CX API Manual

**Contents**

1. [**업데이트 내역**](#_bookmark0)  
   (Update History)
   1. [**게임코드표**](#_bookmark1)  
      (Game code table)
   2. [**오퍼레이터 등록 요청**](#_bookmark2)  
      (Request for operator registration)
   3. [**API**](#_bookmark3)
      1. [**회원 확인**](#_bookmark5)  
         (Member Confirmation)
      2. [**회원 생성**](#_bookmark6)  
         (Creating a Member)
      3. [**게임사 요청**](#_bookmark7)  
         (Request from the game company)
      4. [**게임 목록 요청**](#_bookmark8)  
         (Request game list)
      5. [**게임 시작 URL 요청**](#_bookmark9)  
         (Request game start URL)
      6. [**보유 알 조회**](#_bookmark10)  
         (Inquiry of retained eggs)
      7. [**HOT 게임 리스트**](#_bookmark11)   
         (HOT Game List)
   4. [**CALLBACK**](#_bookmark12)
      1. [**회원 확인 (오퍼레이터가 만들어야 하는 부분)**](#_bookmark13)   
         Member confirmation (the part that the operator must make)
      2. [**잔액 확인 (오퍼레이터가 만들어야 하는 부분)**](#_bookmark14)  
         Check the balance (the part that the operator has to make)
      3. [**베팅 (오퍼레이터가 만들어야 하는 부분)**](#_bookmark15)  
         **Betting (what the operator has to make)**
      4. [**결과 (오퍼레이터가 만들어야 하는 부분)**](#_bookmark16)  
         **Results (what the operator should make)**
      5. [**환불 (오퍼레이터가 만들어야 하는 부분)**](#_bookmark17)  
         **Refund (the part that the operator should make)**

# Update history

This is the update history part.**Summary**

**History**

|  |  |  |
| --- | --- | --- |
| **version** | **Date** | **Description** |
| V1.0.0 |  | The initial version |
| V1.2.0 |  | [3-6. Check the having eggs] Add |
| V1.3.0 |  | [3-5. Request URL for game start] Add parameter (ip, bet\_limit) |
| V1.4.0 |  | [3-7. HOT Game list] Add |
|  |  |  |
|  |  |  |

**Table for Game Code**

|  |  |
| --- | --- |
| **gamecode** | **Description** |
| 1 | EVOLUTION |
| 2 | QTech |
| 3 | EVOPLAY |
| 4 | CQ9 |
| 5 | ALL BET |
| 6 | YGG |
| 7 | PragmaticPlay |
| 8 | DreamGame |
| 9 | Boongo |
| 10 | Real Time Gameing |
| 11 | WM |
| 12 | WM Live |
| 13 | VIVO |
| 14 | Amatic |
| 15 | Green Tube |
| 16 | Quick Spin |
| 17 | Wazdan |
| 18 | EGT Jackpot |
| 19 | Classic Casino |

|  |  |
| --- | --- |
| 20 | Netent |
| 21 | EZugi |
| 22 | Red TIger |
| 23 | Playson |
| 24 | PlayStar |
| 25 | Habanero |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Request for operator registration

**Summary**

Operator registration request. The following information is required when requesting registration.

|  |  |
| --- | --- |
|  | 1. **P of the server to which the API will be called** 2. **Service Domain** 3. **Callback URL**    * **Member check URL**    * **Balance Check URL**    * **Betting URL**    * **Betting result URL**    * **Refund URL** |

# API

**Summary**

The URL of the request server. The URL is subject to change.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **URL** | [**http://smapi.enjoycx.com/**](http://smapi.enjoycx.com/) | | | |
| **HASH** | MD5( OPKEY + "|" + parameter. ) | | | |
|  | | | |
|  | OPKEY | "29b25cec12185642c4b2ab6900ffd95c" |  |
| 파라미터 | opkey=29b25cec12185642c4b2ab6900ffd95c&userid=user001&nick  =user001 |  |
| HASH | MD5( "29b25cec12185642c4b2ab6900ffd95c" + "|" + "opkey=29b25cec12185642c4b2ab6900ffd95c&userid=user001&nic k=user001" ) = 29B4892D15B37726164F5CD80461609A |  |
| Final parameter | opkey=29b25cec12185642c4b2ab6900ffd95c&userid=user001&nick  =user001&hash=29B4892D15B37726164F5CD80461609A |  |
| **Memebr ID** | 20byte | | | |
| **nick** | 20byte | | | |

3-1. Member Confirmation

**Summary**

API to verify membership. The return result is returned to JSON.

**Request URL**

|  |  |
| --- | --- |
| **URL** | /account?opkey=OPKEY&userid=회원ID&hash=HASH |
| **Method** | GET |

**Request Parameters**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| userid | True | String | Member ID |
| hash | True | String | [위에 방법대로 조합된 HASH값](#_bookmark4) HASH values combined by the above method |

**Response**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | | |
| **Example** | { result: 결과, msg: 메시지, data: { id: 회원ID, nick: 회원닉네임 } } | | | | | | |
| **Field** | **Required** | | **Type** | | **Description** | | |
| result | True | | Int | | Determines the presence or absence of a member.  If you have a member, send 1 and if not, send 10001. If you receive 10001, request to create a member. | | |
| msg | false | | String | | Send Message. | | |
| data | false | | JSON | | Sends result data to JSON. | | |
|  | | | | | | |
|  | **Field** | **Required** | **Type** | | **Description** |  |
| id | True | String | | Returns the ID. |  |
| nick | True | String | | Returns the Nickname |  |

**Example**

php

## 3-2. Creating a Member

**Summary**

The API that creates the member. The return result is returned to JSON.

**Request URL**

|  |  |
| --- | --- |
| **URL** | /create\_account?opkey=OPKEY&userid=회원ID&nick=회원닉네임&hash=HASH |
| **Method** | GET |

**Request Parameters**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| userid | True | String | Member ID |
| nick | True | String | Member Nickname |
| hash | True | String | [위에 방법대로 조합된 HASH값](#_bookmark4) HASH values combined by the above method |

**Response**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | | |
| **예시** | { result: 결과, msg: 메시지, data: { id: 회원ID, nick: 회원닉네임 } } | | | | | | |
| **Field** | **Required** | | **Type** | | **Description** | | |
| result | True | | Int | | Returns the result of member creation. Returns 1 if successful. (Return 1 even for members who already exist). | | |
| msg | false | | String | | Sending Message | | |
| data | false | | JSON | | Sends result data to JSON. | | |
|  | | | | | | |
|  | **Field** | **Required** | **Type** | | **Description** |  |
| id | True | String | | Returns ID |  |
| nick | True | String | | Returns Nickname |  |

**Example**

php

## 3-3. Request from the game company

**Summary**

API requesting game company. The return result is returned to JSON.

**Request URL**

|  |  |
| --- | --- |
| **SLOT** | |
| **URL** | /provlist?opkey=OPKEY&hash=HASH |
| **Method** | GET |

**Request Parameters**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| hash | True | String | [위에 방법대로 조합된 HASH값](#_bookmark4) HASH values combined by the above method |

**Response**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | | | | |
| **예시** | { result: 결과, msg: 메시지, data: { list: [ { code, name, type } ] } } | | | | | | | | |
| **Field** | **Required** | | **Type** | | **Description** | | | | |
| result | True | | Int | | Returns the result of a game company request. Returns 1 if successful. | | | | |
| msg | false | | String | | Sending Message | | | | |
| data | false | | JSON | | Sends result data to JSON. | | | | |
|  | | | | | | | | |
|  | **Field** | **Required** | **Type** | | **Description** | | |  |
| list | True | JSON | | Returns the List | | |  |
|  | | | | | |  |
| **Field** | **Required** | | **Type** | **Description** |  |
| code | True | | String | Game Company Code |
| name | True | | String | Name of Game Company |
| type | True | | Stirng | Game Type ( slot, live ) |

## 3-4. Request game list

**Summary**

API requesting game list. The return result is returned to JSON.

**Request URL**

|  |  |
| --- | --- |
| **SLOT** | |
| **URL** | /gamelist?opkey=OPKEY&thirdpartycode=게임사코드&hash=HASH |
| **Method** | GET |

**Request Parameters**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| thirdpartycode | True | String | Game Company Code |
| hash | True | String | [위에 방법대로 조합된 HASH값](#_bookmark4)  HASH values combined by the above method |

**Response**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | | | | |
| **예시** | { result: 결과, msg: 메시지, data: { list: [ { code, name\_eng, name\_kor, type,  is\_desktop, is\_mobile, img\_1 } ] } } | | | | | | | | |
| **Field** | **Required** | | **Type** | | **Description** | | | | |
| result | True | | Int | | Returns the result of a game list request. Returns 1 if successful. | | | | |
| msg | false | | String | | Sending Message | | | | |
| data | false | | JSON | | Sends result data to JSON. | | | | |
|  | | | | | | | | |
|  | **Field** | **Required** | **Type** | | **Description** | | |  |
| list | True | JSON | | Returns the list | | |  |
|  | | | | | |  |
| **Field** | **Required** | | **Type** | **Description** |  |
| code | True | | String | Game code |
| name\_eng | True | | String | English Name |
| name\_kor | True | | String | Korean Name |
| type | True | | String | Game type |
| is\_desktop | True | | String | Desktop Support |
| is\_mobile | True | | String | Mobile Support |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | img\_1 | True | String | Thumbnail image |  |  |

## 3-5. Request game start URL

**Summary**

API requesting game start. The return result is returned to JSON.

**We recommend that you start the game with a new window. You have to redirect the game start URL right after you get it back.**

**Request URL**

|  |  |
| --- | --- |
| **URL** | /play?opkey=OPKEY&userid=회원ID&thirdpartycode=게임사코드&gamecode=  게임코드&platform=플렛폼&ip=아이피&bet\_limit=벳타입&hash=HASH |
| **Method** | GET |

**Request Parameters**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| userid | True | Stirng | Member ID |
| thirdpartycode | True | Int | Game company Code |
| gamecode | True | String | Game code |
| platform | True | String | Execution Platform |
| ip | False | String | Member’s device IP |
| bet\_limit | False | String | Code for Betting money limited  (Refer to betlimit.xlsx) |
| hash | True | String | [위에 방법대로 조합된 HASH값](#_bookmark4) HASH values combined by the above method |

**Response**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | | |
| **예시** | { result: 결과, msg: 메시지, data: { link: 게임 시작 URL } } | | | | | | |
| **Field** | **Required** | | **Type** | | **Description** | | |
| result | True | | Int | | Returns the result of the game start URL request.  Returns 1 if successful.  (If the result is 1 (successful), redirect to the data.link you received.) | | |
| msg | false | | String | | Sending Message | | |
| data | false | | JSON | | Sends result data to JSON. | | |
|  |  | | | | | | |
|  | **Field** | **Required** | **Type** | | **Description** |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | link | True | String | Returns the game start URL. |  |

## 3-6. Inquiry of retained eggs

This is the part where you look up your eggs.**Summary**

**Request URL**

|  |  |
| --- | --- |
| **URL** | /current/r?opkey=OPKEY&hash=HASH |
| **Method** | GET |

**Request Parameters**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| hash | True | String | [위에 방법대로 조합된 HASH값](#_bookmark4)  HASH values combined by the above method |

**Response**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | | |
| **예시** | { result: 결과, msg: 메시지, data: { r: 10000 } | | | | | | |
| **Field** | **Required** | | | | **Type** | **Description** | |
| result | True | | | | Int | If successful, send 1. | |
| msg | false | | | | String | Sending Message | |
| data | True | | | | JSON | Returns the Data | |
|  |  | | | | | | |
|  | **Field** | **Type** | **Description** | | |  |
| r | Int | Currently having Eggs | | |

## 3-7. HOT Game list

**Summary**

API requesting Hot Game list in slot. Returned to JSON.

**Request URL**

|  |  |
| --- | --- |
| **URL** | /hotgames/top30?opkey=OPKEY&hash=HASH |
| **Method** | GET |

**Request Parameters**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| hash | True | String | HASH values combined by the above method |

**Response**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | | | | | | |
| **예시** | { result: 결과, msg: 메시지, data: { list: [ { code, name, type } ] } } | | | | | | | | | | |
| **Field** | **Required** | | **Type** | | **Description** | | | | | | |
| result | True | | Int | | Returns the result of a game company request. Returns 1 if successful. | | | | | | |
| msg | false | | String | | Sending Message | | | | | | |
| data | false | | JSON | | Sends result data to JSON. | | | | | | |
|  | | | | | | | | | | |
|  | **Field** | **Required** | **Type** | | | **Description** | | | |  |
| list | True | JSON | | | Returns the list | | | |  |
|  | | | | | | | |  |
| **Field** | | | **Required** | | **Type** | **Description** |  |
| rank | | | True | | Int | Ranking |
| thirdpartycode | | | True | | String | Game company code |
| thirdpartyname | | | True | | String | Name of Game company |
| code | | | True | | String | Game code |
| name | | | True | | String | Game name |
| img | | | True | | String | Thumbnail image |

## CALLBACK (the part that the operator must make)

**Summary**

As the game progresses, the CX game server will call back to the operator.

Callback will be conducted through the callback URL upon operator registration and response time is recommended not more than 500ms and must be less than 1 second.

If the time is delayed, the game company (Evolution, DG, PP, etc.) might recognize it as an error.

CX limits all communication wait to less than 1 second for smooth progress.

Among the callback data, token is for internal use of CX, so it is okay to ignore it.

* round\_id: Round ID, It cannot be a key because it changes every time the cycle passes.
* trans\_id: Transfer ID, Unique value generated each time a transaction occurs. Must be used as a key.
* The trans\_id generated during betting and the trans\_id generated during the result are different values, so please refer to it.
* When a refund occurs, trans\_id is the trans\_id of the bet to be refunded.
* In the case of slots, depending on the game company, multiple results can be concluded at once.
  + Usually, freespin occurs in freespin and the result does not occur after the last spin. Depending on the game company, freespin may result in one result per spin.
  + Depending on the game company, you can also receive the result of 0 won to end the game.
* For casinos, multi-betting games such as blackjack can have multiple bets on the same round\_id, where trans\_id is sent differently per case.
* Results may also have multiple results in the same round\_id. Trans\_id is also transmitted differently per case.

## 4-1. Member confirmation (the part that the operator must make)

This is the callback part to check the membership.**Summary**

**Request Parameters (CX => OP( 오퍼레이터 ))**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| userid | True | Stirng | Member ID |
| token | True | String | Member Token |
| third\_party | True | String | Name of Game company |

**Response (OP => CX)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | |
| **예시** | { result: 결과, balance: 잔액 } | | | | | |
| **Field** | **Required** | **Type** | **Description** | | | |
| result | True | Int | Returns the result of the membership verification URL request.  Returns 1 if successful.  Please refer to the table below and give the return value. | | | |
|  | **Code** | **Description** |  |
| 1 | Success |  |
| 10 | Non-existent users (including blocked members) |  |
| 99 | Other Error |  |
| balance | True | Int | Returns the account balance of the requested member. | | | |

## 4-2. Check the balance (the part that the operator has to make)

This is the callback part to check the balance.**Summary**

**Request Parameters (CX => OP( 오퍼레이터 ))**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| userid | True | Stirng | Member ID |
| token | True | String | Member Token |
| third\_party | True | String | Name of Game company |

**Response (OP => CX)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | |
| **예시** | { result: 결과, balance: 잔액 } | | | | | |
| **Field** | **Required** | **Type** | **Description** | | | |
| result | True | Int | Returns the result of the membership verification URL request.  Returns 1 if successful.  Please refer to the table below and give the return value. | | | |
|  | **Code** | **Description** |  |
| 1 | Success |  |
| 10 | Non-existent users (including blocked members) |  |
| 99 | Other Error |  |
| balance | True | Int | Returns the account balance of the requested member. | | | |

## 4-3. Betting (what the operator has to make)

It's the part where you get the bet.**Summary**

* If the same trans\_id is transmitted and duplication occurs, set [1: Success] for normal progress.

**Request Parameters (CX => OP( 오퍼레이터 ))**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| userid | True | Stirng | Member ID |
| token | True | String | Member Token |
| third\_party\_code | True | String | Game Company key |
| third\_party\_name | True | String | Name of Game Company |
| game\_code | True | String | Game Code |
| amount | True | String | Used money |
| round\_id | True | String | Round ID |
| trans\_id | True | String | Transaction ID |
| date | True | String | Date |
| game\_type | True | String | Game Type( slot, live ) |

**Response (OP => CX)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | |
| **예시** | { result: 결과, balance: 잔액 } | | | | | |
| **Field** | **Required** | **Type** | **Description** | | | |
| result | True | Int | Returns the result of the membership verification URL request.  Returns 1 if successful.  Please refer to the table below and give the return value. | | | |
|  | **Code** | **Description** |  |
| 1 | Success |  |
| 10 | Non-existent users (including blocked members) |  |
| 20 | Not enough Balance |  |
| 99 | Other Error |  |
| balance | True | Int | Returns the account balance of the requested member. | | | |

## 4-4. Results (what the operator should make)

The part where you receive the results.**Summary**

* If the same trans\_id is transmitted and duplication occurs, set [1: Success] for normal progress.

**Request Parameters (CX => OP( 오퍼레이터 ))**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| userid | True | Stirng | Member ID |
| token | True | String | Member Token |
| third\_party\_code | True | String | Game company Key |
| third\_party\_name | True | String | Game Company Name |
| game\_code | True | String | Game Code |
| amount | True | String | Used Money |
| round\_id | True | String | Round ID |
| trans\_id | True | String | Transaction ID |
| date | True | String | Date |
| game\_type | True | String | Game Type( slot, live ) |

**Response (OP => CX)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | |
| **예시** | { result: 결과, balance: 잔액 } | | | | | |
| **Field** | **Required** | **Type** | **Description** | | | |
| result | True | Int | Returns the result of the membership verification URL request.  Returns 1 if successful.  Please refer to the table below and give the return value. | | | |
|  | **Code** | **Description** |  |
| 1 | Success |  |
| 10 | Non-existent users (including blocked members) |  |
| 90 | Non-existent betting |  |
| 99 | Other Error |  |
| balance | True | Int | Returns the account balance of the requested member. | | | |

## 4-5. Refund (the part that the operator should make)

This is the refund processing part.**Summary**

* If a timeout occurs between CX and OP during betting or other errors occur, a refund of the bet will occur.
* If there is no betting, you can set the result to 90 and respond.

When a refund occurs, trans\_id is the trans\_id of the bet to be refunded.

**Request Parameters (CX => OP( 오퍼레이터 ))**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Required** | **Type** | **Description** |
| opkey | True | String | Operator Key |
| userid | True | Stirng | Member ID |
| token | True | String | Member Token |
| third\_party\_code | True | String | Game Company Key |
| third\_party\_name | True | String | Game Company Name |
| game\_code | True | String | Game Code |
| amount | True | String | Used Money |
| round\_id | True | String | Round ID |
| trans\_id | True | String | Transaction ID |
| date | True | String | Date |
| game\_type | True | String | Game Type( slot, live ) |

**Response (OP => CX)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **JSON** | | | | | | |
| **예시** | { result: 결과, balance: 잔액 } | | | | | |
| **Field** | **Required** | **Type** | **Description** | | | |
| result | True | Int | Returns the result of the membership verification URL request.  Returns 1 if successful.  Please refer to the table below and give the return value. | | | |
|  | **Code** | **Description** |  |
| 1 | Success |  |
| 10 | Non-existent users (including blocked members) |  |
| 90 | Non-existent betting |  |
| 99 | Other Error |  |
| balance | True | Int | Returns the account balance of the requested member. | | | |

- **End**