

HTTP 어댑터 사용자 안내서

AnyLink 7

TMAXSOFT

저작권 공지

Copyright 2025. TmaxSoft Co., Ltd. All Rights Reserved.

회사 정보

(주)티맥스소프트

주소 : 경기도 성남시 분당구 황새울로258번길 29, 티맥스수내타워 8-9층

기술 서비스 센터: 1544-8629

홈페이지: <https://www.tmaxsoft.com>

제한된 권리

이 소프트웨어(Tmax AnyLink®) 사용설명서와 프로그램은 저작권법과 국제 조약에 의해 보호됩니다. 사용설명서와 프로그램은 TmaxSoft Co., Ltd.와의 사용권 계약 하에서만 사용할 수 있으며, 사용설명서는 사용권 계약의 범위 내에서만 배포 또는 복제할 수 있습니다. 이 사용설명서의 전부 또는 일부를 TmaxSoft의 사전 서면 동의 없이 전자, 기계, 녹음 등의 수단으로 전송, 복제, 배포하거나 2차적 저작물을 작성할 수 없습니다.

이 소프트웨어 사용설명서와 프로그램의 사용권 계약은 어떠한 경우에도 사용설명서 및 프로그램과 관련된 지적 재산권(등록 여부를 불문)을 양도하는 것으로 해석되지 않으며, 브랜드나 로고, 상표 등을 사용할 권한을 부여하지 않습니다. 사용설명서는 오로지 정보 제공만을 목적으로 하며, 이로 인한 계약상의 직접적 또는 간접적 책임을 지지 않습니다. 또한 사용설명서 상의 내용이 법적 또는 상업적인 특정 조건을 만족시킬 것을 보장하지 않습니다. 사용설명서는 제품의 업그레이드나 수정에 따라 예고 없이 변경될 수 있으며, 내용상의 오류가 없음을 보장하지 않습니다.

상표 공지

Tmax AnyLink®는 TmaxSoft Co., Ltd.의 등록 상표입니다. 본 사용설명서에 기재된 모든 제품과 회사 이름은 각각 해당 소유주의 상표로서 참조용으로만 사용되며 반드시 상표 표시 (™, ®)를 하지는 않습니다.

오픈소스 소프트웨어 공지

본 제품의 일부 파일 또는 모듈은 다음의 라이선스를 준수합니다. : APACHE2.0, CDDL1.0, EDL1.0, OPEN SYMPHONY SOFTWARE1.1, TRILEAD-SSH2, Bouncy Castle, BSD, MIT, SIL OPEN FONT1.1

관련 상세 정보는 제품의 다음 디렉터리에 기재된 사항을 참고하시기 바랍니다. : \${AnyLink_HOME}\AnyLink-licenses

유지 보수

구분	지원항목	서비스 내용
제품지원	패치 & 업그레이드	무상 패치 서비스 제공 메이저 버전 업그레이드 시 할인 혜택 웹 지원을 통한 패치 내역 제공
기술 지원 - 기본 서비스	장애 지원	장애 발생 시 원인 분석 및 조치 Service Desk팀 → 기술팀 → R&D의 3단계 장애 분석 및 조치
	일상 지원(온라인 지원)	E-mail, 전화, 원격, 웹 사이트 등 온라인 자원을 통한 질의 응답 서비스
	고객 맞춤 지원(방문 지원)	고객의 요청으로 수행하는 방문 지원 서비스
기술 지원 - 옵션 서비스	예방 지원	정기 점검을 통한 시스템 운영현황 보고 및 장애 예방 <ul style="list-style-type: none"> 관리자 또는 운영자의 요구사항 수렴 운영 현황(시스템, 엔진 운영) 보고서 제공 필요 시 시스템 개선 권장 사항 보고
유지 보수 비용 및 기간	계약 시 별도 협의	계약 시 EOL/EOS 문서 제공

안내서 이력

제품 버전	안내서 버전	발행일	비고
AnyLink 7	3.1.2	2025-01-20	-
AnyLink 7	3.1.1	2023-03-13	-
AnyLink 7	2.1.4	2019-08-23	-
AnyLink 7	2.1.1	2017-03-24	-

목차

1. 소개	1
1.1. 개요	1
1.2. 환경 구성	1
1.3. 어댑터 동작방식	1
2. 환경설정	2
2.1. 어댑터 설정	2
2.2. 엔드포인트 설정	4
2.2.1. 인바운드 엔드포인트	6
2.2.2. 아웃바운드 엔드포인트	8
2.3. 아웃바운드 룰 설정	11
3. 예제	16
3.1. 개요	16
3.2. 어댑터 생성	16
3.2.1. 아웃바운드 어댑터 생성	17
3.3. 엔드포인트 생성	17
3.3.1. 인바운드 엔드포인트 생성	17
3.3.2. 아웃바운드 엔드포인트 생성	19
3.4. 스튜디오 리소스 생성	20
3.4.1. 거래그룹 생성	20
3.4.2. 거래 생성	21
3.4.3. 메시지 생성	21
3.4.4. 아웃바운드 룰 생성	22
3.4.5. 플로우 생성	23
3.5. 스튜디오 리소스 설정	24
3.5.1. 거래그룹 설정	24
3.5.2. 거래 설정	26
3.5.3. 메시지 설정	27
3.5.4. 아웃바운드 룰 설정	28
3.5.5. 플로우 설정	28
3.6. 배포	34
3.7. 거래 테스트	35

1. 소개

본 장에서는 AnyLink HTTP 어댑터의 기본적인 동작방식에 대해서 설명한다.

1.1. 개요

HTTP 어댑터는 통신 프로토콜이 HTTP 기반으로 정의되어 있을 경우 사용한다. HTTP 어댑터는 인바운드, 아웃바운드를 지원하며, 엔드포인트에 설정한 통신 방향에 따라서 처리하는 동작 방식이 달라진다.

1.2. 환경 구성

AnyLink에서 HTTP 인바운드 엔드포인트를 사용하기 위해서는 다음과 같은 환경 구성이 필요하다. 자세한 설정 방법은 JEUS 안내서를 참조한다.

- JEUS에 HTTP Listener 등록
- JEUS 서버에 Servlet 등록

1.3. 어댑터 동작방식

HTTP 어댑터는 HTTP 엔드포인트를 등록할 수 있다. 엔드포인트는 통신 방향에 따라 인바운드, 아웃바운드로 나뉜다.

- 인바운드

엔드포인트에서 설정한 path와 포트를 Listen하다가 엔드포인트로 요청이 들어오면 설정되어 있는 거래로 요청 메시지를 전달하여 플로우를 시작한다.

- 아웃바운드

플로우에서 HTTP 아웃바운드 룰을 통하여 엔드포인트로 요청을 보내면 엔드포인트에 설정되어 있는 URL로 요청 메시지를 송신한다.

2. 환경설정

본 장에서는AnyLink HTTP 어댑터의 환경설정 방법과 아웃바운드 룰을 설정하는 방법에 대해서 설명한다.

2.1. 어댑터 설정

HTTP 어댑터는 WebAdmin을 이용하여 설정한다.



AnyLink WebAdmin의 접속 및 사용법에 관한 자세한 내용은 "AnyLink WebAdmin 안내서"를 참고한다.

웹 브라우저를 이용해 AnyLink WebAdmin에 로그인한 후 초기 화면에서 **[구성관리] > [어댑터]**를 선택하면 다음과 같이 어댑터 목록이 나타난다.

어댑터 목록 화면

어댑터 목록 화면의 **[Add]** 버튼을 클릭하면 어댑터를 등록할 수 있다. **어댑터 등록 화면**에서 생성할 HTTP 어댑터에 대한 정보를 입력하고 **[저장]** 버튼을 클릭한다. 정상적으로 등록되었을 경우 “저장되었습니다.”라는 결과 메시지가 나타난다.

어댑터 등록

어댑터 목록

기본정보

어댑터 아이디*

HTTP_ADT

어댑터 이름*

HTTP_ADT

어댑터 종류*

HTTP

설명

HTTP 어댑터

업무시스템 이름*

클러스터001

배포 사유

상세설정

아웃바운드 스크드 롤 아이디

미사용

저장

취소

어댑터 등록 화면

• 기본정보 (* : 필수 입력항목)

항목	설명
어댑터 아이디 *	<p>생성할 리소스 어댑터의 아이디를 입력한다.</p> <p>3~30자리 영문자 또는 숫자 조합, 특수문자(-, _) 입력이 가능하다. 해당 값은 동일 업무시스템에서 유일한 값이다.</p>
어댑터 이름 *	<p>생성할 리소스 어댑터의 이름을 입력한다.</p> <p>3~30자리 영문자, 한글 또는 숫자 조합, 특수문자(-, _) 입력이 가능하다.</p>

항목	설명
어댑터 종류 *	<p>어댑터의 종류를 선택한다. 리스트에서 HTTP를 선택한다.</p> <ul style="list-style-type: none"> • TCP • HTTP • Tmax • WebService • DB • FILE • FTP • MQ • SAP • UDP • SMTP • WEBDAV • Tuxedo • JMS • ebXML • ISO8583 • ProObject
설명	생성하는 어댑터에 대한 설명을 입력한다. 입력하지 않아도 무관하다.
업무시스템 이름 *	어댑터가 포함될 업무시스템을 리스트에서 선택한다.
배포 사유	배포 사유를 작성한다. 작성한 사유는 [배포관리] > [배포이력] 메뉴에서 조회하면 확인할 수 있다.

• **[상세설정] 탭**

항목	설명
아웃바운드 스레드 풀 아이디	<p>아웃바운드 시에 사용될 스레드 풀 아이디를 설정한다. 엔진에서 수행된 스레드 풀과 다른 아이디가 지정될 경우 아웃바운드 시에 해당 스레드 풀로 스레드를 변경한다.</p> <p>어댑터별로 스레드 풀을 분리하여 서로 다른 어댑터 간에 영향이 없게 하려는 경우에 설정한다.</p>

2.2. 엔드포인트 설정

WebAdmin의 **[구성관리] > [어댑터]** 메뉴를 선택하면 왼쪽에 등록된 어댑터 트리가 조회된다. 트리에서

엔드포인트를 추가할 어댑터를 클릭하면 오른쪽에 어댑터 상세정보를 조회할 수 있다.

어댑터 상세정보 화면의 [엔드포인트 목록] 탭을 클릭하면 해당 어댑터 하위에 있는 엔드포인트 목록이 조회된다.

어댑터 상세정보

기본정보

어댑터 아이디*

HTTP_ADT_01

어댑터 이름*

HTTP_ADT_01

어댑터 종류*

HTTP

업무시스템 이름*

ANL001

설명

배포 사유

상세설정

엔드포인트 그룹 목록

엔드포인트 목록

	No	엔드포인트 아이디	엔드포인트 이름	통신 방향	버전	상태	최종 변경일자	담당자
<input type="checkbox"/>	1	HTTP_IN_EP_01	HTTP_IN_EP_01	Inbound	2		2018-08-17 15:47:35	admin(admin)
<input type="checkbox"/>	2	HTTP_OUT_EP_01	HTTP_OUT_EP_01	Inbound	1		2018-08-17 15:47:19	admin(admin)

+ Add

- Del

▶ Start

■ Stop

어댑터 상세정보 - [엔드포인트 목록]

[엔드포인트 목록] 탭 아래 [Add] 버튼을 클릭하면 엔드포인트 등록 화면이 나타난다. 각 항목의 정보를 입력하고 [저장] 버튼을 클릭하여 엔드포인트를 등록한다. 엔드포인트가 등록되면 WebAdmin의 왼쪽 트리에서 등록된 어댑터와 엔드포인트를 확인할 수 있다.

엔드포인트 등록 화면은 기본 정보와 [연결정보], [상세설정], [메시지 변경 옵션] 탭으로 구성된다. 기본 정보의 '통신 방향' 항목에 선택한 값에 따라서 탭 화면이 다르게 구성된다. 각 통신 방향에 대한 탭 화면에 대한 설명은 해당 절을 참고한다.

HTTP 엔드포인트 등록

기본정보

업무시스템 이름

bizSystem_test

어댑터 종류

HTTP

어댑터 이름

HTTP_ADT

엔드포인트 아이디*

HTTP_EP_IN

엔드포인트 이름*

HTTP_EP_IN

엔드포인트 그룹 이름

엔드포인트 상태

Running

부팅 초기값

Running

통신 방향

Inbound

설명

HTTP 인바운드 엔드포인트

배포 사유

연결정보

상세설정

메시지 변경 옵션

연결방식

Path*

/http/txGrp_HTTP_01

기본 인코딩

UTF-8

Inbound 포트 지정

12345

POST URLDecode

☐

응답 URL Encode

☐

SOAP 사용

☐

멀티파트 바이트 사용

☐

oneway 응답 코드

응답 Content-Length 사용

☐

인증

엔드포인트 등록 화면

• 기본정보

다음은 엔드포인트 등록 화면의 **기본정보** 영역에 대한 설명이다. (* : 필수 입력항목)

항목	설명
엔드포인트 아이디 *	엔드포인트 아이디를 입력한다. 3~30자리 영문자 또는 숫자 조합, 특수문자(-, _) 입력이 가능하다. 해당 값은 동일 어댑터 또는 엔드포인트 그룹에서 유일한 값이다.
엔드포인트 이름 *	엔드포인트 이름을 입력한다. 3~30자리 영문자, 한글 또는 숫자 조합, 특수문자(-, _) 입력이 가능하다.
엔드포인트 그룹 이름	엔드포인트의 그룹 이름이다.
엔드포인트 상태	생성할 때 상태 초기값을 선택한다. <ul style="list-style-type: none"> • Running : Running 상태이면 엔드포인트가 시작된 상태가 된다. • Stopped : Stopped 상태이면 엔드포인트가 중지된 상태가 된다.
부팅 초기값	부팅할 때 상태 초기값을 선택한다. <ul style="list-style-type: none"> • Running : 부팅할 때 엔드포인트가 시작(Running)된 상태가 된다. • Stopped : 부팅할 때 엔드포인트가 중지(Stopped)된 상태가 된다.
통신 방향	엔드포인트의 통신 방향을 설정한다. <ul style="list-style-type: none"> • Inbound : 외부로부터 메시지를 받기위한 엔드포인트이다. • Outbound : 외부로 메시지를 전달하기 위한 엔드포인트이다.
설명	생성하는 어댑터에 대한 설명을 입력한다. 입력하지 않아도 무관하다.
배포 사유	배포 사유를 작성한다. 작성한 사유는 [배포관리] > [배포이력] 메뉴에서 조회하면 확인할 수 있다.

2.2.1. 인바운드 엔드포인트

HTTP 인바운드 엔드포인트는 외부에서 AnyLink로 들어오는 HTTP 전문을 처리하는 엔드포인트이다.

다음은 [엔드포인트 등록 화면](#)의 **기본정보** 영역에 '통신 방향' 항목을 'Inbound'로 선택한 경우 각 탭 화면에 대한 설명이다.

• [연결정보] 탭 (* : 필수 입력항목)

연결방식

Path*
기본 인코딩

Inbound 포트 지정

POST URLDecode
☐
응답 URL Encode
☐

SOAP 사용
☐
멀티파트 바이트 사용
☐

oneway 응답 코드
응답 Content-Length 사용
☐

인증
 ⓘ

엔드포인트 등록 화면 - [연결정보] - 인바운드

항목	설명
Path *	엔드포인트 호출에 사용할 Path를 설정한다. Path는 Servlet의 path와 AnyLink의 거래 path로 이루어져 있다. 위의 예제에서는 /http가 Servlet-path이고, /txGrp_HTTP_01이 AnyLink의 거래path이다(Servlet의 default contextpath는 /http로 배포된다).
기본 인코딩	엔드포인트에서 사용할 기본 인코딩을 설정한다. 메시지에 설정된 인코딩이 우선 적용되고 메시지에 설정되지 않았을 경우에 적용된다. 메시지와 엔드포인트 모두 설정이 되지 않은 경우에는 시스템 기본 인코딩으로 적용된다.
Inbound 포트 지정	엔드포인트의 포트를 지정한다. 지정하지 않을 경우 JEUS에 등록되어 있는 모든 포트를 대상으로 한다.
POST URLDecode	POST 요청 전문에 대해 URLDecode의 동작 여부를 설정한다(GET에서는 항상 URLDecode를 수행한다).
응답 URLEncode	POST 응답 전문에 대해 URLEncode의 동작 여부를 설정한다(GET에서는 항상 URLEncode를 수행한다).
SOAP 사용	SOAP 메시지 형식을 만들어 통신한다.
멀티파트 바이트 사용	인바운드시 사용할 수 있다. 수신한 멀티파트 요청을 임시파일에 저장하지 않고 바이트로 저장한다.
응답 Content-Length 사용 [2023.02 stable 버전 기능추가]	HTTP 응답 헤더에 Content-Length 값을 body 데이터 길이로 동적 변경한다. (기본값: 미사용) <ul style="list-style-type: none"> • 옵션 미사용: 응답 헤더 'Transfer-Encoding: chunked'로 응답 전송 • 옵션 사용: 응답 헤더 'Content-Length: body 길이' 로 응답 전송

• [상세설정] 탭

연결정보

상세설정

메시지 변경 옵션

메시지 핸들러

메시지 핸들러

메시지 파라미터 명

멀티파트 파라미터 명

거래그룹/거래

거래그룹/거래 선택

엔드포인트 등록 화면 - [상세설정] - 인바운드

항목	설명
메시지 핸들러	요청 메시지에 대한 사용자 메시지 핸들러를 설정한다.
메시지 파라미터 명	GET 파라미터를 파싱하여 해당 파라미터를 key로 사용하고 있는 value 값을 전문으로 사용한다. GET 방식만 지원한다.
멀티파트 파라미터 명	멀티파트 메시지를 수신했을 때 요청 메시지의 파트 이름을 설정한다.

항목	설명
거래그룹/거래	엔드포인트로 들어온 요청이 수행할 거래그룹/거래를 선택한다.

• [메시지 변경 옵션] 탭

메시지를 변환하여 수신한다. 메시지 변경 옵션은 '메시지 핸들러' 전에 적용된다.

연결정보

상세설정

메시지 변경 옵션

메시지 수신시

지환 이전 문자

지환 이후 문자

엔드포인트 등록 화면 - [메시지 변경 옵션] - 인바운드

항목	설명
지환 이전 문자	메시지에서 변환할 문자를 설정한다. 개행문자(\n)나 탭(\r), 특수문자(\u) 또한 설정할 수 있다.
지환 이후 문자	지환 이전 문자를 변환할 문자를 설정한다.

2.2.2. 아웃바운드 엔드포인트

HTTP 아웃바운드 엔드포인트는 AnyLink에서 외부로 HTTP 전문을 전송할 수 있는 엔드포인트이다.

다음은 [엔드포인트 등록 화면](#)의 기본정보 영역에 '통신 방향' 항목을 'Outbound'로 선택한 경우 각 탭 화면에 대한 설명이다.

• [연결정보] 탭 (* : 필수 입력항목)

연결정보

상세설정

메시지 변경 옵션

SSL

연결방식

Split 사용 유무

☐

URL*

http://192.168.14.102:9998/HTTP_EP_IN_1

기본 인코딩

연결 타임아웃*

3000

ms

응답 대기 타임아웃

0

ms

소켓 타임아웃*

30000

ms

최대 연결 수

0

keepAlive 사용

☐

SOAP 사용

☐

Agent 사용여부

☐ 사용
☒ 미사용

메시지 헤더

+ 추가

No	Name	Value	변수 값 암호화 사용 유무	삭제

엔드포인트 등록 화면 - [연결정보] - 아웃바운드

항목	설명
Split 사용 유무	항목을 선택하면 URL을 프로토콜, IP/호스트, Path, 포트의 구성요소별로 나누어 작성할 수 있다.
URL *	아웃바운드 엔드포인트를 통해 연결할 목적지의 URL을 입력한다.
기본 인코딩	엔드포인트에서 사용할 기본 인코딩을 설정한다. 메시지에 설정된 인코딩이 우선 적용되고 메시지에 설정되지 않았을 경우에 적용된다. 메시지와 엔드포인트 모두 설정이 되지 않은 경우에는 시스템 기본 인코딩으로 적용된다.
연결 타임아웃 *	연결 타임아웃(Connect timeout)을 설정한다. 연결을 요청해서 연결될 때까지의 타임아웃이다. (기본값: 3000)
응답대기 타임아웃	응답대기 타임아웃을 설정한다. 엔드포인트 상태가 stop이 되기 전에 응답대기를 위한 타임아웃이다.
소켓 타임아웃 *	소켓 타임아웃을 설정한다. 소켓으로부터 데이터를 읽어 올 때 발생하는 타임아웃이다. (기본값: 30000)
최대 연결 수	최대 연결 수를 설정한다. 외부와 엔드포인트간의 최대 연결수이다.
KeepAlive 사용	HTTP 헤더의 커넥션을 Keep-Alive로 사용할 때 설정한다.
SOAP 사용	SOAP 메시지 형식을 만들어 통신한다.
Agent 사용여부	엔드포인트를 Agent로 사용할 경우 설정한다. '사용'을 선택하는 경우 Agent 아이디를 설정한다. Agent 엔드포인트로 사용하는 경우 아래 목록의 라이브러리를 서버의 \${Agent 설치 폴더}/lib/common 경로에 추가해야 한다. <ul style="list-style-type: none"> • httpclient-4.3.2.jar • httpcore-4.3.2.jar • httpasyncclient-4.0.1.jar • commons-logging-1.1.3.jar • httpcore-nio-4.3.2.jar
메시지 헤더	HTTP 헤더에 설정 값을 key - value 페어로 입력한다. 변수 값 암호화 사용 유무를 체크할 경우 업무시스템에 설정된 암호화 알고리즘으로 암호화한다.

• [상세설정] 탭

연결정보
상세설정
메시지 변경 옵션
SSL

메시지 핸들러

🔍

핸들러

메시지 파라미터 명

엔드포인트 등록 화면 - [상세설정] - 아웃바운드

항목	설명
메시지 핸들러	응답 메시지에 대한 사용자 메시지 핸들러를 설정한다.
메시지 파라미터 명	해당 파라미터를 key로 하여 value에 전문을 설정한다. GET 방식만 지원한다.

• [메시지 변경 옵션] 탭

메시지를 변환하여 송신한다. 메시지 변경 옵션은 '메시지 핸들러' 전에 적용된다. 화면에 대한 자세한 설명은 [인바운드 엔드포인트](#)를 참고한다.

• [SSL] 탭

다음은 아웃바운드 엔드포인트 등록 화면의 [SSL] 탭 화면이다.

엔드포인트 등록 화면 - [SSL] - 아웃바운드

◦ SSL

항목	설명
SSL 사용 설정	SSL 인증의 사용 여부를 선택한다. 'SSL인증서 없이 사용'은 SSL은 사용하지만 인증서 없이 통신할 때 설정한다.

◦ Trust Store 설정

SSL 관련 Trust Store정보를 설정한다. 해당 영역은 SSL 인증에 사용될 Trust Store(공개키, 인증서) 파일 설정과 관련된다.

항목	설명
스토어 위치	해당 Trust Store의 저장 위치를 설정한다.
스토어 타입	해당 Trust Store의 저장 형식을 설정한다. (기본값: jks)
클라이언트 인증	클라이언트 인증 여부를 선택한다. '예'로 설정하면 Key Store를 설정할 수 있다.
스토어 패스워드	해당 Trust Store를 접근하기 위한 암호를 설정한다.

◦ Key Store 설정

항목	설명
스토어 위치	Key store의 저장 위치를 설정한다.

항목	설명
스토어 타입	Key store의 저장 형식을 설정한다. (기본값: jks)
Private Key 패스워드	개인 키를 접근하기 위한 암호를 설정한다.
스토어 패스워드	Key store를 접근하기 위한 암호를 설정한다.

2.3. 아웃바운드 룰 설정

본 절에서는 아웃바운드 룰 설정방법에 대해서 설명한다. HTTP 아웃바운드 룰은 AnyLink 스튜디오를 이용하여 설정한다.



AnyLink 스튜디오의 자세한 사용법은 "AnyLink 스튜디오 안내서"를 참고한다.

AnyLink 스튜디오 거래그룹 네비게이터의 **거래/거래그룹** 항목의 컨텍스트 메뉴에서 **[새로만들기] > [아웃바운드 룰]** > **[HTTP 아웃바운드 룰]**를 선택하면 다음과 같이 **HTTP 아웃바운드 룰 생성 화면**이 나타난다. 각 항목을 입력하고 **[Finish]** 버튼을 클릭한다.

HTTP 아웃바운드 룰 생성 화면

항목	설명
아웃바운드 아이디	아웃바운드 룰의 아이디를 입력한다. 영어, 숫자, 특수문자(_) 입력이 가능하며 첫 글자는 영어만 가능하다. 해당 값은 동일 거래 노드에 유일한 값이다.
아웃바운드 이름	아웃바운드 룰의 이름을 입력한다. 한글, 영어, 숫자, 특수문자(-, _) 입력이 가능하다. 아웃바운드 룰 이름은 XML Naming Conversion을 따른다.

HTTP 아웃바운드 룰을 생성하면 다음과 같은 설정 화면이 나타난다.

The screenshot shows the 'HTTP 아웃바운드 룰' (HTTP Outbound Rule) configuration page. It includes fields for Protocol (HTTP), Outbound Rule ID (HTTP_Outbound), Outbound Rule Name (HTTP_Outbound), and a timeout value (10,000 ms). There are also checkboxes for 'Endpoint * (Group)' and '설정' (Settings). The '요청 메시지' (Request Message) section has a table with columns: 이름 (Name), 메시지 ID (Message ID), 타입 ID (Type ID), and 배열 선택 (Array Selection). The '응답 헤더 메시지' (Response Header Message) section has a table with columns: 이름 (Name), 메시지 ID (Message ID), 타입 ID (Type ID), and 배열 선택 (Array Selection). There are also checkboxes for '응답 헤더 메시지 사용' (Use Response Header Message) and '응답 종류 확인 필드' (Response Type Check Field).

HTTP 아웃바운드 룰 정의 화면

• HTTP 아웃바운드 룰 정의 (* : 필수 입력항목)

항목	설명
아웃바운드 룰 ID *	아웃바운드 룰 아이디를 입력한다. 영어와 숫자, 특수문자(_) 입력이 가능하다. 해당 값은 동일 거래 노드에 유일한 값이다.
아웃바운드 룰 이름 *	아웃바운드 룰 이름을 입력한다. 한글, 영어, 숫자, 특수문자(-, _) 입력이 가능하다.

항목	설명
요청처리 타임아웃(ms)	요청처리 타임아웃을 입력한다. 요청처리 타임아웃은 아웃바운드 룰이 호출된 시점에서부터 타임아웃 시간 동안 연결을 할당받지 못하거나 외부로 요청을 보냈지만 응답을 받지 못했을 경우 발생한다.
Endpoint(Group) *	아웃바운드 룰이 등록될 엔드포인트 또는 엔드포인트 그룹을 선택한다. 해당 프로젝트가 업무시스템에 할당되고 서버에 접속한 상태여야만 선택이 가능하다.
설명	아웃바운드 룰의 설명을 입력한다. 설명은 입력하지 않아도 무방하다.
응답 수신 방식	메시지에 대한 응답 수신 방식을 선택한다. <ul style="list-style-type: none"> • SYNC : 외부로 요청을 보냈던 엔드포인트로 응답이 들어온다. • ONEWAY : 응답이 없는 ONEWAY이다.
HTTP Method	HTTP의 요청방법을 설정한다. <ul style="list-style-type: none"> • GET : URL에 해당하는 자료의 전송을 요청한다. • POST : 서버가 처리할 수 있는 자료를 전송한다. • PUT : 해당 URL에 자료를 저장한다. • DELETE : 해당 URL의 자료를 삭제한다. • INBOUND : 거래가 Inbound 요청일 경우에만 사용 가능하며, Inbound 요청으로 수신한 Method로 HTTP 요청방법을 설정한다.
URL 인코딩	엔드포인트에 설정된 URL로 요청을 보낼 때 URL 인코딩을 할 것인지 여부를 설정한다. POST 방식일 때만 설정이 가능하다.
URL 디코딩	엔드포인트에 설정된 URL에서 응답이 올 때 URL 디코딩을 할 것인지 여부를 설정한다. POST 방식일 때만 설정이 가능하다.
Content Type	HTTP header에 명시될 Content-Type을 설정한다. 일반적으로는 메시지의 Type에 맞는 Content-Type을 설정하거나 설정하지 않아도 무방하다. 설정하지 않은 경우 text/plain을 default로 설정한다. Key-Value 형식 메시지를 사용할 경우 "application/x-www-form-urlencoded" Content-Type을 사용하면 수신쪽 서버릿에서 getParameter(Key)를 통해 Value를 가져 올 수 있다.
XML Tag 이전 데이터 Trim	XML Tag의 앞에 공백문자가 들어가면 파싱오류가 일어나게 된다. 이 옵션을 설정하면 오류를 막기 위하여 공백문자를 제거한다.
XML Self-Close Flag	빈 요소의 XML TAG를 설정한다. <ul style="list-style-type: none"> • FALSE : '<ELEM></ELEM>' 형태로 설정된다. • TRUE : '<ELEM/>' 형태로 설정된다.
ByPass	ByPass 거래에서 해당 아웃바운드 룰을 호출할 수 있도록 설정한다. 체크할 경우 요청 및 응답 메시지가 비활성화된다.

항목	설명
Resource path parameter	ByPass 거래에서 요청 헤더에 필드 값이 존재하면 해당 값을 URL에 추가한다. 예를 들어 /http/\${ResourcePath-Param}/\${test}로 설정 시 헤더에 {ResourcePath-Param}의 값이 aaaa이고, {test} 값이 bbbb라면 url + /http/aaaa/bbbb의 값으로 처리된다.
DynamicInetAddress 사용	요청 메시지의 필드 값으로 송신할 Host와 Port를 설정한다. [검색] 버튼을 클릭하면 메시지 ID와 필드 ID를 선택할 수 있다.

• HTTP 아웃바운드 룰 메시지 정의

HTTP 아웃바운드 룰에서는 요청 메시지, 정상 응답 메시지를 설정할 수 있다.

항목	설명
요청 메시지	아웃바운드 룰에서 사용될 요청 메시지를 선택한다. 요청 메시지는 멀티 메시지 설정이 가능하다. 멀티 메시지는 XML 전문을 사용하는 경우 사용되는 기능으로, 각각의 전문에 네임스페이스 URI와 Local Part를 별도로 설정하고자 하는 경우에 사용한다. '멀티 메시지 사용' 을 선택한 경우 HTTP Multi Part 정보를 설정할 수 있다(HTTP 아웃바운드 룰 멀티파트 화면 참고).
정상 응답 메시지	아웃바운드 룰에서 사용될 응답 메시지를 선택한다.
응답 헤더 메시지	응답 헤더 메시지를 선택한다. 응답 헤더 메시지는 응답 메시지가 정상과 업무 오류로 나누어져 있을 경우 사용된다(일반적인 에러가 아닌 정상 메시지 형태로 업무 오류 메시지를 수신할 경우 사용). <ul style="list-style-type: none"> 응답 종류 확인 : 응답 헤더 메시지를 사용할 경우에 정상 응답 메시지와 업무 오류 응답 메시지를 구분하기 위한 메시지 필드와 해당 값에 따른 정상 또는 업무 오류를 입력한다.

• HTTP Multi Part 설정

HTTP 아웃바운드 룰에서 멀티파트를 설정할 수 있다.

HTTP Multi Part 설정

• 멀티파트 사용

☒

• SOAP 배치 사용

☐

• 메시지 파트 이름

• 매핑 유형

파일 매핑

• 임시 파일 경로

Mapping...

• 파트 이름

Mapping...

HTTP 아웃바운드 룰 멀티파트 화면

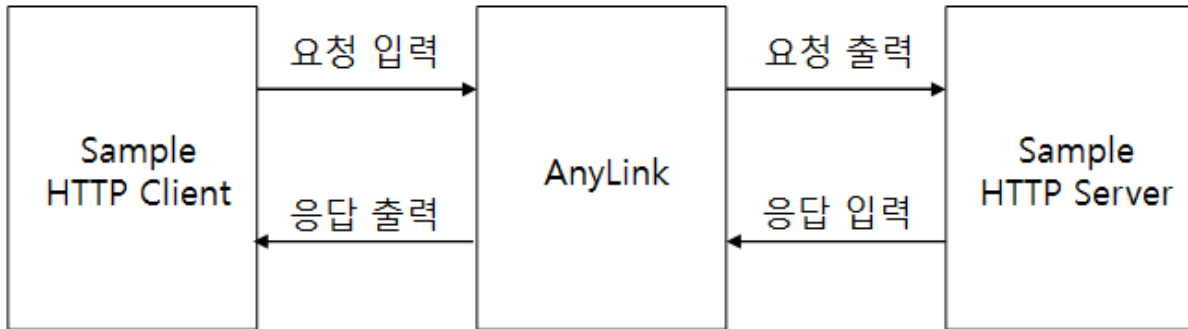
항목	설명
멀티파트 사용	메시지를 송신의 경우 멀티파트를 사용할지에 대해 설정한다.
SOAP 배치 사용	SOAP 배치용 헤더로 송신한다. 설정할 경우 '메시지 파트 이름', '파일 매핑'을 선택한 경우 '파트 이름', '바이트 매핑'을 선택한 경우 '파트 이름', '파일 이름' 항목은 설정할 수 없다.
메시지 파트 이름	요청 메시지의 첫 번째 파트 이름(name)을 설정한다.
매핑 유형	다음 중에 선택한다. <ul style="list-style-type: none"> 파일 매핑 : 임시파일로 저장한다. 바이트 매핑 : 임시파일이 아닌 바이트로 저장한다.
임시 파일 경로	'매핑 유형' 항목을 '파일 매핑'으로 선택한 경우 송신할 파일의 경로를 설정한다.
파트 이름	송신할 파일에 대한 파트 이름을 설정한다.
파일 내용	'매핑 유형' 항목을 '바이트 매핑'으로 선택한 경우 바이트로 저장한 파일 내용을 설정한다.
파일 이름	'매핑 유형' 항목을 '바이트 매핑'으로 선택한 경우 바이트로 저장한 파일의 이름을 설정한다.

3. 예제

본 장에서는 AnyLink HTTP 어댑터를 사용한 예제를 설명한다.

3.1. 개요

다음은 HTTP 기본 거래 구성이다.



기본 거래 구성도

다음은 거래 순서를 수행하는 과정에 대한 설명이다.

1. Sample HTTP Client를 통해 요청 입력 전문을 전송한다.
2. 인바운드 어댑터로 전문 수신 후 서비스 플로우 수행한다.
3. 서비스 플로우에서 요청 입력 전문을 요청 출력 전문으로 매핑한다.
4. 아웃바운드 어댑터를 통해 요청 출력 전문을 Sample HTTP Server로 전송한다.
5. Sample HTTP Server에서 응답 입력 전문을 전송한다.
6. 서비스 플로우에서 응답 입력 전문을 응답 출력 전문으로 매핑한다.
7. 인바운드 어댑터로 응답 출력 전문을 Sample HTTP Client로 전송한다.

사용된 전문은 모두 JSON 형식의 메시지를 사용한다.

3.2. 어댑터 생성

AnyLink WebAdmin에 로그인한 후 초기 화면에서 **[구성관리] > [어댑터]**를 선택한 후 **어댑터 목록 화면의 [Add]** 버튼을 클릭하면 **어댑터 등록 화면**이 나타난다. 어댑터 생성에 대한 자세한 내용은 [어댑터 설정](#)을 참고한다.



AnyLink WebAdmin의 접속 및 사용법에 관한 자세한 내용은 "AnyLink WebAdmin 안내서"를 참고한다.

3.2.1. 아웃바운드 어댑터 생성

다음은 HTTP 어댑터를 생성하는 예제이다.

어댑터 등록

어댑터 목록

기본정보

어댑터 아이디*

HTTP_ADT_Tutorial

어댑터 이름*

HTTP_ADT_Tutorial

어댑터 종류*

HTTP

설명

HTTP 아웃바운드 어댑터

업무시스템 이름*

ANL001

배포 사유

상세설정

아웃바운드 스레드 풀 아이디

미사용

어댑터 등록 화면

• 기본정보

항목	설정값
어댑터 아이디	HTTP_ADT_Tutorial
어댑터 이름	HTTP_ADT_Tutorial
어댑터 종류	HTTP
설명	HTTP 아웃바운드 어댑터
업무시스템 이름	ANL001

• [상세설정] 탭

항목	설정값
아웃바운드 스레드 풀 아이디	미사용

3.3. 엔드포인트 생성

어댑터 생성이 완료되면 **[어댑터]** 탭에서 등록된 어댑터를 조회하고, 조회된 어댑터를 클릭하면 **어댑터 상세정보 화면**으로 이동한다.

[엔드포인트 목록] 탭 아래 **[Add]** 버튼을 클릭하면 **엔드포인트 등록 화면**이 나타난다. 엔드포인트 생성에 대한 자세한 설명은 [엔드포인트 설정](#)을 참고한다.

3.3.1. 인바운드 엔드포인트 생성

다음은 HTTP 인바운드 엔드포인트를 등록하는 예제이다.

• 기본정보

HTTP 엔드포인트 등록 엔드포인트 목록

기본정보

업무시스템 이름

ANL001

어댑터 종류

HTTP

어댑터 이름

HTTP_ADT_Tutorial

엔드포인트 아이디*

HTTP_EP_IN

엔드포인트 이름*

HTTP_EP_IN

엔드포인트 그룹 이름

엔드포인트 상태

Running

부팅 초기값

Running

통신 방향

Inbound

설명

백포 사유

인바운드 엔드포인트 등록 화면 - 기본정보

항목	설정값
엔드포인트 아이디	HTTP_EP_IN
엔드포인트 이름	HTTP_EP_IN
엔드포인트 상태	Running
부팅 초기값	Running
통신 방향	Inbound

• [연결정보] 탭

연결정보

상세설정

메시지 변경 옵션

연결방식

Path*

/http/tc/HTTP_EP_IN_Tutorial

기본 인코딩

Inbound 포트 지정

POST URLDecode

☐

응답 URL Encode

☐

SOAP 사용

☐

멀티파트 바이트 사용

☐

인바운드 엔드포인트 등록 화면 - [연결정보]

항목	설정값
Path	/http/tc/HTTP_EP_IN_Tutorial

• [상세설정] 탭

연결정보

상세설정

메시지 변경 옵션

메시지 핸들러

🔍 핸들러

메시지 파라미터 명

멀티파트 파라미터 명

거래그룹/거래

com.tmax.tutorial.httpTutorial

🔍 거래그룹/거래 선택

인바운드 엔드포인트 등록 화면 - [상세설정]

항목	설정값
거래그룹/거래	com.tmax.tutorial.httpTutorial

(거래그룹/거래 항목은 직접 입력해도 되고 스튜디오에서 거래그룹/거래 생성 후 배포한 리소스를 **거래그룹/거래 선택 화면**에서 선택하여도 무방하다.)

3.3.2. 아웃바운드 엔드포인트 생성

다음은 HTTP 아웃바운드 엔드포인트를 등록하는 예제이다. 본 예제에서는 [연결정보] 탭만 설정한다.

• 기본정보

HTTP 엔드포인트 등록					엔드포인트 목록
기본정보					
업무시스템 이름	ANL001	어댑터 종류	HTTP	어댑터 이름	HTTP_ADT_Tutorial
엔드포인트 아이디*	HTTP_EP_OUT	엔드포인트 이름*	HTTP_EP_OUT	엔드포인트 그룹 이름	
엔드포인트 상태	Running	부팅 초기값	Running	통신 방향	Outbound
설명	HTTP 아웃바운드 어댑터				
배포 사유					

아웃바운드 엔드포인트 등록 화면 - 기본정보

항목	설정값
엔드포인트 아이디	HTTP_EP_OUT
엔드포인트 이름	HTTP_EP_OUT
엔드포인트 상태	Running
부팅 초기값	Running
통신 방향	Outbound

• [연결정보] 탭

연결정보	상세설정	메시지 변경 옵션	SSL	
연결방식				
Split 사용 유무	<input type="checkbox"/>			
URL*	http://192.168.14.102:9998/HTTP_EP_IN_1		기본 인코딩	
연결 타임아웃*	3000 ms	응답 대기 타임아웃	0 ms	
소켓 타임아웃*	30000 ms	최대 연결 수	0	
keepAlive 사용	<input type="checkbox"/>			
SOAP 사용	<input type="checkbox"/>			
Agent 사용여부	<input checked="" type="radio"/> 사용 <input type="radio"/> 미사용			
메시지 헤더				
<input type="button" value="+ 추가"/>				
No	Name	Value	변수 값 암호화 사용 유무	삭제

아웃바운드 엔드포인트 등록 화면 - [연결정보]

항목	설정값
URL	http://192.168.14.102:9998/http/HTTP_EP_IN_Tutorial
연결 타임아웃	3000
소켓 타임아웃	30000

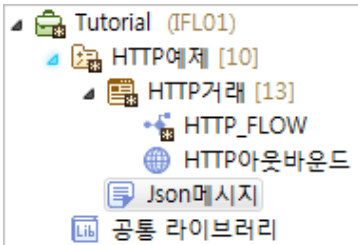
3.4. 스튜디오 리소스 생성

본 절에서는 스튜디오에서 리소스를 생성하는 방법에 대해서 설명한다.



AnyLink 스튜디오 사용법에 대한 자세한 내용은 "AnyLink 스튜디오 안내서"를 참고한다.

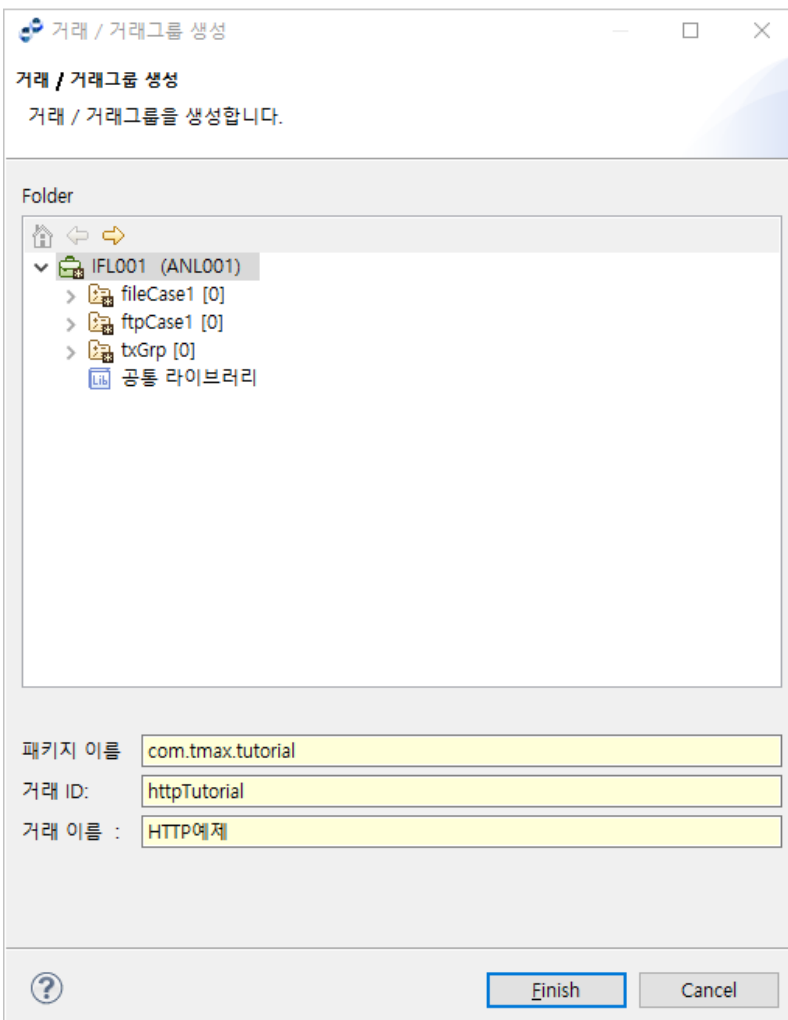
AnyLink 스튜디오를 이용하여 다음과 같이 거래 리소스를 생성한다.



HTTP 거래 리소스 리스트

3.4.1. 거래그룹 생성

거래그룹 네비게이터에서 프로젝트를 선택한 다음 컨텍스트 메뉴에서 [새로만들기] > [거래 / 거래그룹]을 선택한다.

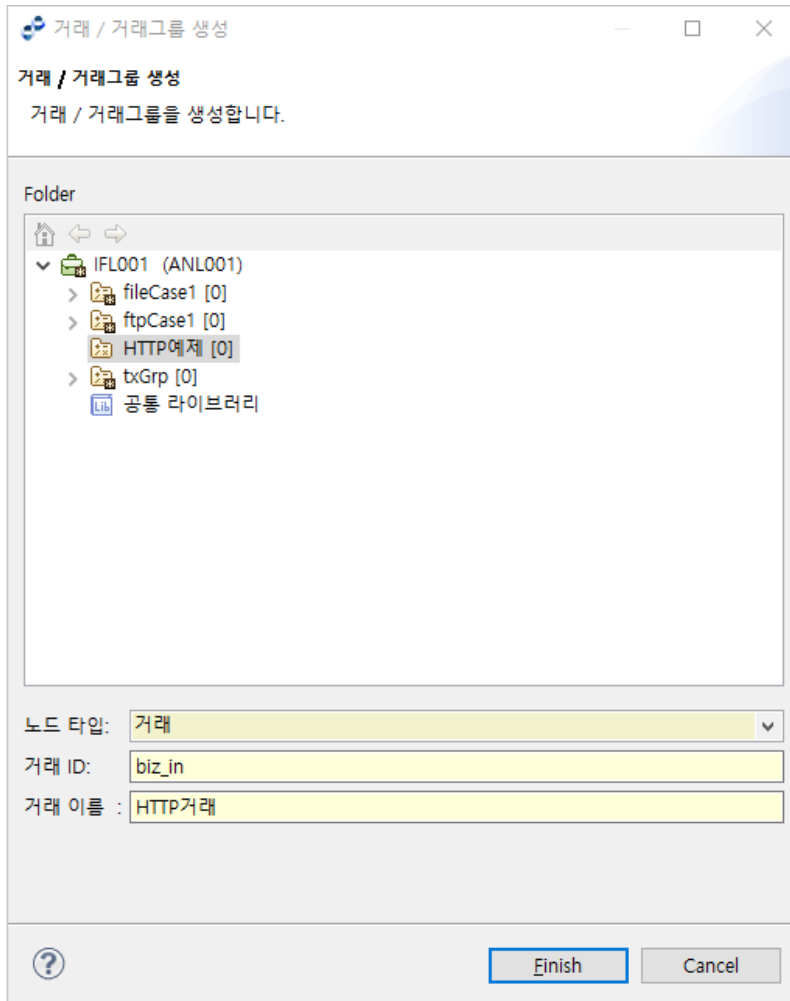


거래그룹 생성

항목	설정값
패키지 이름	com.tmax.tutorial
거래 ID	httpTutorial
거래 이름	HTTP예제

3.4.2. 거래 생성

거래그룹 네비게이터에서 **거래그룹**을 선택한 다음 컨텍스트 메뉴에서 **[새로만들기]** > **[거래 / 거래그룹]**을 선택한다.

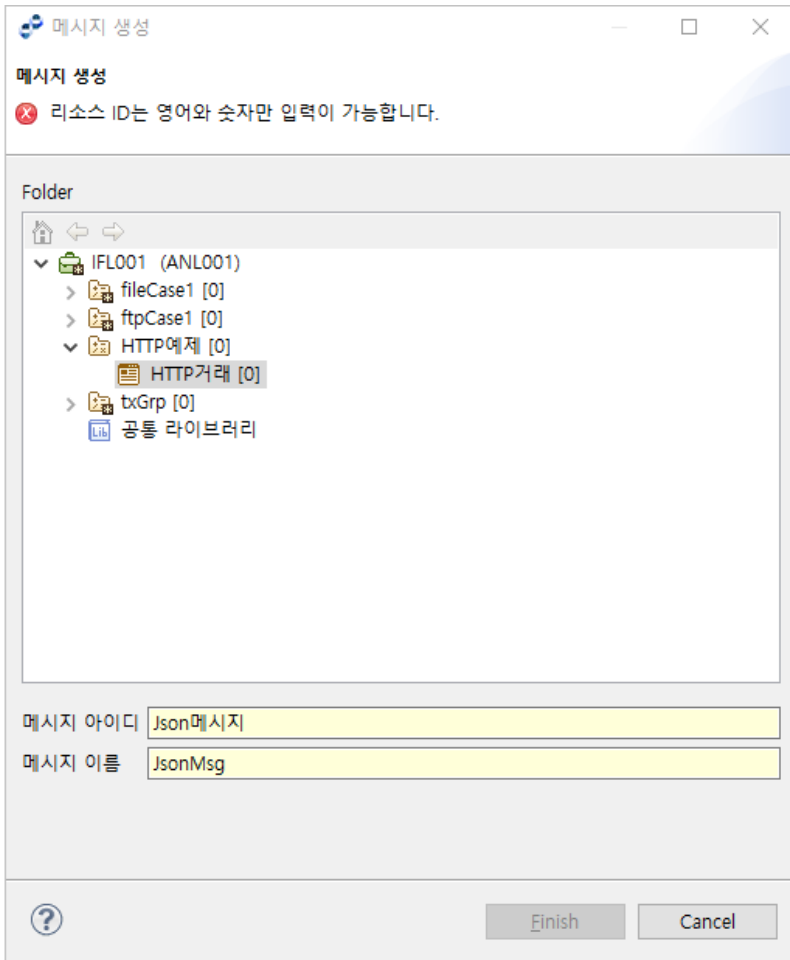


거래 생성

항목	설정값
노드 타입	거래
거래 ID	biz_in
거래 이름	HTTP거래

3.4.3. 메시지 생성

거래그룹 네비게이터에서 **거래그룹**을 선택한 다음 컨텍스트 메뉴에서 **[새로만들기]** > **[메시지]**를 선택한다.

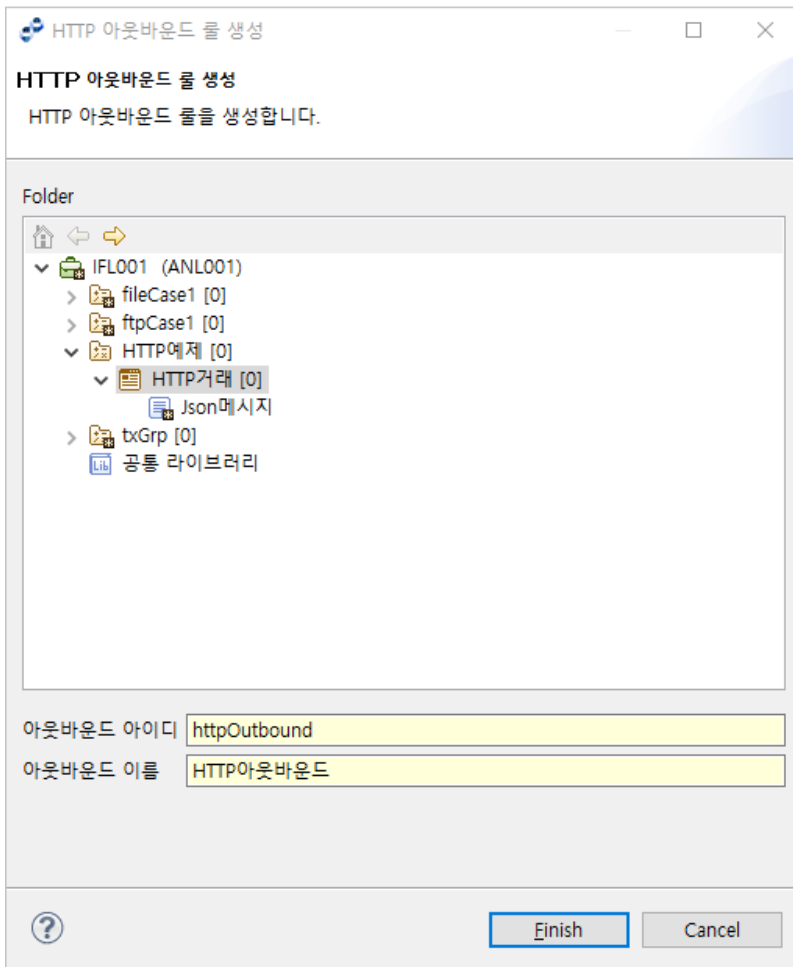


메시지 생성

항목	설정값
메시지 아이디	JsonMsg
메시지 이름	Json메시지

3.4.4. 아웃바운드 룰 생성

거래그룹 네비게이터에서 거래를 선택한 다음 컨텍스트 메뉴에서 [새로만들기] > [아웃바운드 룰] > [HTTP 아웃바운드 룰]을 선택한다.

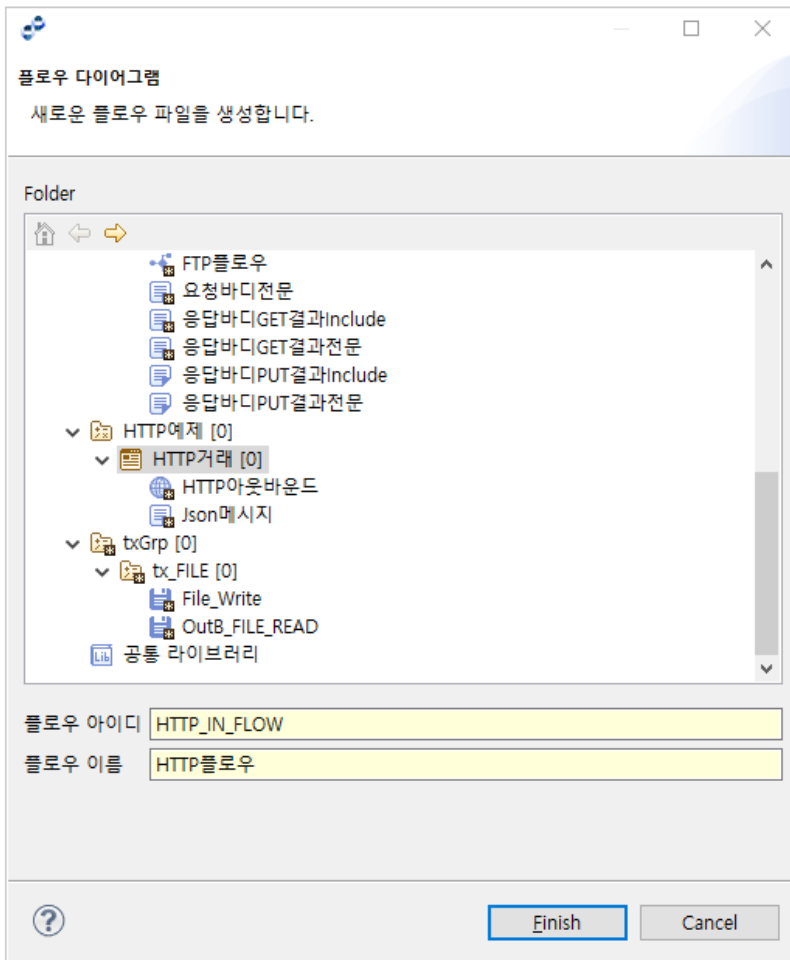


아웃바운드 룰 생성

항목	설정값
아웃바운드 아이디	httpOutbound
아웃바운드 이름	HTTP아웃바운드

3.4.5. 플로우 생성

거래그룹 네비게이터에서 거래를 선택한 다음 컨텍스트 메뉴에서 [새로만들기] > [플로우]를 선택한다.



플로우 를 생성

항목	설정값
플로우 아이디	HTTP_IN_FLOW
플로우 이름	HTTP플로우

3.5. 스튜디오 리소스 설정

본 절에서는 생성한 리소스를 설정하는 방법에 대해서 설명한다.



AnyLink 스튜디오 사용법에 대한 자세한 내용은 "AnyLink 스튜디오 안내서"를 참고한다.

3.5.1. 거래그룹 설정

다음은 거래그룹을 설정하는 예제이다. **[파싱정보]** 탭과 **[파싱 옵션]** 탭은 설정하지 않는다.

- **[거래그룹 정보]** 탭

거래그룹 정의

기본정보

- 거래그룹 ID: httpTutorial
- 패키지 이름: com.tmax.tutorial
- 거래그룹 이름: HTTP예제
- 거래 타입: NONE
- 버전: 0
- 설명:
- Bypass 설정: NO
- XA 설정: NO

요청 메시지

이름	메시지 ID	타입 ID	배열 선택

네임스페이스 URI: 로컬 파트: 그룹: NONE 그룹번호:

정상 응답 메시지

이름	메시지 ID	타입 ID	배열 선택

네임스페이스 URI: 로컬 파트: 그룹: NONE 그룹번호:

업무오류 응답 메시지

이름	메시지 ID	타입 ID	배열 선택

네임스페이스 URI: 로컬 파트: 그룹: NONE 그룹번호:

오류 응답처리

오류 응답 방식: NONE

거래그룹 설정 - [거래그룹 정보]

항목	설정값
XA 설정	NO
요청 메시지	사용 안함
정상 응답 메시지	사용 안함
Bypass 설정	NO
오류 응답 방식	NONE

• [거래그룹 옵션] 탭

거래그룹 옵션 정의

- 거래 플로우 타임아웃(ms): 60,000
- 외부 GUID 사용: NO
- GUID 타입: STRING
- 응답 유무: REQUEST_RESPONSE
- 전달 보장 설정:
 - 전달 보장:
 - 큐 ID:
 - 개별 설정:
 - 재시도 간격(s):
 - 최대 재시도 횟수:
 - 만료 시간(s):
- 중복거래 확인 여부: NO
 - 중복거래 확인 필드:

메시지 ID	필드 ID
- 중복거래 보관 주기:
- 거래 우선순위: MEDIUM

거래그룹 설정 - [거래그룹 옵션]

항목	설정값
거래 플로우 타임아웃(ms)	60000
응답 유무	REQUEST_RESPONSE

항목	설정값
거래 우선순위	MEDIUM

3.5.2. 거래 설정

다음은 거래를 설정하는 예제이다. **[파싱정보]** 탭과 **[파싱 옵션]** 탭은 설정하지 않는다.

• [거래정보] 탭

거래 설정 - [거래정보]

항목	설정값
Bypass 설정	NO
XA 설정	NO
오류 응답 방식	PARENT
서비스 타입	FLOW
서비스 이름	HTTP_IN_FLOW_Message_START_Event_BE815D(랜덤으로 생성됨)
요청 메시지	Json메시지(JsonMsg)
정상 응답 메시지	Json메시지(JsonMsg)

• [거래 옵션] 탭

거래 옵션

거래 옵션 정의

- 거래 플로우 타임아웃(ms)
- 타임아웃 설정 안함 ☒
- 외부 GUID 사용 NO
- GUID 타입 STRING
- 응답 유무 PARENT
- 전달 보장 설정 ☐
 - 전달 보장
 - 류 ID
 - 개별 설정 ☐
 - 재시도 간격(s)
 - 최대 재시도 횟수
 - 만료 시간(s)

- 중복거래 확인 여부 NO
- 중복거래 확인 필드

메시지 ID

필드 ID

추가 삭제
- 중복거래 보관 주기
- 거래 우선순위 MEDIUM

거래 설정 - [거래 옵션]

항목	설정값
거래 플로우 타임아웃(ms)	미설정
타임아웃 설정 안함	체크
응답 유무	PARENT
거래 우선순위	MEDIUM

3.5.3. 메시지 설정

AnyLink 스튜디오를 이용해서 메시지를 설정한다. 메시지 설정 화면에 대한 자세한 내용은 "AnyLink 스튜디오 안내서"를 참고한다.

메시지 에디터

메시지 정의

- 메시지 ID JsonMsg
- 메시지 이름 Json메시지

메시지 목록 *

메시지 ID	메시지 타입
JsonMsgJSON	JSON

추가

삭제

수정

JsonMsgJSON 메시지 구성 *

No.	Physical Name	Logical Name	Field Type	Included Str. Name	Included Str. Path	Array Size	Comment	Mask	Encode(JSON)	Date Pattern(yyyy-MM-dd)	Corr. Field
1	Header	헤더	string						Char	yyyy-MM-dd	false
2	Body	바디	string						Char	yyyy-MM-dd	false

추가

복수 필드 추가

삭제

검색

변환

VO Export

유효성검사

설정

- JsonMsgJSON 메시지 타입 설정

Message Encoding

NONE

Null 처리 방식

NULL

UnmarshallingException 무시

☐

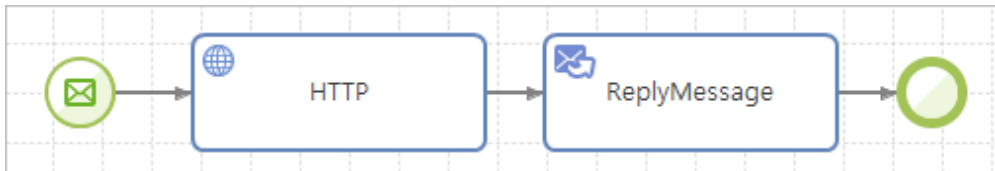
Object 이름

메시지 설정 - JSON 메시지

2. 변수 설정
3. 메시지 이벤트 설정
4. 아웃바운드 호출 설정
5. 응답 호출 설정

플로우 그리기

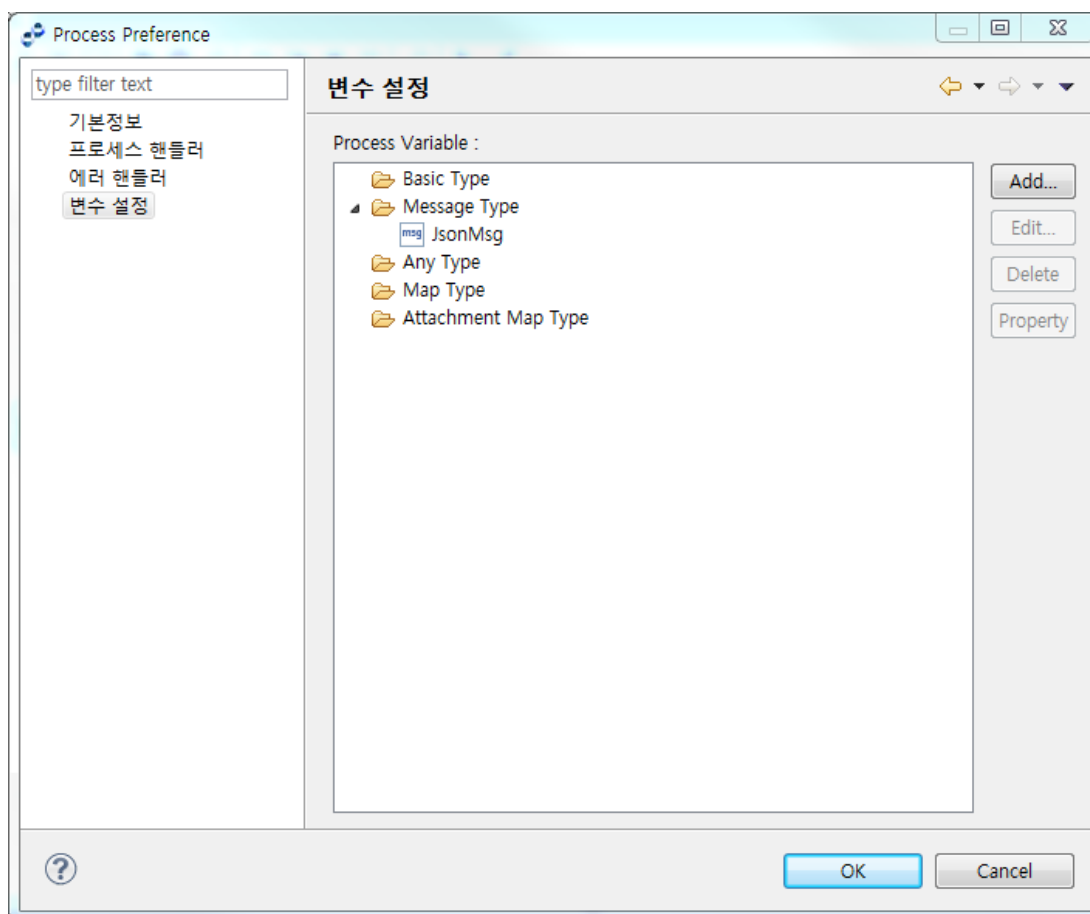
다음은 예제에 대한 서비스 플로우이다.



서비스 플로우 에디터 - 기본 그림

변수 설정

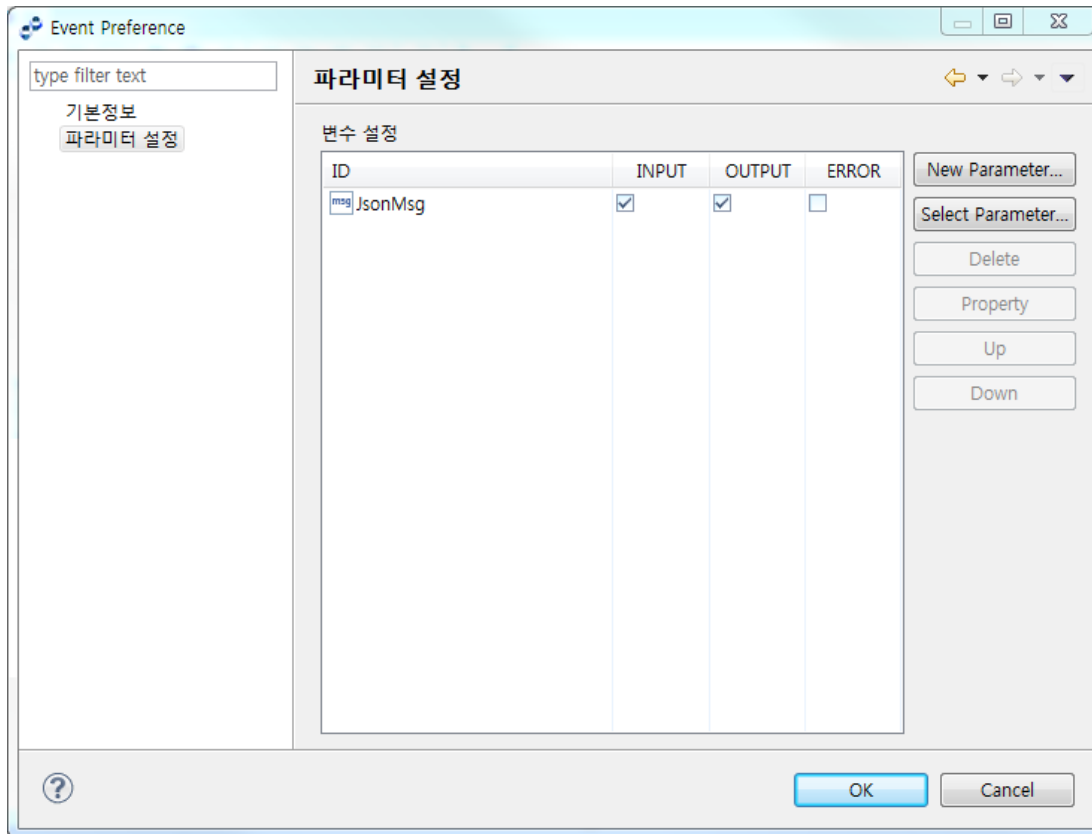
변수 설정을 하기 위해 플로우 에디터 바탕의 컨텍스트 메뉴에서 **[Property]**를 선택한다.



Property Preference - 변수 설정

메시지 이벤트 설정

플로우 에디터의 [메시지 이벤트] 컨텍스트 메뉴에서 [Property]를 선택한다. **Event Preference** 화면에서 [파라미터 설정]을 선택한다.

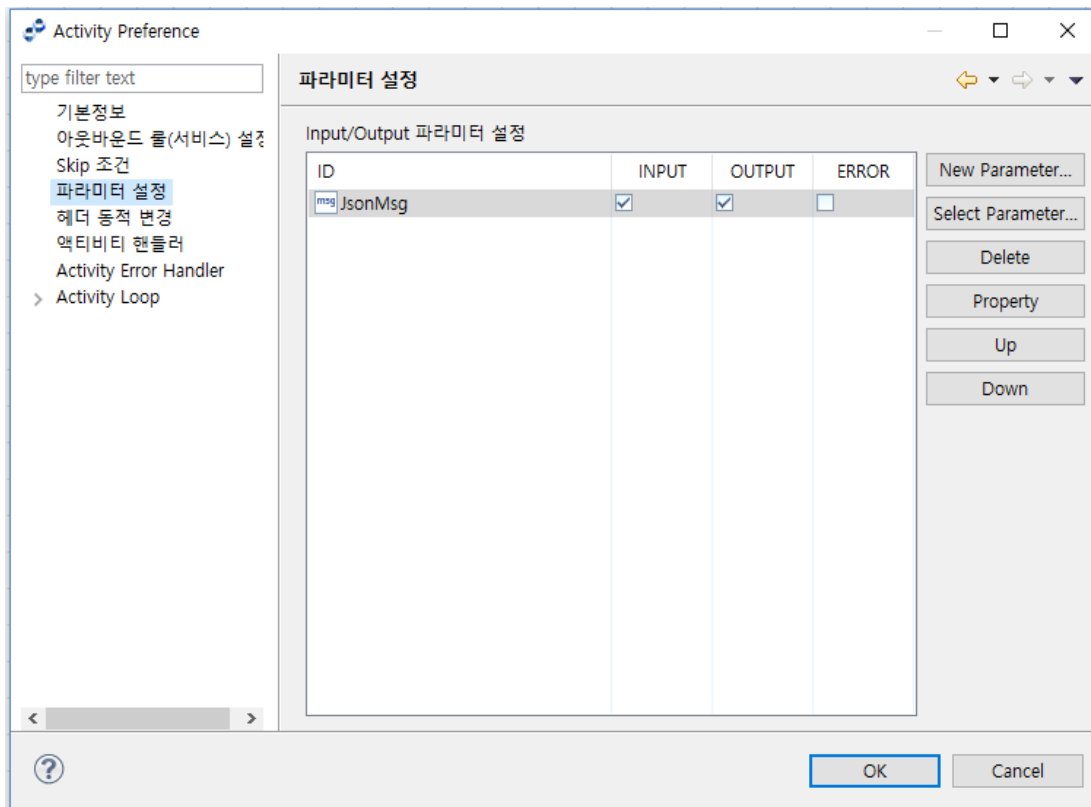


Event Preference - 파라미터 설정

아웃바운드 호출 설정

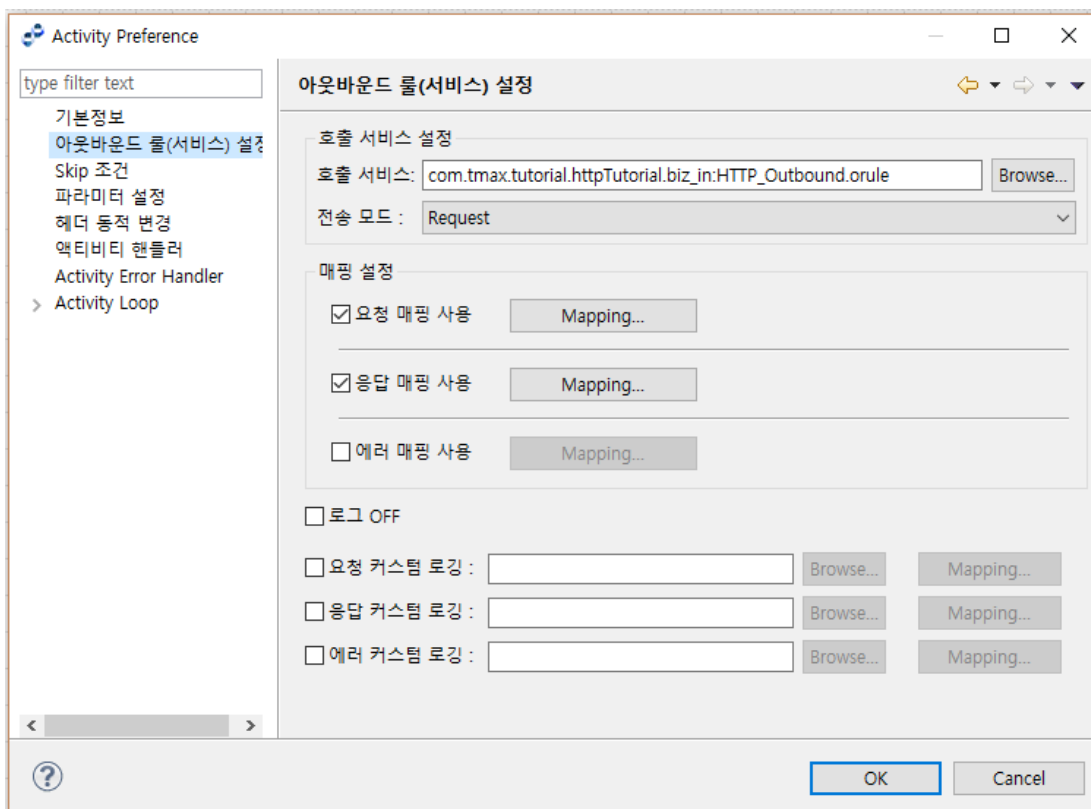
다음의 과정으로 아웃바운드 호출을 설정한다.

1. 플로우 에디터의 [아웃바운드 호출] > [아웃바운드 룰]의 컨텍스트 메뉴에서 [Property]를 선택한다. **Activity Preference** 화면에서 [파라미터 설정]을 선택한다.



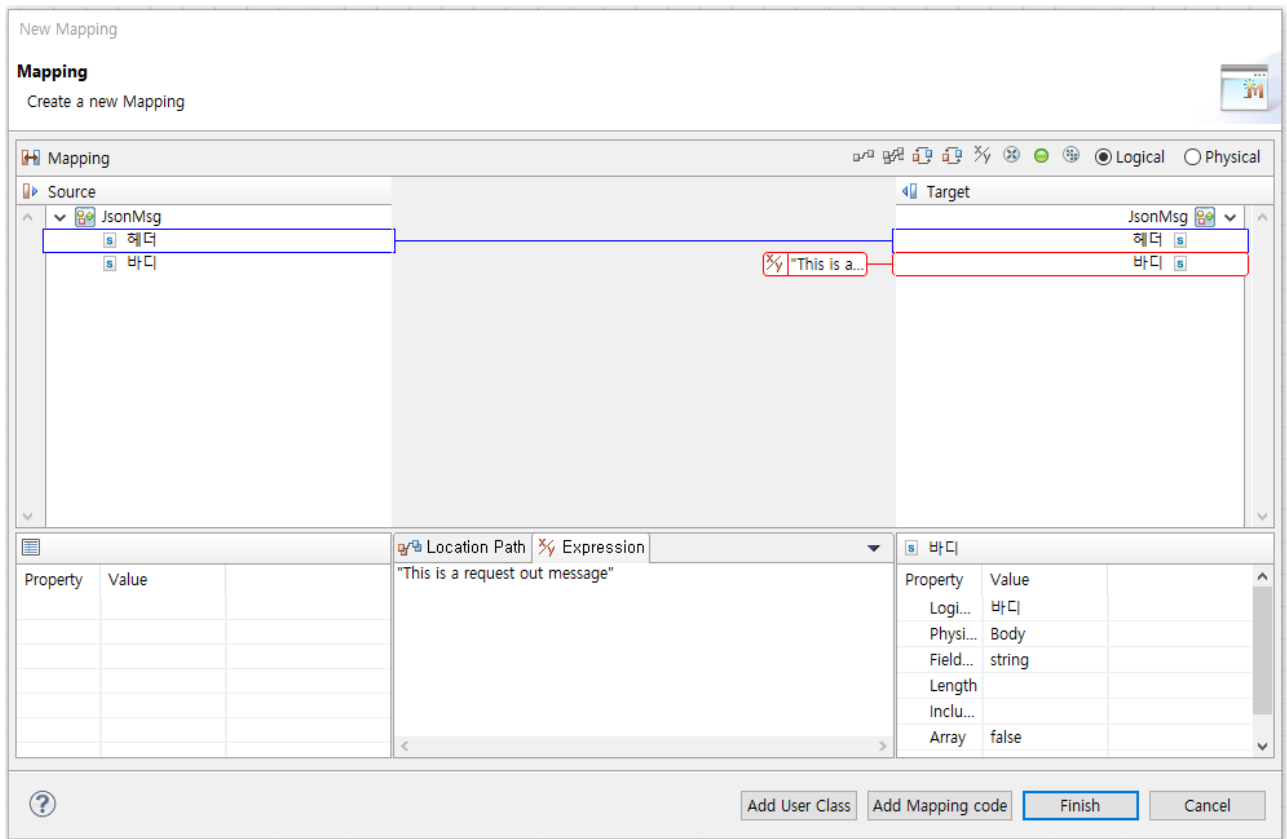
아웃바운드 호출 - 파라미터 설정

2. **Activity Preference** 화면에서 [아웃바운드 룰(서비스) 설정]을 선택한다.

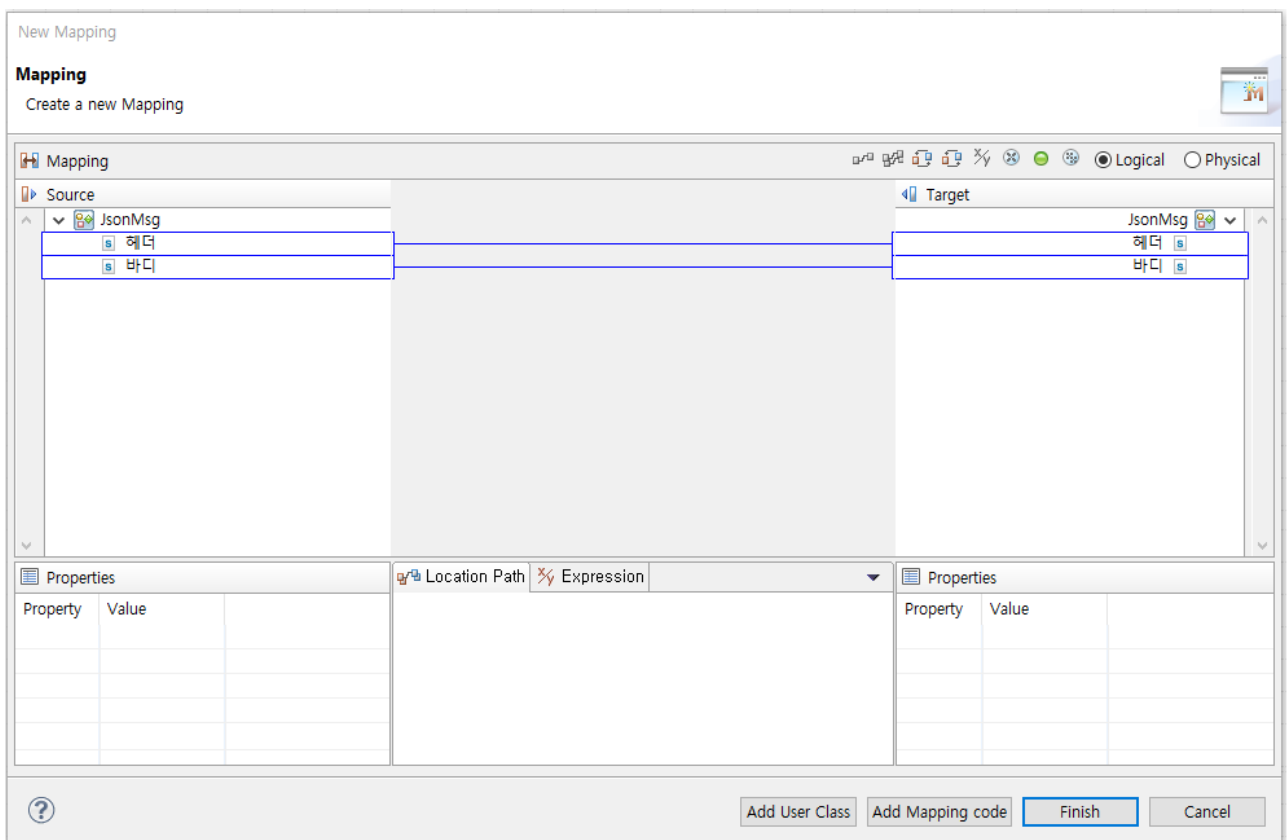


아웃바운드 호출 - 아웃바운드 룰(서비스) 설정

3. **Activity Preference** 화면에서 '요청 매핑 사용'과 '응답 매핑 사용'을 체크하고 [Mapping] 버튼을 클릭해서 각 매핑정보를 설정한다.



아웃바운드 호출 - 아웃바운드 룰(서비스) 설정 - 요청 매핑

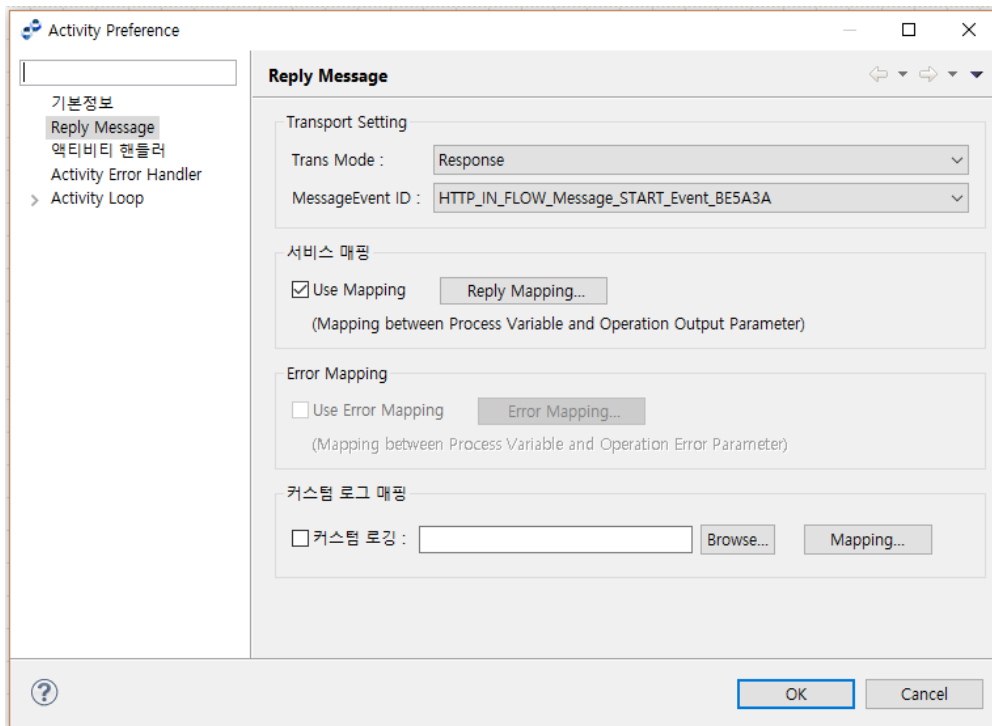


아웃바운드 호출 - 아웃바운드 룰(서비스) 설정 - 응답 매핑

응답 호출 설정

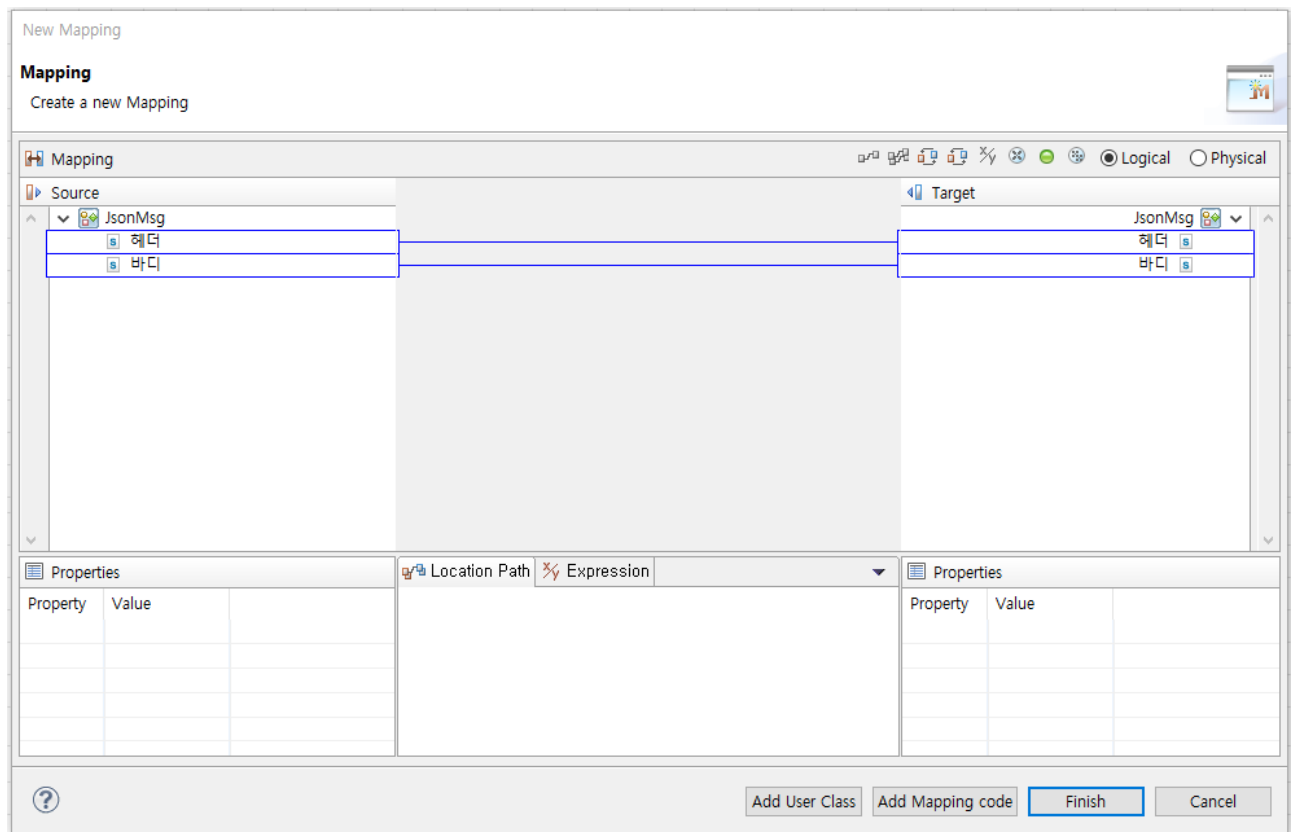
다음의 과정으로 응답 호출 설정을 설정한다.

1. 플로우 에디터의 [응답 송신] > [Reply 메시지]의 컨텍스트 메뉴에서 [Property]를 선택한다. **Activity Preference** 화면에서 [Reply Message]를 선택한다.



응답 호출 - Reply Message

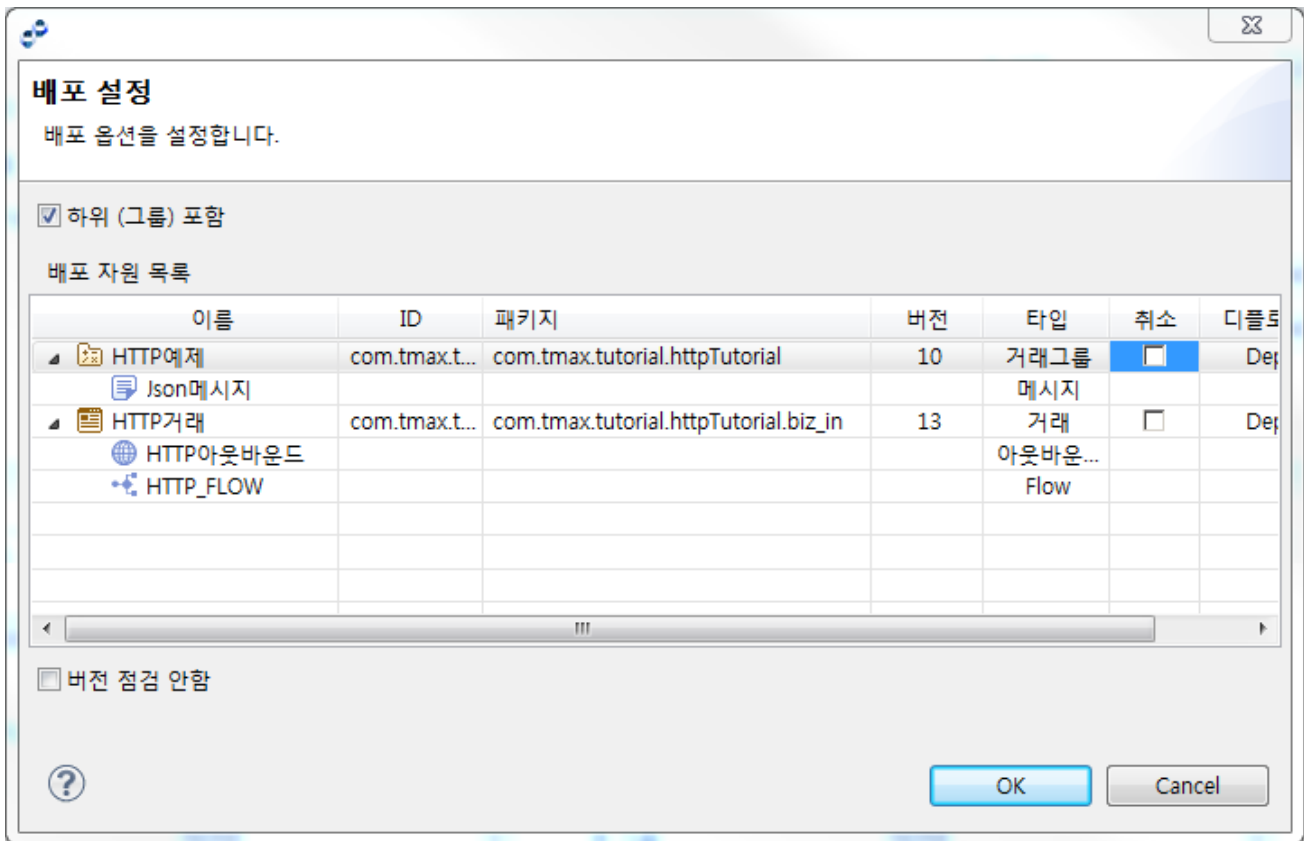
2. **Activity Preference** 화면에서 'Use Mapping'을 체크하고 [Reply Mapping] 버튼을 클릭해서 각 매핑정보를 설정한다. **Mapping** 화면의 Source 영역의 컨텍스트 메뉴에서 [Add source]를 선택하여 JsonMsg를 선택한 후 매핑한다.



응답 호출 - Reply Message - 서비스 매핑 - Reply Mapping

3.6. 배포

거래그룹/거래 배포정보를 설정한다.



거래그룹 / 거래 배포



거래 배포 설정화면에 대한 자세한 사용법은 "AnyLink 스튜디오 안내서"를 참고한다.

3.7. 거래 테스트

클라이언트 프로그램은 AnyLink의 HTTP 아웃바운드를 사용하고, 서버 프로그램은 AnyLink의 HTTP 인바운드를 사용한다.

다음은 테스트를 수행한 결과 화면이다.

트랜잭션 트레이스

기본정보

GUID	7F000001IUNOPLMCCDGFNCH00000061	PGUID	
거래 아이디	com.tmax.tutorial.httpTutorial.biz_in		
거래 이름	HTTP거래		
엔드포인트	HTT_ADT_Tutorial.HTTP_EP_IN		
서버	server1	상태	SUCCESS
요청시간	2019-06-24 23:23:40	경과시간(ms)	64

트레이스

FEP

전제

No	구분	실행 모듈	타임스탬프	GAP(ms)	메시지	에러
1		HTTP Adapter (HTT_ADT_Tutorial) → [HTTP_EP_IN → biz_in]	2019-06-24 23:23:40.483	0		
2		Flow Start (HTTP_IN_FLOW)	2019-06-24 23:23:40.486	3		
3		MESSAGE Event (HTTP_IN_FLOW_Message_START_Event_BE815D)	2019-06-24 23:23:40.487	1		
4		HTTP Activity (HTTP_IN_FLOW_HTTP_BE8261)	2019-06-24 23:23:40.49	3		
5		HTTP Adapter (HTT_ADT_Tutorial) → [HTTP_EP_OUT → HTT_ADT_Tutorial]	2019-06-24 23:23:40.493	3		
6		HTTP Adapter (HTT_ADT_Tutorial) → [HTTP_EP_OUT →]	2019-06-24 23:23:40.509	16		
7		HTTP Activity (HTTP_IN_FLOW_HTTP_BE8261)	2019-06-24 23:23:40.516	7		
8		Reply Activity (HTTP_IN_FLOW_ReplyMessage_BE3164)	2019-06-24 23:23:40.526	10		
9		HTTP Adapter (HTT_ADT_Tutorial) → [→ HTTP_EP_IN]	2019-06-24 23:23:40.528	2		
10		Reply Activity (HTTP_IN_FLOW_ReplyMessage_BE3164)	2019-06-24 23:23:40.53	2		
11		NONE Event (HTTP_IN_FLOW_None_END_Event_BE3562)	2019-06-24 23:23:40.544	14		
12		Flow End (HTTP_IN_FLOW)	2019-06-24 23:23:40.546	2		

⏮

⏪

1

⏩

⏭

50

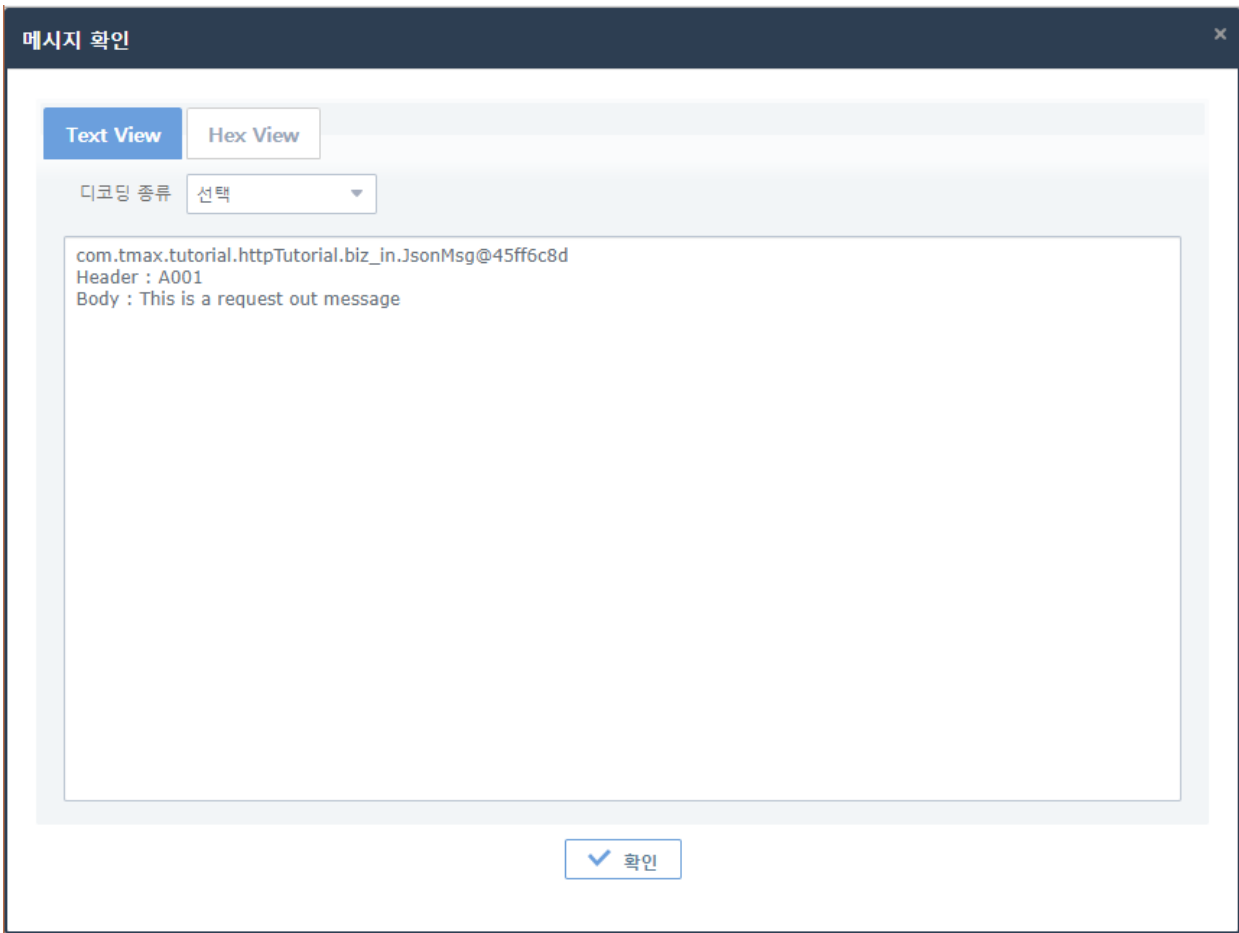
✓

확인

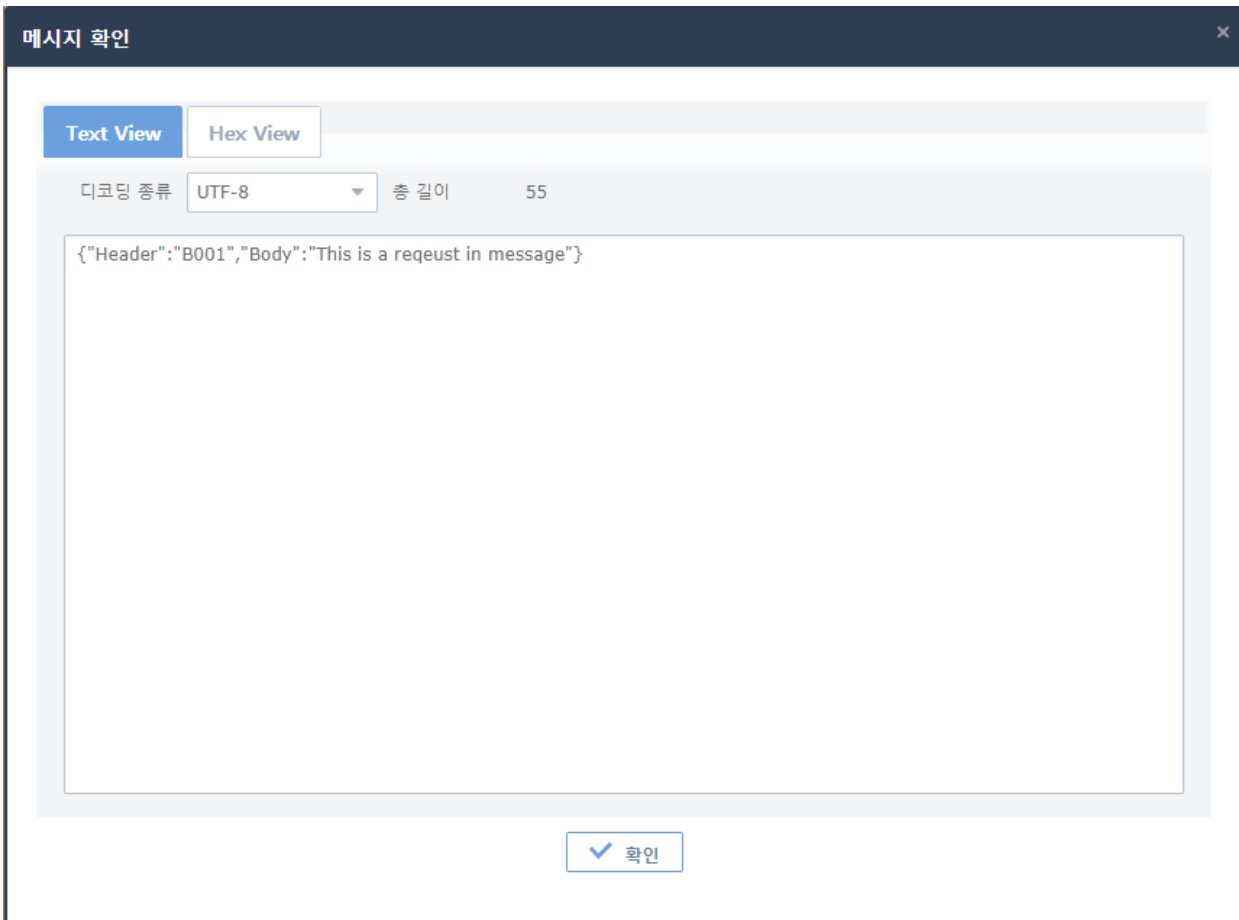
거래 테스트 - 트랜잭션 로그

메시지 확인	
Text View	Hex View
디코딩 종류	UTF-8
총 길이	55
<pre>{ "Header": "A001", "Body": "This is a request in message" }</pre>	
확인 재전송	

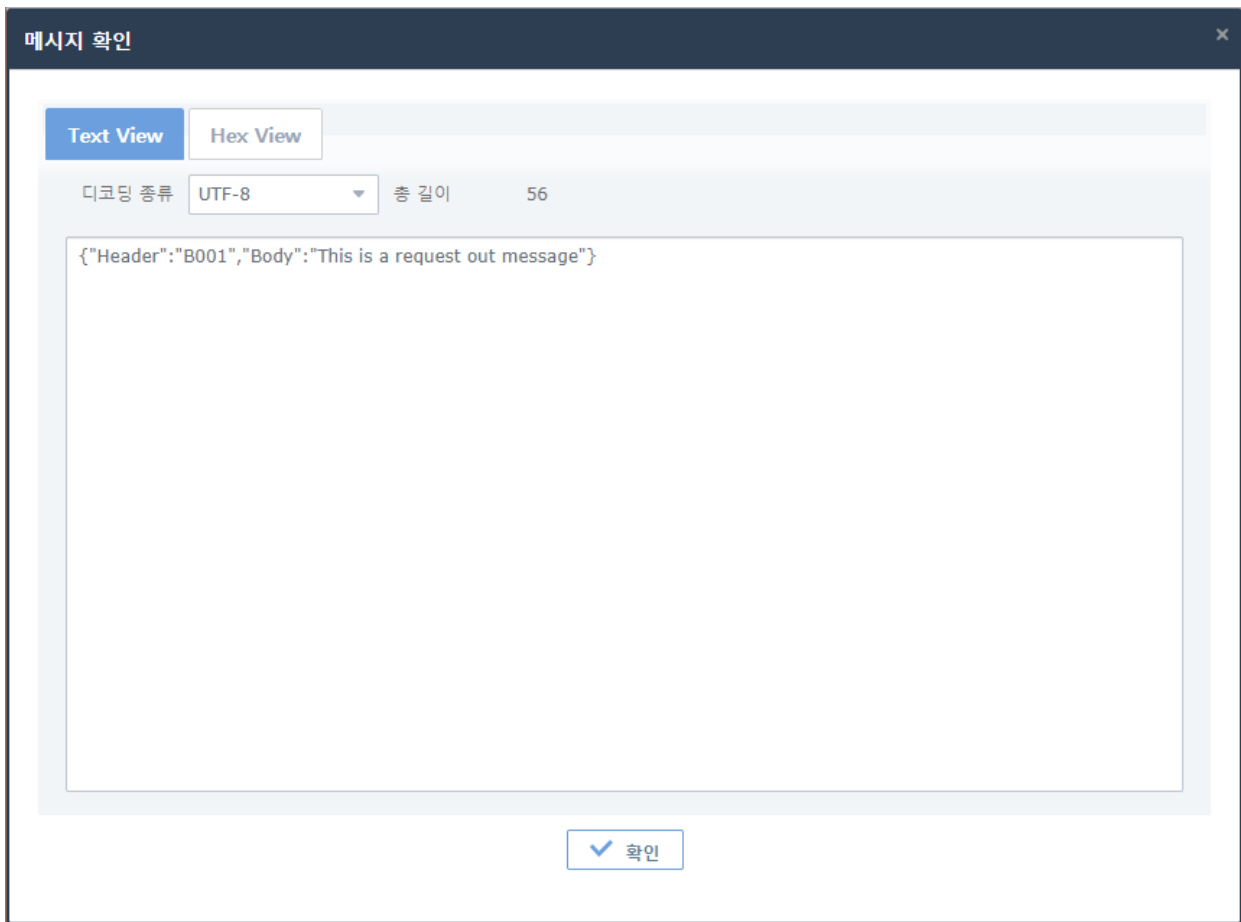
거래 테스트 - 요청 입력 메시지(No. 1)



거래 테스트 - 요청 출력 메시지(No. 4)



거래 테스트 - 응답 입력 메시지(No. 6)



거래 테스트 - 응답 출력 메시지(No. 9)