string | true     | none        |
| » currency | body | string | true     | 虛擬貨幣名稱      |

#### Example responses

#### 200 Response

```
[
 {
 "currency": "ETH",
 "symbol": "Ξ",
 "token": "0xAc17F958D2ee523a2206206994597C13D831xxx",
 "chainId": 1,
 "min": 0,
 "max": 0,
 "fee": 0
 }
]
```

## Responses

| Status | Meaning                      | Description | Schema |
|--------|------------------------------|-------------|--------|
| 200    | <a href="#">OK</a>           | 出金管道        | Inline |
| 401    | <a href="#">Unauthorized</a> | 登入資訊錯誤      | None   |

| Status  | Meaning | Description    | Schema                    |
|---------|---------|----------------|---------------------------|
| default | Default | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code **200**

| Name       | Type   | Required | Restrictions | Description                                    |
|------------|--------|----------|--------------|------------------------------------------------|
| » currency | string | true     | none         | 虛擬貨幣名稱                                         |
| » symbol   | string | false    | none         | 貨幣標識符                                          |
| » token    | string | false    | none         | 虛擬貨幣合約地址(ex4:USDT on ether)                    |
| » chainId  | string | true     | none         | <a href="#">匯款公鏈 ID</a> 1 for Ethereum Mainnet |
| » min      | number | false    | none         | 最小轉出數量                                         |
| » max      | number | false    | none         | 目前可用最大轉出數量(包含會員等級限額)                           |
| » fee      | number | false    | none         | 轉出手續費(同幣別內扣)                                   |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## walletWithdrawSubmit

### Code samples

```
const inputBody = `{
 "userId": "string",
 "currency": "ETH",
 "chainId": 1,
 "amount": 0,
 "address": "string",
 "phoneVerify": "string"
}`;
const headers = {
 'Content-Type':'application/json',
 'Accept':'application/json',
 'apikey':'API_KEY'
};

fetch('http://localhost/api/master/v0/wallet/withdraw/submit',
{
 method: 'POST',
 body: inputBody
});
```

```

 body: inputBody,
 headers: headers
 })
 .then(function(res) {
 return res.json();
}).then(function(body) {
 console.log(body);
});

```

POST /wallet/withdraw/submit

確認虛擬貨幣出金

確認虛擬貨幣出金

#### Body parameter

```
{
 "userId": "string",
 "currency": "ETH",
 "chainId": 1,
 "amount": 0,
 "address": "string",
 "phoneVerify": "string"
}
```

## Parameters

| Name          | In   | Type   | Required | Description             |
|---------------|------|--------|----------|-------------------------|
| body          | body | object | true     | user id                 |
| » userId      | body | string | true     | none                    |
| » currency    | body | string | true     | 虛擬貨幣名稱                  |
| » chainId     | body | string | true     | <a href="#">匯款公鏈 ID</a> |
| » amount      | body | number | true     | none                    |
| » address     | body | string | true     | none                    |
| » phoneVerify | body | string | true     | 手機驗證碼                   |

## Detailed descriptions

» **chainId:** [匯款公鏈 ID](#)

1 for Ethereum Mainnet

#### Example responses

## 403 Response

```
{
 "currency": "api.wallet.withdraw.submit.error.currency.notExist",
 "chainId": "api.wallet.withdraw.submit.error.chainId.notExist",
 "amount": "api.wallet.withdraw.submit.error.amount.overMax",
 "address": "api.wallet.withdraw.submit.error.address.format",
 "phoneVerify": "api.wallet.withdraw.submit.error.phoneVerify.notMatch"
}
```

## Responses

| Status  | Meaning                      | Description    | Schema                    |
|---------|------------------------------|----------------|---------------------------|
| 200     | <a href="#">OK</a>           | OK             | None                      |
| 401     | <a href="#">Unauthorized</a> | 登入資訊錯誤         | None                      |
| 403     | <a href="#">Forbidden</a>    | 提領資訊錯誤         | Inline                    |
| default | Default                      | Error response | <a href="#">BaseError</a> |

## Response Schema

Status Code **403**

回傳對應欄位的錯誤訊息/代號

| Name          | Type   | Required | Restrictions | Description |
|---------------|--------|----------|--------------|-------------|
| » currency    | string | false    | none         | none        |
| » chainId     | string | false    | none         | none        |
| » amount      | string | false    | none         | none        |
| » address     | string | false    | none         | none        |
| » phoneVerify | string | false    | none         | none        |

To perform this operation, you must be authenticated by means of one of the following methods: BaseApiKey

## Schemas

### BaseAny

```
{}
```

## Properties

*None*

## BaseArrayAny

---

```
[
 {}
]
```

## Properties

| Name             | Type                      | Required | Restrictions | Description |
|------------------|---------------------------|----------|--------------|-------------|
| <i>anonymous</i> | <a href="#">[BaseAny]</a> | false    | none         | none        |

## BaseError

---

```
{
 "code": 0,
 "message": "string",
 "data": {}
}
```

## Properties

| Name    | Type                    | Required | Restrictions | Description   |
|---------|-------------------------|----------|--------------|---------------|
| code    | integer                 | true     | none         | Error code    |
| message | string                  | false    | none         | Error Message |
| data    | <a href="#">BaseAny</a> | false    | none         | none          |

## CartShoppingModel

---

```
{
 "hash": "string",
 "userId": "string",
 "list": [
 {
 "chainId": 0,
 }
]
}
```

```

 "contract": "string",
 "collection": "string",
 "id": 0,
 "nftName": "string",
 "shop": "string",
 "currency": "string",
 "price": 0,
 "enable": true
 }
],
"cost": [
 {
 "type": "string",
 "currency": "string",
 "amount": 0
 }
],
"wallet": [
 {
 "currency": "string",
 "balance": 0,
 "enable": true,
 "willPay": 0
 }
],
"payment": [
 {
 "currency": "string",
 "amount": 0,
 "payBy": [
 {
 "currency": "string",
 "used": 0,
 "rate": 0,
 "amount": 0
 }
]
 }
]
}

```

## Properties

| Name   | Type     | Required | Restrictions | Description         |
|--------|----------|----------|--------------|---------------------|
| hash   | string   | false    | none         | 目前資料 hash(防呆,更新時比對) |
| userId | string   | false    | none         | none                |
| list   | [object] | false    | none         | none                |

| Name         | Type     | Required | Restrictions | Description   |
|--------------|----------|----------|--------------|---------------|
| » chainId    | integer  | false    | none         | none          |
| » contract   | string   | false    | none         | none          |
| » collection | string   | false    | none         | 集合名稱          |
| » id         | integer  | false    | none         | none          |
| » nftName    | string   | false    | none         | none          |
| » shop       | string   | false    | none         | none          |
| » currency   | string   | false    | none         | none          |
| » price      | number   | false    | none         | none          |
| » enable     | boolean  | false    | none         | none          |
| cost         | [object] | false    | none         | none          |
| » type       | string   | false    | none         | gas/price/fee |
| » currency   | string   | false    | none         | none          |
| » amount     | number   | false    | none         | none          |
| wallet       | [object] | false    | none         | none          |
| » currency   | string   | false    | none         | none          |
| » balance    | number   | false    | none         | none          |
| » enable     | boolean  | false    | none         | none          |
| » willPay    | number   | false    | none         | none          |
| payment      | [object] | false    | none         | none          |
| » currency   | string   | false    | none         | none          |
| » amount     | number   | false    | none         | none          |
| » payBy      | [object] | false    | none         | none          |
| »» currency  | string   | false    | none         | none          |
| »» used      | number   | false    | none         | none          |
| »» rate      | number   | false    | none         | none          |
| »» amount    | number   | false    | none         | none          |

# CartSellingModel

---

```
{
 "hash": "string",
 "userId": "string",
 "list": [
 {
 "chainId": 0,
 "contract": "string",
 "collection": "string",
 "id": 0,
 "nftName": "string",
 "shops": [
 {
 "shop": "string",
 "currency": "string",
 "price": 0,
 "enable": true
 }
]
 }
],
 "cost": [
 {
 "type": "string",
 "currency": "string",
 "amount": 0
 }
],
 "wallet": [
 {
 "currency": "string",
 "balance": 0,
 "enable": true,
 "willPay": 0
 }
],
 "payment": [
 {
 "currency": "string",
 "amount": 0,
 "payBy": [
 {
 "currency": "string",
 "used": 0,
 "rate": 0,
 "amount": 0
 }
]
 }
]
}
```

```
]
}
```

## Properties

| Name         | Type     | Required | Restrictions | Description         |
|--------------|----------|----------|--------------|---------------------|
| hash         | string   | false    | none         | 目前資料 hash(防呆,更新時比對) |
| userId       | string   | false    | none         | none                |
| list         | [object] | false    | none         | none                |
| » chainId    | integer  | false    | none         | none                |
| » contract   | string   | false    | none         | none                |
| » collection | string   | false    | none         | 集合名稱                |
| » id         | integer  | false    | none         | none                |
| » nftName    | string   | false    | none         | none                |
| » shops      | [object] | false    | none         | none                |
| »» shop      | string   | false    | none         | none                |
| »» currency  | string   | false    | none         | none                |
| »» price     | number   | false    | none         | none                |
| »» enable    | boolean  | false    | none         | none                |
| cost         | [object] | false    | none         | none                |
| » type       | string   | false    | none         | gas / price / fee   |
| » currency   | string   | false    | none         | none                |
| » amount     | number   | false    | none         | none                |
| wallet       | [object] | false    | none         | none                |
| » currency   | string   | false    | none         | none                |
| » balance    | number   | false    | none         | none                |
| » enable     | boolean  | false    | none         | none                |
| » willPay    | number   | false    | none         | none                |
| payment      | [object] | false    | none         | none                |
| » currency   | string   | false    | none         | none                |

| Name        | Type     | Required | Restrictions | Description |
|-------------|----------|----------|--------------|-------------|
| » amount    | number   | false    | none         | none        |
| » payBy     | [object] | false    | none         | none        |
| »» currency | string   | false    | none         | none        |
| »» used     | number   | false    | none         | none        |
| »» rate     | number   | false    | none         | none        |
| »» amount   | number   | false    | none         | none        |

## InvoiceStatus

```
{
 "userId": "string",
 "invoiceType": "string",
 "invoiceId": "string",
 "status": "success",
 "createOn": 0,
 "jobs": [
 {
 "jobType": "string",
 "jobId": "string",
 "data": {},
 "status": "success",
 "finishOn": 0,
 "description": "string"
 }
]
}
```

## Properties

| Name        | Type     | Required | Restrictions | Description                     |
|-------------|----------|----------|--------------|---------------------------------|
| userId      | string   | false    | none         | none                            |
| invoiceType | string   | false    | none         | 訂單類型                            |
| invoiceId   | string   | false    | none         | 訂單編號(ID)                        |
| status      | string   | false    | none         | none                            |
| createOn    | number   | false    | none         | 建立時間                            |
| jobs        | [object] | false    | none         | none                            |
| » jobType   | string   | false    | none         | 協同 /全散 /上大壁 /下大壁 /左上壁 /右上壁 /左下壁 |

| Property      | Type   | Default  | None         | Description |
|---------------|--------|----------|--------------|-------------|
| Name          | Type   | Required | Restrictions |             |
| » jobId       | string | false    | none         | none        |
| » data        | object | false    | none         | none        |
| » status      | string | false    | none         | none        |
| » finishOn    | number | false    | none         | none        |
| » description | string | false    | none         | none        |

## Enumerated Values

| Property | Value      |
|----------|------------|
| status   | success    |
| status   | fail       |
| status   | partial    |
| status   | processing |
| status   | waiting    |
| status   | error      |
| status   | pause      |
| status   | cancel     |
| status   | success    |
| status   | fail       |
| status   | partial    |
| status   | processing |
| status   | waiting    |
| status   | error      |
| status   | pause      |
| status   | cancel     |

## MintSummaryModel

```
{
 "id": "string",
```

```

"hash": "string",
"createOn": 0,
"banner": "string",
"image": "string",
"status": "string",
"title": "string",
"mintOn": 0,
"currency": "string",
"price": 0,
"totalMints": 0,
"totalSupply": 0,
"totalMinted": 0
}

```

## Properties

| Name        | Type    | Required | Restrictions | Description  |
|-------------|---------|----------|--------------|--------------|
| id          | string  | false    | none         | none         |
| hash        | string  | false    | none         | none         |
| createOn    | integer | false    | none         | none         |
| banner      | string  | false    | none         | 大圖連結         |
| image       | string  | false    | none         | 小圖/NFT示意圖連結  |
| status      | string  | false    | none         | none         |
| title       | string  | false    | none         | none         |
| mintOn      | integer | false    | none         | none         |
| currency    | string  | false    | none         | none         |
| price       | number  | false    | none         | none         |
| totalMints  | integer | false    | none         | 站內 mint 總次數  |
| totalSupply | integer | false    | none         | 合約總供應量       |
| totalMinted | integer | false    | none         | 合約已 mint 總數量 |

## MintDetailModel

```
{
"id": "string",
"hash": "string",
"createOn": 0,
```

```

"isLock": true,
"isPublish": true,
"editable": true,
"mintOn": 0,
"banner": "string",
"image": "string",
"description": "string",
"homepage": "string",
"communities": [
 {
 "platform": "string",
 "url": "string"
 }
],
"status": "string",
"title": "string",
"currency": "string",
"price": 0,
"collection": "string",
"chainId": 0,
"contract": "string",
"function": "string",
"argument": {
 "fix": 0,
 "min": 0,
 "max": 0
},
"expected": 0
}

```

## Properties

| Name      | Type    | Required | Restrictions | Description      |
|-----------|---------|----------|--------------|------------------|
| id        | string  | false    | none         | none             |
| hash      | string  | false    | none         | none             |
| createOn  | integer | false    | none         | none             |
| isLock    | boolean | false    | none         | 是否已鎖定            |
| isPublish | boolean | false    | none         | 是否已上架(可查詢)       |
| editable  | boolean | false    | none         | 是否可編輯, 伺服器視狀態給予值 |
| mintOn    | integer | false    | none         | 合約可執行鑄造時間        |
| banner    | string  | false    | none         | 大圖連結             |
| image     | string  | false    | none         | 小圖/NFT示意圖連結      |

| description<br>Name | Type     | false<br>Required | none<br>Restrictions | 項目及鑄造說明文字                           | Descripti |
|---------------------|----------|-------------------|----------------------|-------------------------------------|-----------|
| homepage            | string   | false             | none                 | 官方網站連結                              |           |
| communities         | [object] | false             | none                 | none                                |           |
| » platform          | string   | false             | none                 | discord / facebook / twitter / gitl |           |
| » url               | string   | false             | none                 | none                                |           |
| status              | string   | false             | none                 | none                                |           |
| title               | string   | false             | none                 | none                                |           |
| currency            | string   | false             | none                 | none                                |           |
| price               | number   | false             | none                 | 單價                                  |           |
| collection          | string   | false             | none                 | NFT 集合名稱                            |           |

|          |         |       |      |                    |
|----------|---------|-------|------|--------------------|
| chainId  | integer | false | none | none               |
| contract | string  | false | none | NFT 合約地址           |
| function | string  | false | none | 呼叫合約函數名稱           |
| argument | object  | false | none | 呼叫合約參數(鑄造數量)       |
| » fix    | integer | false | none | none               |
| » min    | integer | false | none | none               |
| » max    | integer | false | none | none               |
| expected | integer | false | none | 預期收到NFT數量(不參與計算,在無 |