



대한항공 REST API 가이드

DARS(국내선 환불금 조회)  
REST API 가이드

---

Ver.1.1



목 차

<b>1. 주의해야 할 사항</b>	<b>1</b>
<b>2. 인증 방법 변경</b>	<b>1</b>
1) 인증 방법	1
2) API TOKEN 발급(Oauth2)	1
2-1) URL	1
2-2) Request Parameter	1
2-3) Response Parameter	2
2-4) Token 발급 후 API 서비스 호출 예시	2
<b>3. API 서비스 URL</b>	<b>3</b>
1) TOBE Domain	3
2) ASIS -> TOBE	3
3) Request, Response Type	3
3-1 ) IF : PAS_Dom_Refund_history_01_1_V1_01	3
3-2 ) IF : PAS_Dom_Refund_Calc_Request_01_1_V1_01	6
3-3) IF : PAS_Dom_Payment_history_01_1_V1_01	9
3-4) IF : PAS_Proration_Request_01_1_V1_01	10
4) Sample Code(JAVA)	13
4-1) TOKEN 발급	13
4-2) REST API 호출	14
5) X-Request-System 참고	15

## 1. 주의해야 할 사항

- 1) 인터페이스를 위해 제공되는 인증키와 비밀번호는 안전한 곳에 보관이 되어야 합니다
- 2) 인증키와 비밀번호가 노출 시 토파스를 통해 대한항공에 해당 사실을 통보해야 합니다
- 3) 인증 토큰은 최대한 재사용 해야 합니다

**인증키(client\_id)와 비밀번호(client\_secret):** 토파스에서 대한항공 담당자에게 발급 요청.

## 2. 인증 방법 변경

### 1) 인증 방법

ASIS 인증	TOBE 인증
Soap Header의 userName,password	API TOKEN 발급(4시간후 만료) → 2.호출하고자 하는 API 호출(발급된 토큰 사용)

### 2) API TOKEN 발급(Oauth2)

#### 2-1) URL

- DEV(개발환경) : <https://apidev.koreanair.com/oauth2/v1/token>
- PRD(운영환경) : <https://api.koreanair.com/oauth2/v1/token>

#### 2-2) Request Parameter

구분	Name	Type	필수	Default Value / Description
Method	<b>POST</b>		Y	
Header	Content-Type	string	Y	"application/x-www-form-urlencoded"
Header	Authorization	string	Y	Basic base64(client_id:client_secret) <b>ex)</b> client_id = AaBbCcDdEe client_secret = aaabbb  Basic base64(AaBbCcDdEe:aaabbb) = <b>Basic QWFCYkNjRGRFZTphYWFiYml=</b>
Body	grant_type	string	Y	"client_credentials"

## 2-3) Response Parameter

구분	Name	Type	Default Value / Description
Body	token_type	string	"Bearer"
Body	access_token	string	Apigee Access Token
Body	expires_in	string	기본값 : 14400s (= 240 minutes = 4 hours)

## 2-4) Token 발급 후 API 서비스 호출 예시

- DEV : <https://apidev.koreanair.com/pss-pasinf-pri/api/v1/pasDomRefundHistory>

구분	Name	Type	필수	Default Value / Description
Method	<b>POST</b>		Y	
Header	Content-Type	string	Y	"application/json"
Header	Authorization	string	Y	Bearer {token}
Header	X-Request-System	string	Y	<a href="#">3. 5) X-Request-System 참고</a>
Body				아래 정의된 Request

\* token : 발급 받은 token 사용

### 3. API 서비스 URL

#### 1) TOBE Domain

- DEV : <https://apidev.koreanair.com/pss-pasinf-pri>
- PRD : <https://api.koreanair.com/pss-pasinf-pri>

#### 2) ASIS -> TOBE

Interface Name	ASIS URL	TOBE URL
PAS_Dom_Refund_history_01_1_V1_01	/PAS.SRV.WS01 BODY : PAS_Dom_Refund_history	(POST) <a href="#">/api/v1/pasDomRefundHistory</a>
PAS_Dom_Refund_Calc_Request_01_1_V1_01	/PAS.SRV.WS01 BODY : PAS_Dom_Refund_Calc_Request	(POST) <a href="#">/api/v1/pasDomRefundCalcRequest</a>
PAS_Dom_Payment_history_01_1_V1_01	/PAS.SRV.WS01 BODY : PAS_Dom_Payment_history	(POST) <a href="#">/api/v1/pasDomPaymentHistory</a>
PAS_Proration_01_1_V1_01	/PAS.SRV.WS01 BODY : PAS_Proration_Request	(POST) <a href="#">/api/v1/pasProrationRequest</a>

#### 3) Request, Response Type

##### 3-1) IF : PAS\_Dom\_Refund\_history\_01\_1\_V1\_01

- Methods : POST
- URL : [/api/v1/pasDomRefundHistory](#)
- Request

Field	Type	Size	필수	설명	Sample
tktno	String		Y	티켓번호	1804201309400

JSON

```
{
  "tktno": "18042013109400111111"
```

```
}

```

- Response

Field	Type	Size	필수	설명	Sample
refundErrInfo	Object			환불 에러 정보	
errCd	String	2		에러코드	00
errMsg	String			에러메시지	Success
refundHistoryInfo	Array			환불 이력 정보 목록	[]
fopSeq	String			fop순번	
currency	String			통화	
payAmt	double			금액	53000.0
fopCd	String			결재수단코드	MS
ccBrndCd	String			브랜드 코드	
ccNo	String			카드번호	
ccVldYn	String			유효 년월	
cclnsMths	String			할부 개월	
ccApvNo	String			승인번호	
skypassNumber	String			skypass번호	
fopContents	String			수단 내용	CPN*MCPN25C DM927799487
coverRefundYN				커버 환불 여부	
coverRefundYn	String			커버 환불 여부(Y/N)	N
rfndInfo				환불 정보	
domRfndCalcCd	String			국내선 환불 계산 코드	M

JSON

```
{
  "refundErrInfo": {
    "errCd": "00",
    "errMsg": "Success"
  },
  "refundHistoryInfo": [
    {
      "fopSeq": "",
      "currency": "",
      "payAmt": 53000.0,
      "fopCd": "MS",
      "ccBrndCd": "",
      "ccNo": "",
      "ccVldYn": "",
      "ccInsMths": "",
      "ccApvNo": "",
      "skypassNumber": "",
      "fopContents": "CPN*MCPN25CDM92779948711111111"
    }
  ],
  "coverRefundYN": {
    "coverRefundYn": "N"
  },
}
```

```

"rfndInfo": {
  "domRfndCalcCd": "M"
}
}

```

## 3-2 ) IF : PAS\_Dom\_Refund\_Calc\_Request\_01\_1\_V1\_01

- Methods : POST
- URL : /api/v1/pasDomRefundCalcRequest
- Request

Field		Type	Size	필수	설명	Sample
refundCalcRequestInfo		Array		Y		[]
	tktno	String		Y	티켓번호	180242249 8102
	noShowFee	String		Y	노쇼 수수료	0
	usedFareAmt	double			사용된 운임 금액	0.0
	loyInfo	String		Y	로열티 정보	0
	refundAmt	double			환불 금액	133700.0
	domRefundMthdCd	String		Y	국내선 환불 방법 코드	K
	canxFee	double			취소 수수료	0.0
	taxRefund	double			세금 환불	11700.0
	nonRefAmt	double			환불 불가 금액	0.0
	refMile	double			환불 마일리지	0.0
	misFee	double			기타 수수료	0.0
	usedYr	double			사용 연도	0.0

JSON

```
[
  {
    "tktno": "180242249810211111111111",
    "noShowFee": "0",
    "usedFareAmt": 0.0,
    "loyInfo": "0",
    "refundAmt": 133700.0,
    "domRefundMthdCd": "K",
    "canxFee": 0.0,
    "taxRefund": 11700.0,
    "nonRefAmt": 0.0,
    "refMile": 0.0,
    "misFee": 0.0,
    "usedYr": 0.0
  }
]
```

- Response

Field		Type	Size	필수	설명	Sample
refundCalcResponseInfo		Array				[]
	refundCalcErrInfo				에러 정보	
	errCd	String	2		에러코드	00
	errMsg	String			에러메시지	Success

	refundCalcInfo					환불 계산 정보 목록	□
		fopSeq	String			결제 수단 순번	FOP01
		payAmt	int			결제 금액	133700
		fopCd	String			결제 수단 코드	CA
		ccBrndCd	String			브랜드 코드	
		ccNo	String			카드번호	
		ccVldYn	String			유효 년월	
		ccInsMths	String			할부 개월	
		ccApvNo	String			승인번호	
		ffpNo	String			FFP 번호	
		carrCd	String			항공사 코드	
		fopContents	String			수단 내용	
		salesindicator	String			판매 지표	
		tktdDate	String			티켓 발권 날짜	

JSON

```
{
  "refundCalcResponseInfo": [
    {
      "refundCalcErrInfo": {
        "errCd": "00",
        "errMsg": "SUCESS"
      },
      "refundCalcInfo": [
        {
          "fopSeq": "FOP01",
          "payAmt": 133700,
          "fopCd": "CA",
          "ccBrndCd": "",
          "ccNo": "",
          "ccVldYn": "",
          "ccInsMths": "",
          "ccApvNo": "",
          "ffpNo": ""
        }
      ]
    }
  ]
}
```



historyInfo		Array			이력목록	[]
	fopSeq	String			순번	1
	currency	String			통화	KRW
	payAmt	double			결제 금액	100.0
	fopCd	String			결제 수단 코드	CA
	ccBrndCd	String			브랜드 코드	
	ccNo	String			카드번호	
	ccVldYn	String			유효 년월	
	ccInsMths	String			할부 개월	
	ccApvNo	String			승인번호	
	fopContents	String			수단 내용	
	tktno	String			티켓번호	1802422498104

JSON

```
{
  "errInfo": {
    "errCd": "00",
    "errMsg": "Success"
  },
  "historyInfo": [
    {
      "fopSeq": "1",
      "currency": "KRW",
      "payAmt": 100.0,
      "fopCd": "CA",
      "ccBrndCd": "",
      "ccNo": "",
      "ccVldYn": "",
      "ccInsMths": "",
      "ccApvNo": "",
      "fopContents": "",
      "tktno": "18021422498104111111"
    }
  ]
}
```

3-4) IF : PAS\_Proration\_Request\_01\_1\_V1\_01

- Methods : POST
- URL : /api/v1/pasProrationRequest
- Request

Field	Type	Size	필수	설명	Sample
requestEntry	String		Y	요청Entry	PRC/R-ICN KE BKK KE ICN/F-Y/C-KRW/A-500000/T-09FEB26

JSON

```
{
  "requestEntry": "PRC/R-ICN KE BKK KE
  ICN/F-Y/C-KRW/A-500000/T-09FEB26"
}
```

- Response

Field		Type	Size	필수	설명	Sample
commandCrypticResponseInfo		Object				
	commandCrypticErrInfo	Object			에러 정보	
	errCd	String	2		에러 코드	00
	errMsg	String			에러 메시지	Success
	commandCrypticInfo				Cryptic 정보	
	responseData	String			응답 데이터	/ \$FARE TYPE: Y AMOUNT: 500000 CURRENCY: KRW DATE: 09FEB26\n\n ORG DST CAR PF PV BA \n-----\n ICN BKK KE 2721 249968 \n BKK ICN KE 2721 249968 \n\nQUOTIENT :

							0.06237\nBASIC SALES PRORATION FOR INFORMATION ONLY\nSTRAIGHT RATE APPLIED\n
--	--	--	--	--	--	--	--

JSON

```
{
  "commandCrypticResponseInfo": {
    "commandCrypticErrInfo": {
      "errCd": "00",
      "errMsg": "SUCCESS"
    },
    "commandCrypticInfo": {
      "responseData": "/$FARE TYPE: Y  AMOUNT: 500000  CURRENCY: KRW
DATE: 09FEB26\n\n ORG  DST  CAR  PF          PV          BA
\n-----\n ICN  BKK  KE   2721
249968  \n BKK  ICN  KE   2721    249968  \n\nQUOTIENT : 0.06237\nBASIC
SALES PRORATION FOR INFORMATION ONLY\nSTRAIGHT RATE APPLIED\n "
    }
  }
}
```

## 4) Sample Code(JAVA)

### 4-1) TOKEN 발급

```
Java
import java.util.Base64;
import java.util.concurrent.TimeUnit;
import org.springframework.http.*;
import org.springframework.web.client.RestTemplate;
import com.fasterxml.jackson.databind.JsonNode;
import com.fasterxml.jackson.databind.ObjectMapper;

/**
 * Token Test (Standalone)
 */
public class TokenTest {

    private static final String TOKEN_URL =
"https://apidev.koreanair.com/oauth2/v1/token"; //dev url
    private static final String CLIENT_ID = "test"; //발급받은 client_id
    private static final String CLIENT_SECRET = "testSecret"; //발급받은
client_secret

    public static void main(String[] args) {
        try {
            RestTemplate restTemplate = new RestTemplate();
            ObjectMapper objectMapper = new ObjectMapper();

            // 1. Create Authorization header
            String credentials = CLIENT_ID + ":" + CLIENT_SECRET;
            String encodedCredentials =
Base64.getEncoder().encodeToString(credentials.getBytes());
            String authorization = "Basic " + encodedCredentials;

            // 2. Set headers
            HttpHeaders headers = new HttpHeaders();
            headers.setContentType(MediaType.APPLICATION_FORM_URLENCODED);
            headers.set("Authorization", authorization);

            // 3. Set body
            String body = "grant_type=client_credentials";

            // 4. Create request entity
            HttpEntity<String> entity = new HttpEntity<>(body, headers);

            // 5. Call API
            ResponseEntity<String> response = restTemplate.exchange(
                TOKEN_URL,
                HttpMethod.POST,
                entity,
                String.class
            );
        }
    }
}
```

```

// 6. Parse response
System.out.println("Status: " + response.getStatusCode());
System.out.println("Response Body: " + response.getBody());

JsonNode jsonNode = objectMapper.readTree(response.getBody());
String accessToken = jsonNode.get("access_token").asText();
int expiresIn = jsonNode.get("expires_in").asInt();

System.out.println("=== Token Info ===");
System.out.println("Access Token: " + accessToken);
System.out.println("Expires In: " + expiresIn + " seconds (" +
(expiresIn / 60) + " minutes)");
System.out.println();
System.out.println("✓ Token retrieved successfully!");

} catch (Exception e) {
    System.err.println("✗ Token request failed: " + e.getMessage());
    e.printStackTrace();
}
}
}

```

#### 4-2) REST API 호출

```

Java
import org.springframework.http.*;
import org.springframework.web.client.RestTemplate;

/**
 * PAS DOM Refund History API Test (Standalone)
 */
public class PasDomRefundHistoryTest {

    private static final String BASE_URL =
"https://apidev.koreanair.com/pss-pasinf-pri"; //dev url
    private static final String API_ENDPOINT =
"/api/v1/pasDomRefundHistory";
    private static final String TOKEN = "issuedToken"; //위 API를 통해 발급
받은 유효한 token 값

    public static void main(String[] args) {
        try {
            RestTemplate restTemplate = new RestTemplate();

            // Request body
            String requestBody = "{\"tktno\":\"1804201309400\"}";

            // HTTP Headers
            HttpHeaders headers = new HttpHeaders();

```

```
headers.setContentType(MediaType.APPLICATION_JSON);
headers.set("Authorization", "Bearer "+TOKEN);
headers.set("X-Request-System", "AAA");
// 3. 5) X-Request-System 참고표 참조

// HTTP Entity
HttpEntity<String> entity = new HttpEntity<>(requestBody,
headers);

// API Call
System.out.println("=== PAS DOM Refund History Request ===");
System.out.println("URL: " + BASE_URL + API_ENDPOINT);
System.out.println("Request Body: " + requestBody);

ResponseEntity<String> response = restTemplate.exchange(
    BASE_URL + API_ENDPOINT,
    HttpMethod.POST,
    entity,
    String.class
);

// Response
System.out.println("Status Code: " + response.getStatusCode());
System.out.println("Response Body: " + response.getBody());

} catch (Exception e) {
    System.err.println("API 호출 실패: " + e.getMessage());
    e.printStackTrace();
}
}
```

## 5) X-Request-System 참고

- 별도 개별 통지