

특허정보활용서비스 KIPRIS^{Plus}

Open API 이용 가이드

2020. 10





문서명	특허정보활용서비스(KIPRIS ^{Plus}) Open API 이용 가이드		
버전	변경일	변경내용	비고
1.00	2020. 10	최초 작성	
1.01	2020. 10	내용 보완 및 디자인 적용	

버전 : 버전은 1.0부터 시작하여 정수 단위로 변경 관리한다. 변경 발생 시 소수점 아래 번호로 관리하고, 목차 내용이 바뀔 정도의 큰 변경이 발생하면 상위 정수를 변경한다.



[목 차]

I. 서비스 이용 신청 방법 1

1. 회원가입 및 로그인 1

1.1. 회원가입 1

1.2. 로그인 1

2. 상품 및 수수료 확인 2

2.1. 상품 확인 2

2.2. 수수료 확인 5

3. 상품 이용 신청 및 승인 6

3.1. 상품 이용 신청 6

3.2. 상품 이용 승인 7

4. Open API 인증키 확인 및 샘플 호출 테스트 8

4.1. Open API 인증키 확인 8

4.2. 샘플 호출 테스트 8



【 목 차 】

II. Open API 이용 가이드	14
1. 개발 환경 구축	14
1.1. Java 환경 구축	14
1.2. Python 환경 구축	18
2. SOAP API 호출	22
2.1. Java를 이용한 SOAP API 호출 구현	22
3. REST API 호출	28
3.1. Java를 이용한 REST API 호출 구현	28
3.2. Python을 이용한 REST API 호출 구현	33
III. FAQ	38
자주 질문하는 문의 및 답변 안내	38

I 서비스 이용 신청 방법

1 회원 가입 및 로그인

1.1. 회원가입

- KIPRIS^{Plus} 웹사이트 메인화면 상단의 회원가입 기능을 통해 회원 종류(일반, 단체)을 선택 및 회원 정보를 입력하여 진행합니다.

* 단체회원의 경우에는 회원가입 신청 후 관리자의 승인이 필요합니다.

① 웹사이트 오른쪽 상단의 회원가입 선택

② 일반회원 및 단체회원 선택

③ 이용약관, 개인정보 수집 및 이용 동의의 체크

④ 회원정보 작성 및 가입

1.2. 로그인

- KIPRIS^{Plus} 웹사이트 메인화면 상단의 기능을 통해 로그인 합니다.

① 웹사이트 오른쪽 상단의 로그인 선택

② 회원 선택 및 로그인

2 상품 및 수수료 확인

2.1. 상품 확인

□ 웹사이트 메인화면의 검색 기능과 메뉴(데이터목록 → 서비스 유형별, 분류 체계별, 무료 상품정보) 선택을 통해 상품화면으로 접근합니다.

1-1) 검색 기능을 통한 상품화면 접근

<p>① 오른쪽 상단의 검색 기능 이용</p> 	<p>② 검색된 상품 확인</p> 
---	---

1-2) 메뉴 선택(데이터목록 → 서비스 유형별)을 통한 상품화면 접근

<p>① 데이터 목록 탭의 서비스 유형별 선택</p> 	<p>② 상품 접근 및 확인</p> 
---	--

1-3) 메뉴 선택(데이터목록 → 분류 체계별)을 통한 상품화면 접근

<p>① 데이터 목록 탭의 분류 체계별 선택</p> 	<p>② 상품 접근 및 확인</p> 
--	--

1-4) 메뉴 선택(데이터목록 → 무료 상품정보)을 통한 상품화면 접근

<p>① 데이터 목록 탭의 무료 상품정보 선택</p>	<p>② 상품 접근 및 확인</p>
--------------------------------------	----------------------------

상품 선택을 통해 접근된 상세화면 내 상품 설명 및 API 서비스 종류(SOAP, REST)에 따른 기능별 명세서 정보를 확인합니다.

2-1) 상품정보 확인(상품 설명)

<p>① 상품설명-01</p>	<p>② 상품설명-02</p>
-------------------------	-------------------------

2-2) 상품정보 확인(서비스 유형, API 분류 및 기능 선택 방법)

- ① : API 개발 및 이용 방법에 따라 SOAP, REST 방식이 존재합니다. 신규 개발일 경우 REST를 선택합니다.(2019년부터 신규 상품 및 기능에 대해서는 REST 방식의 API로만 제공합니다.)
- ② : API 기능을 종류별로 분리하여 제공합니다. API 분류 정보를 선택합니다.
- ③ : API 기능을 선택합니다. 각 API 기능별로 다른 명세서가 존재합니다.

2-2) API 기능별 명세서 확인(API 세부 입·출력 정보 및 샘플 정보)

오퍼레이션 명 (Operation Name)

IPC정보 - patentIpclInfo

요청 주소 (Request URL)

http://plus.kipris.or.kr/

입력값 (Request Parameter)

Variable (변수명)	Level	Variable Type (변수 타입)	Description (설명)
applicationNumber	1	String	출원번호

입력값(샘플)

```
http://plus.kipris.or.kr/openapi/rest/patUtiModInfoSearchService/patentIpclInfo?applicationNumber=1020060118886&access
```

출력값 (Response Parameter)

Variable (변수명)	Level	Variable Type (변수 타입)	Description (설명)
patentIpclInfo	1		
InternationalpatentclassificationNumber	2	String	IPC번호
InternationalpatentclassificationDate	2	String	IPC일자

출력값(샘플)

```
<body>
<items>
<patentIpclInfo>
<InternationalpatentclassificationNumber>H04W 72/06</InternationalpatentclassificationNumber>
<InternationalpatentclassificationDate>(2010.01)</InternationalpatentclassificationDate>
</patentIpclInfo>
<patentIpclInfo>
<InternationalpatentclassificationNumber>H04W 28/22</InternationalpatentclassificationNumber>
<InternationalpatentclassificationDate>(2010.01)</InternationalpatentclassificationDate>
</patentIpclInfo>
<patentIpclInfo>
<InternationalpatentclassificationNumber>H04W 52/42</InternationalpatentclassificationNumber>
<InternationalpatentclassificationDate>(2010.01)</InternationalpatentclassificationDate>
</patentIpclInfo>
</items>
</body>
```

2.2. 수수료 확인

- 웹사이트 메인화면의 메뉴(이용안내→서비스 수수료 안내)를 선택하여 이용하려는 상품의 수수료 정보를 확인합니다.

① 이용안내 탭의 서비스 수수료 안내 선택

The screenshot shows the KIPRIS Plus homepage with the 'Service Fee Information' tab highlighted. On the left, there's a banner for 'New Product Introduction' (2020 KIPRIS Plus New Product Introduction). On the right, there's a section titled 'Public Notice' with several items listed, each with a date and a link. A large blue arrow points upwards from the 'Service Fee Information' tab towards the list of notices.

② 서비스 수수료안내 화면에서 수수료 정보 확인

The screenshot shows the 'Service Fee Information' page. At the top, there's a navigation bar: Home > Usage Guide > Service Fee Information. Below it, there's a title 'Service Fee Information' and a download button. The main content area has a title 'KIPRIS Plus Service Fee Information' and a note about the period (2020.6.30 (Mon) ~ VAT Period). There are several bullet points below, followed by a table.

① 한국 산업재산권	데이터 상품명	Bulk Data		Open API
		당해연도	과거분 전체	
한국 공보	01 특허·실용·등록·증명·변동·변경·등록·증명·변동·변경·분류·코드·이력	3,273,463	18,004,044	
	02 특허·실용·등록·증명·변동·변경·등록·증명·변동·변경·분류·코드·이력	3,273,463	18,004,044	
	03 특허·실용·등록·증명·변동·변경·등록·증명·변동·변경·분류·코드·이력	682,516	3,753,836	
	04 특허·실용·등록·증명·변동·변경·등록·증명·변동·변경·분류·코드·이력	84,304	463,674	
	05 특허·실용·등록·증명·변동·변경·등록·증명·변동·변경·분류·코드·이력	84,304	463,674	
	06 특허·실용·등록·증명·변동·변경·등록·증명·변동·변경·분류·코드·이력	84,304	463,674	
	07 특허·파일리	84,304	463,674	
	08 청구항·변동·이력	682,516	3,753,836	
	09 존속기간·연장·등록·증명·변동·변경·분류·코드·이력	84,304	463,674	
	10 디자인·공보	682,516	3,753,836	
	11 디자인·통합·이력	682,516	3,753,836	
	12 디자인·형태·검색·DB	무료		
	13 디자인·분류·코드·이력	84,304	463,674	※ 월 1,000건 이내 무료 조회 가능

3 상품 이용 신청 및 승인

3.1. 상품 이용 신청

KIPRIS^{Plus} 웹사이트 내 상품화면으로 접근하여 서비스 신청 및 일괄 신청 기능(로그인 후 진행 가능)을 통해 이용 신청합니다.

1) 서비스 신청(한 개 상품에 대해 신청)

<p>① 상품 화면 내 서비스 신청 선택</p> 	<p>② 정보 입력 후 서비스 이용 신청</p> 
---	--

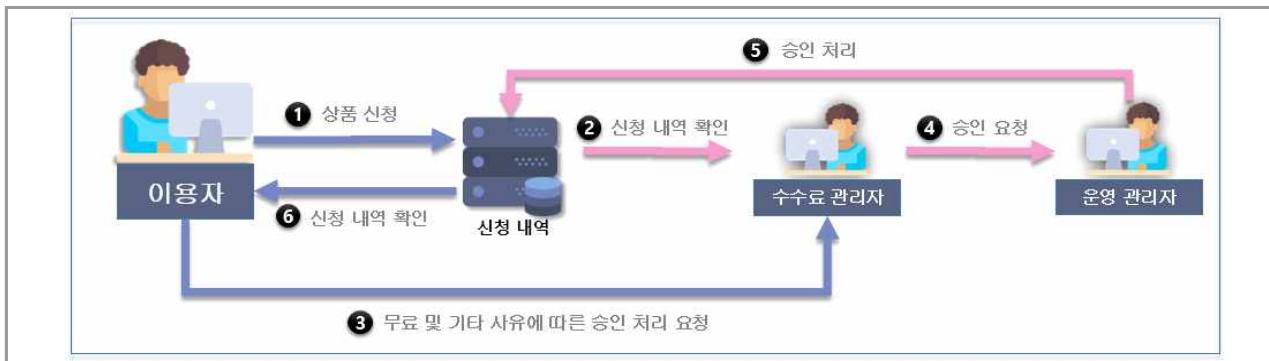
2) 서비스 일괄 신청(동시에 여러 개의 상품을 신청)

<p>① 상품 화면 내 서비스 일괄 신청 선택</p> 	<p>② 정보 입력 후 서비스 일괄 이용 신청</p> 
---	--

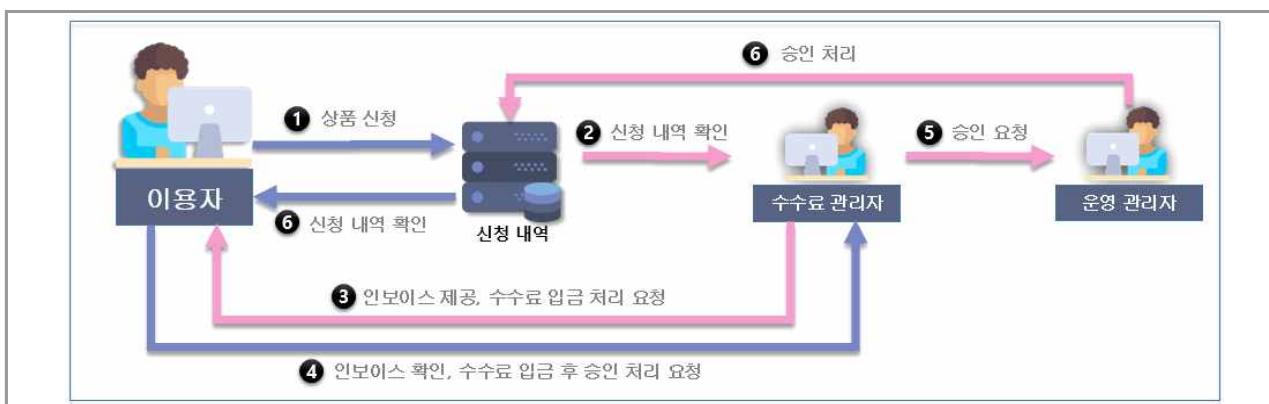
3.2. 상품 이용 승인

- 결재 유형(유료, 무료 등)에 따라 KIPRIS^{Plus} 관리자에 관련 정보를 전달하여 이용 신청한 상품에 대한 승인 처리를 요청합니다.

1-1) 무료 및 기타(데이터기프트 등) 관련 승인 프로세스



1-2) 유료 수수료 관련 승인 프로세스



- 이용 신청한 상품이 KIPRIS^{Plus} 관리자를 통해 승인되면 그 결과에 대해 마이페이지의 서비스 구매내역에서 확인 가능합니다.

① 웹사이트 오른쪽 상단의 마이페이지 선택	② 서비스 구매내역 확인
<p>The screenshot shows the KIPRIS Plus homepage with a navigation bar at the top. A blue arrow points to the '마이페이지' (MyPage) link in the top right corner. Below the navigation bar, there's a section for 'KIPRIS Plus 신규 데이터 상품 안내' (New Data Product Introduction) with a '30' badge. The main content area displays news items and a search bar.</p>	<p>The screenshot shows the 'Service Purchase History' page under the 'MyPage' section. It lists recent purchases with columns for 번호 (Number), 유형 (Type), 서비스명 (Service Name), 구분 (Category), 신청일 (Application Date), 시작일 (Start Date), 종료일 (End Date), and 승인여부 (Approval Status). Each row has a 'View' button (보기) and a 'Delete' button (삭제). The table header includes 'Home > 마이페이지 > 서비스 구매내역'.</p>

4

Open API 인증키 확인 및 샘플 호출 테스트

4.1. Open API 인증키 확인

- 로그인 후 마이페이지의 API KEY 관리에서 유형별(SOAP, REST)로 해당 아이디의 인증키를 확인 가능하며 필요 시 재발급 가능합니다.

1) API(SOAP, REST) 인증키 확인

<p>① 웹사이트 오른쪽 상단의 마이페이지 선택</p> 	<p>② API KEY 관리 확인</p> 
--	---

4.2. 샘플 호출 테스트

- 관리자로부터 이용 승인받은 상품의 API 유형별(SOAP, REST)로 발급받은 인증키가 정상적으로 동작하는지 확인이 필요합니다.

1) 서비스 구매내역에서 승인된 상품 및 API 유형을 확인합니다.

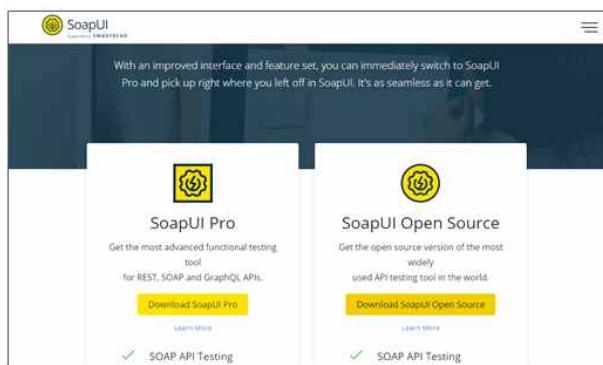
<p>① 웹사이트 오른쪽 상단의 마이페이지 선택</p> 	<p>② 서비스 구매내역 확인</p> 
--	---

2) SOAP 방식의 API 호출 테스트를 통한 인증키 정상여부 확인

* 이용 목적에 따라 진행 전 공식사이트 내 라이선스 정책을 확인 부탁드립니다.
[\(https://www.soapui.org/downloads/soapui/\)](https://www.soapui.org/downloads/soapui/)

① SoapUI Open Source 다운로드 및 설치

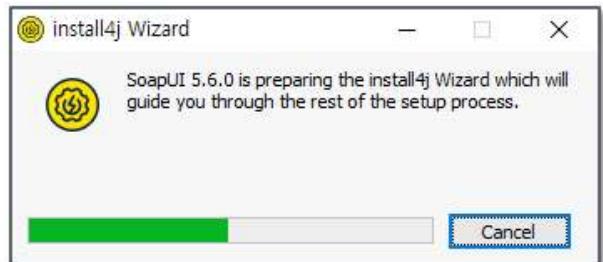
① SoapUI 다운로드 사이트 접근



② SoapUI 설치 프로그램 실행



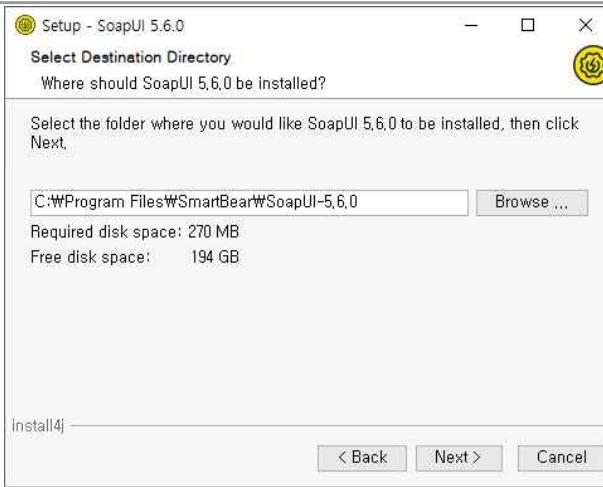
③ SoapUI 설치-01



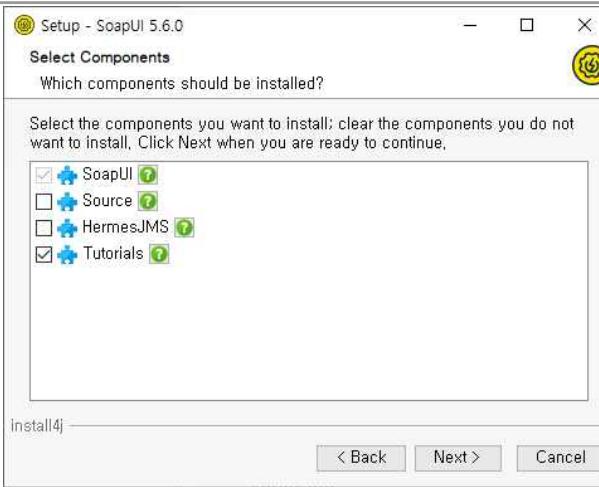
④ SoapUI 설치-02



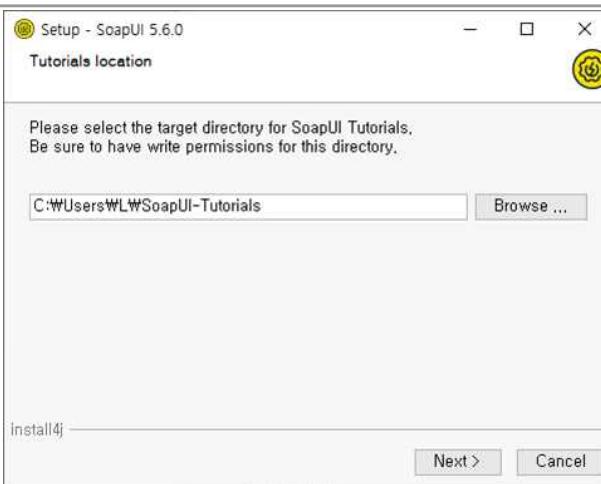
⑤ SoapUI 설치-03



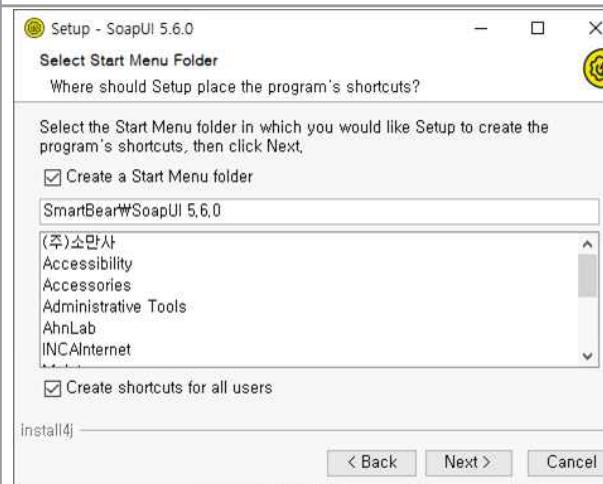
⑥ SoapUI 설치-04



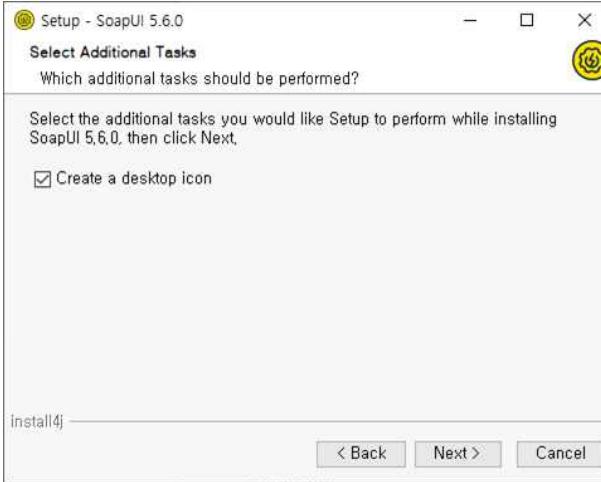
⑦ SoapUI 설치-05



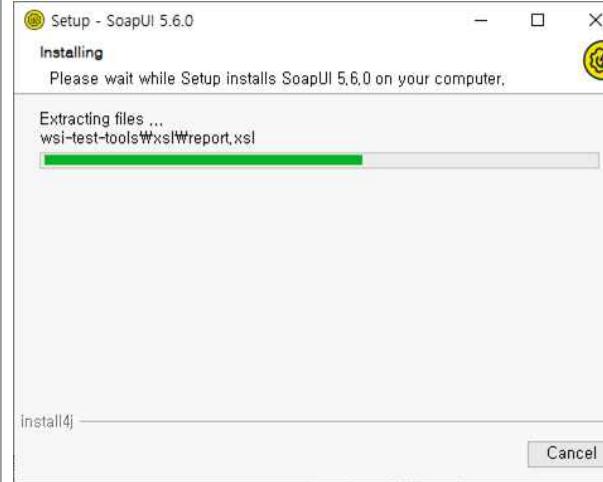
⑧ SoapUI 설치-06



⑨ SoapUI 설치-07



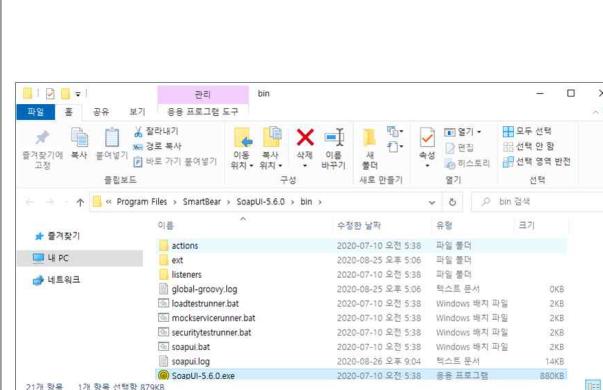
⑩ SoapUI 설치-08



⑪ SoapUI 설치-09



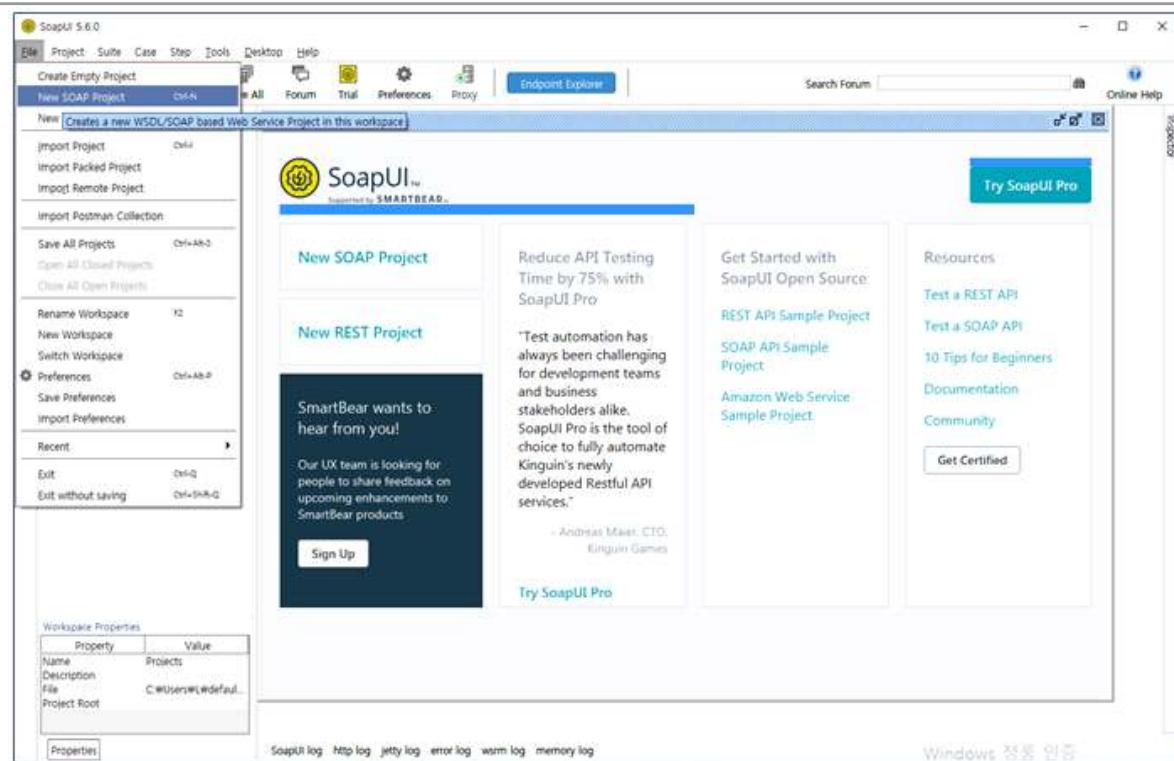
⑫ SoapUI 설치-10



③ SOAP UI 실행 및 API 호출 결과 확인

① SoapUI 실행 화면

[메뉴] File → New SOAP Project 선택



② KIPRIS^{Plus} 상품화면 내 SOAP의 WSDL 정보 확인

오퍼레이션 명 (Operation Name)
서지상세정보 - bibliographicInfo
요청 주소 (Request URL)
http://plus.kipris.or.kr/kiprisplusws/services/PatentBibliographicInfoService?wsdl

③ New SOAP Project 생성

[항목] Initial WSDL에 KIPRIS^{Plus}에서 제공한 WSDL 정보 입력 후 OK

New SOAP Project	
Creates a WSDL/SOAP based Project in this workspace	
Project Name:	특허-실용-공개-등록공보
Initial WSDL:	:kr/kiprisplusws/services/PatentBibliographicInfoService?wsdl
Create Requests:	<input checked="" type="checkbox"/> Create sample requests for all operations?
Create TestSuite:	<input type="checkbox"/> Creates a TestSuite for the imported WSDL
Relative Paths:	<input type="checkbox"/> Stores all file paths in project relatively to project file (requires save)
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

④ SOAP API 기능 선택

WSDL 등록 후 생성된 API 기능의 Request 정보 확인

The screenshot shows the KIPRIS Plus service interface. On the left, the 'Projects' tree is visible with '特허-실용 공개-등록공보' selected. Under it, 'PatentBibliographicInfoServiceSoap1' is expanded, showing various service functions like 'abstractinfo', 'agentinfo', 'applicantinfo', 'bibliographicInfo', and 'biblioSummaryInfo'. 'biblioSummaryInfo' is further expanded to reveal 'SOAP Request 1'. The right pane is titled 'SOAP Request 1' and shows the XML structure of the request message:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <xsd:userId>키프리스플러스ID</xsd:userId>
    <xsd:userKey>SOAP 인증키</xsd:userKey>
  </soapenv:Header>
  <soapenv:Body>
    <xsd:biblioSummaryInfo>
      <!--Optional:-->
      <xsd:applicationNumber?></xsd:applicationNumber>
    </xsd:biblioSummaryInfo>
  </soapenv:Body>
</soapenv:Envelope>

```

⑤ SOAP API 호출

인증정보(아이디, 인증키) 및 기능별 입력정보(출원번호 등)를 작성

The screenshot shows the KIPRIS Plus service interface. The 'Projects' tree is visible with '特허-실용 공개-등록공보' selected. Under it, 'PatentBibliographicInfoServiceSoap1' is expanded, showing 'biblioSummaryInfo' which is selected. The right pane is titled 'SOAP Request 1' and shows the XML structure of the request message, which now includes an 'applicationNumber' element:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <xsd:userId>키프리스플러스ID</xsd:userId>
    <xsd:userKey>SOAP 인증키</xsd:userKey>
  </soapenv:Header>
  <soapenv:Body>
    <xsd:biblioSummaryInfo>
      <!--Optional:-->
      <xsd:applicationNumber>1020080025578</xsd:applicationNumber>
    </xsd:biblioSummaryInfo>
  </soapenv:Body>
</soapenv:Envelope>

```

[참고] SOAP API 호출 시 Header 내 인증정보(아이디, 인증키) 작성 예시

```

<soapenv:Header>
  <xsd:userId>ID</xsd:userId>
  <xsd:userKey>AccessKey</xsd:userKey>
</soapenv:Header>

```

⑥ SOAP API 기능 호출 및 결과 확인

인증정보(아이디, 인증키) 및 기능별 입력정보(출원번호 등)를 작성 완료 후 호출(▶)하여 결과값 확인

The screenshot shows the KIPRIS Plus service interface. The 'Projects' tree is visible with '特허-실용 공개-등록공보' selected. Under it, 'PatentBibliographicInfoServiceSoap1' is expanded, showing 'biblioSummaryInfo' which is selected. The right pane is titled 'SOAP Request 1' and shows the XML structure of the request message. Below it, the 'Raw' tab shows the XML structure of the response message, which includes detailed information about the patent application:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <xsd:userId>키프리스플러스ID</xsd:userId>
    <xsd:userKey>SOAP 인증키</xsd:userKey>
  </soapenv:Header>
  <soapenv:Body>
    <xsd:biblioSummaryInfoResponse xmlns:ns="http://plus.kipris.or.kr/xsd">
      <ns:biblioSummaryInfoResponse xmlns:ns="http://plus.kipris.or.kr/xsd">
        <ns:applicationNumber>1020080025578</ns:applicationNumber>
        <ns:applicationDate>2008.03.19</ns:applicationDate>
        <ns:applicationFlag>Y</ns:applicationFlag>
        <ns:claimCount>5</ns:claimCount>
        <ns:examinerName>21진영</ns:examinerName>
        <ns:fnlDisposal>결정(일반)</ns:fnlDisposal>
        <ns:inventionTitle>Apparatus for preventing entrance of tunnel</ns:inventionTitle>
        <ns:inventionTitleEng>도로노면 세척 및 갈방방지시스템</ns:inventionTitleEng>
        <ns:openDate>2008.07.09</ns:openDate>
        <ns:publicationNumber>10-0844582-0009</ns:publicationNumber>
        <ns:registerDate>2008.07.01</ns:registerDate>
        <ns:registerStatus>등록</ns:registerStatus>
        <ns:translatorSubmitDate></ns:translatorSubmitDate>
      </ns:biblioSummaryInfoResponse>
    </soapenv:Body>
</soapenv:Envelope>

```

3) REST 방식의 API 호출 테스트를 통한 인증키 정상여부 확인

① REST API 호출 및 결과 확인

① KIPRIS^{Plus} 상품화면 내 입력값(샘플) 정보 확인

오퍼레이션 명 (Operation Name)					
서지요약정보 - getBibliographySumryInfoSearch					
요청 주소 (Request URL)					
http://plus.kipris.or.kr/					
입력값 (Request Parameter)					
항목명(영문)	항목명(국문)	항목크기	항목구분	샘플데이터	항목설명
applicationNumber	출원번호	20	1	10-2005-0050026	
입력값(샘플) http://plus.kipris.or.kr/kipo-api/kipi/patUtilModInfoSearchService/getBibliographySumryInfoSearch?applicationNumber=10200					

② 웹 브라우저 주소창에 REST API 호출 URL 정보 입력 및 결과 확인



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

<?xml version="1.0" encoding="UTF-8"?>
<response>
  <header>
    <requestMsgID></requestMsgID>
    <responseTime>2020-08-26 18:02:38.238</responseTime>
    <responseMsgID></responseMsgID>
    <successYN>Y</successYN>
    <resultCode>00</resultCode>
    <resultMsg>NORMAL SERVICE.</resultMsg>
  </header>
  <body>
    <item>
      <applicationDate>2005.06.10</applicationDate>
      <applicationNumber>10-2005-0050026</applicationNumber>
      <totalCount>1</totalCount>
      <finalDisposition>특허청(일반)</finalDisposition>
      <inventionTitle>휴대용 슬리밍기 및 그의 음성분석방법</inventionTitle>
      <inventionTitleEng>MOBILE TERMINAL AND METHOD FOR ANALYZING SOUND THEREOF</inventionTitleEng>
      <openDate>2006.12.14</openDate>
      <openNumber>10-2006-0128519</openNumber>
      <originalApplicationDate></originalApplicationDate>
      <originalApplicationKind></originalApplicationKind>
      <originalApplicationNumber></originalApplicationNumber>
      <originalExaminationRequestDate>2010.04.20</originalExaminationRequestDate>
      <originalExaminationRequestFlag>Y</originalExaminationRequestFlag>
      <publicationDate>2012.01.06</publicationDate>
      <publicationNumber></publicationNumber>
      <registerDate>2012.01.03</registerDate>
      <registerNumber>10-104063-0000</registerNumber>
      <registerStatus>소멸</registerStatus>
      <translationSubmitDate></translationSubmitDate>
    </item>
  </body>
</count>
<count>1</count>
<pageNo>1</pageNo>
<totalCount>1</totalCount>
</count>
</response>

```

[참고] REST API 호출 URL 작성 예시

<http://plus.kipris.or.kr/kipo-api/kipi/patUtilModInfoSearchService/getBibliographySumryInfoSearch?applicationNumber=출원번호&ServiceKey=AccessKey>

II

Open API 이용 가이드

1

개발 환경 구축

1.1. Java 환경 구축

□ 프로그램 버전 및 설치

< 개발 환경 구축을 위한 프로그램별 버전 >

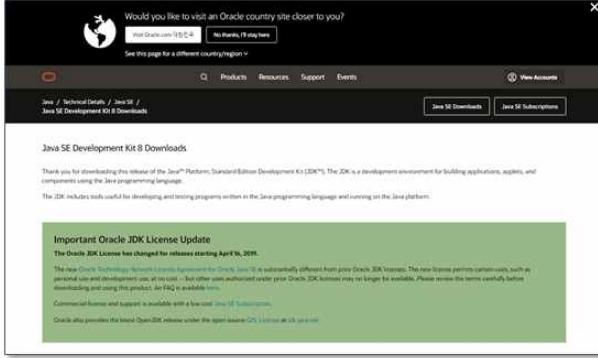
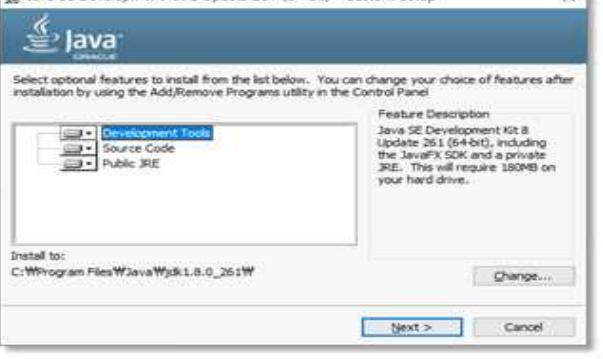
프로그램 종류	프로그램 상세 명칭	버전
Java	Java SE Development Kit 8	8u261
Eclipse	Eclipse IDE for Enterprise Java Developers	4.16
Tomcat	Apache Tomcat 9.0	9.0.38

※ SOAP API 개발 환경 구축은 Java 8 이하 버전을 추천 드립니다.

1) Java SE Development Kit 8(8u261) 다운로드 및 설치

※ 이용 목적에 따라 진행 전 공식사이트 내 라이선스 정책을 확인 부탁드립니다.

[\(https://oracle.com/java/technologies/javase/jdk8-downloads.html\)](https://oracle.com/java/technologies/javase/jdk8-downloads.html)

① Java 다운로드 사이트 접근	② 운영체제 호환 설치 파일 다운로드																																										
	<table border="1"> <thead> <tr> <th>Product / File Description</th> <th>File Size</th> <th>Download</th> </tr> </thead> <tbody> <tr> <td>Linux ARM S2 Hard Float ABI</td> <td>75.4 MB</td> <td>jdk-8u261-linux-arm-s2-vfp-hftar.gz</td> </tr> <tr> <td>Linux ARM dL1 Hard Float ABI</td> <td>70.3 MB</td> <td>jdk-8u261-linux-dl1-vfp-hfttar.gz</td> </tr> <tr> <td>Linux x86 RPM Package</td> <td>12152 MB</td> <td>jdk-8u261-linux-i586.rpm</td> </tr> <tr> <td>Linux x86 Compressed Archive</td> <td>156.81 MB</td> <td>jdk-8u261-linux-i586.tar.gz</td> </tr> <tr> <td>Linux x86 RPM Package</td> <td>12153 MB</td> <td>jdk-8u261-linux-i586rpm</td> </tr> <tr> <td>Linux x64 Compressed Archive</td> <td>156.48 MB</td> <td>jdk-8u261-linux-x64tar.gz</td> </tr> <tr> <td>macOS x86</td> <td>205.94 MB</td> <td>jdk-8u261-macosx-x64.dmg</td> </tr> <tr> <td>Solaris SPARC 64-bit (SVR4 package)</td> <td>125.77 MB</td> <td>jdk-8u261-solaris-sparcv9tar.Z</td> </tr> <tr> <td>Solaris SPARC 64-bit</td> <td>88.72 MB</td> <td>jdk-8u261-solaris-sparcv9tar.gz</td> </tr> <tr> <td>Solaris x64 (SVR4 package)</td> <td>154.23 MB</td> <td>jdk-8u261-solaris-x64tar.Z</td> </tr> <tr> <td>Solaris x64</td> <td>92.47 MB</td> <td>jdk-8u261-solaris-x64tar.gz</td> </tr> <tr> <td>Windows x86</td> <td>154.52 MB</td> <td>jdk-8u261-windows-i586.exe</td> </tr> <tr> <td>Windows x64</td> <td>166.28 MB</td> <td>jdk-8u261-windows-x64.exe</td> </tr> </tbody> </table>	Product / File Description	File Size	Download	Linux ARM S2 Hard Float ABI	75.4 MB	jdk-8u261-linux-arm-s2-vfp-hftar.gz	Linux ARM dL1 Hard Float ABI	70.3 MB	jdk-8u261-linux-dl1-vfp-hfttar.gz	Linux x86 RPM Package	12152 MB	jdk-8u261-linux-i586.rpm	Linux x86 Compressed Archive	156.81 MB	jdk-8u261-linux-i586.tar.gz	Linux x86 RPM Package	12153 MB	jdk-8u261-linux-i586rpm	Linux x64 Compressed Archive	156.48 MB	jdk-8u261-linux-x64tar.gz	macOS x86	205.94 MB	jdk-8u261-macosx-x64.dmg	Solaris SPARC 64-bit (SVR4 package)	125.77 MB	jdk-8u261-solaris-sparcv9tar.Z	Solaris SPARC 64-bit	88.72 MB	jdk-8u261-solaris-sparcv9tar.gz	Solaris x64 (SVR4 package)	154.23 MB	jdk-8u261-solaris-x64tar.Z	Solaris x64	92.47 MB	jdk-8u261-solaris-x64tar.gz	Windows x86	154.52 MB	jdk-8u261-windows-i586.exe	Windows x64	166.28 MB	jdk-8u261-windows-x64.exe
Product / File Description	File Size	Download																																									
Linux ARM S2 Hard Float ABI	75.4 MB	jdk-8u261-linux-arm-s2-vfp-hftar.gz																																									
Linux ARM dL1 Hard Float ABI	70.3 MB	jdk-8u261-linux-dl1-vfp-hfttar.gz																																									
Linux x86 RPM Package	12152 MB	jdk-8u261-linux-i586.rpm																																									
Linux x86 Compressed Archive	156.81 MB	jdk-8u261-linux-i586.tar.gz																																									
Linux x86 RPM Package	12153 MB	jdk-8u261-linux-i586rpm																																									
Linux x64 Compressed Archive	156.48 MB	jdk-8u261-linux-x64tar.gz																																									
macOS x86	205.94 MB	jdk-8u261-macosx-x64.dmg																																									
Solaris SPARC 64-bit (SVR4 package)	125.77 MB	jdk-8u261-solaris-sparcv9tar.Z																																									
Solaris SPARC 64-bit	88.72 MB	jdk-8u261-solaris-sparcv9tar.gz																																									
Solaris x64 (SVR4 package)	154.23 MB	jdk-8u261-solaris-x64tar.Z																																									
Solaris x64	92.47 MB	jdk-8u261-solaris-x64tar.gz																																									
Windows x86	154.52 MB	jdk-8u261-windows-i586.exe																																									
Windows x64	166.28 MB	jdk-8u261-windows-x64.exe																																									
③ 다운로드 받은 설치파일 실행-01	④ 다운로드 받은 설치파일 실행-02																																										
																																											

⑤ 다운로드 받은 설치파일 실행 03



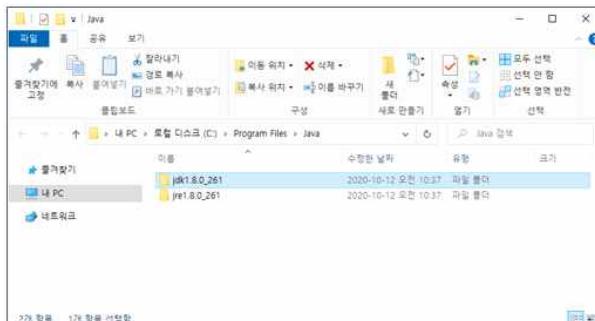
⑥ 다운로드 받은 설치파일 실행-04



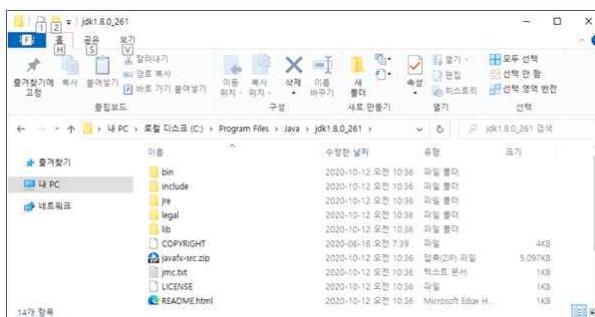
⑦ 다운로드 받은 설치파일 실행-05



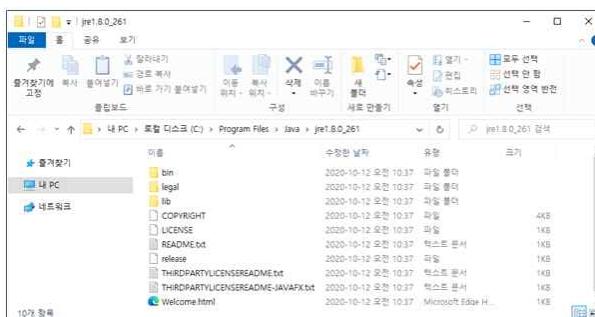
⑧ 설치경로 내 폴더 및 파일 생성 확인-01



⑨ 설치경로 내 폴더 및 파일 생성 확인-02



⑩ 설치경로 내 폴더 및 파일 생성 확인-03



2) Eclipse IDE for Enterprise Java Developers(4.16) 다운로드 및 설치

* 이용 목적에 따라 진행 전 공식사이트 내 라이선스 정책을 확인 부탁드립니다.

(<https://www.eclipse.org/downloads/packages/>)

① Eclipse 다운로드 사이트 접근

The screenshot shows the Eclipse IDE for Java Developers download page. It features a main header with the product name and a 'Download' button. Below the header, there are sections for 'RELATED LINKS' and 'MORE DOWNLOADS'. Under 'MORE DOWNLOADS', it lists several Eclipse versions: Eclipse 2020-09 (4.17), Eclipse 2020-05 (4.16), Eclipse 2019-12 (4.14), Eclipse 2019-09 (4.13), Eclipse 2019-06 (4.12), and Eclipse 2019-03 (4.11).

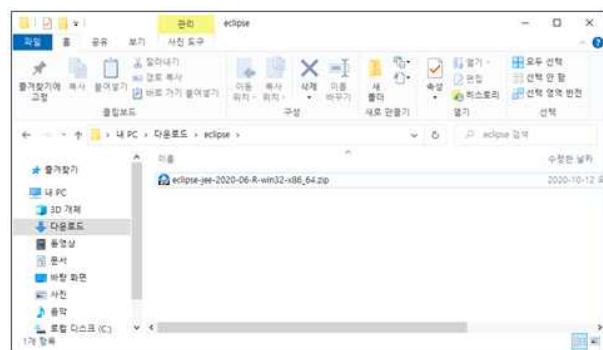
② Java 버전과 호환 가능한 4.16 버전 선택

This screenshot shows the same Eclipse IDE for Java Developers download page as above, but with a different version selected. The 'Eclipse 2020-09 (4.17)' link is highlighted, while the others are grayed out. The 'MORE DOWNLOADS' section also reflects this selection.

③ Enterprise Java Developers 다운로드

The screenshot shows the 'Try the Eclipse Installer 2020-06 R' page. It highlights the 'Java Developers' package, which has 130,730 downloads. The package size is 309 MB and it includes Java IDE, Tools for Web Services, JPA and Data Tools, JDT, Mylyn, Maven and Gradle, Git, and more.

④ 다운로드 받은 ZIP 파일 확인



⑤ ZIP 파일 압축 해제 후 eclipse 폴더 확인

A Windows File Explorer window showing the contents of the 'eclipse' folder, which was extracted from the ZIP file. The folder contains subfolders like 'configuration', 'dropins', 'features', 'p2', 'plugins', 'readme', 'eclipseproduct', 'artifacts.xml', 'eclipse.ini', and 'eclipse.exe'.

⑥ eclipse 폴더를 C드라이브 경로로 이동

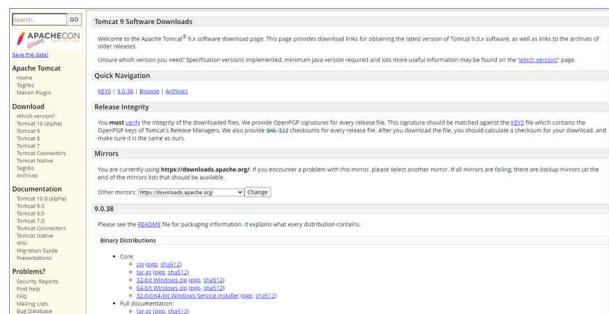
A Windows File Explorer window showing the 'eclipse' folder moved to the root of the 'C:\' drive. The folder structure is identical to the one in the previous screenshot, indicating a successful move.

3) Apache Tomcat 9.0(9.0.38) 다운로드 및 설치

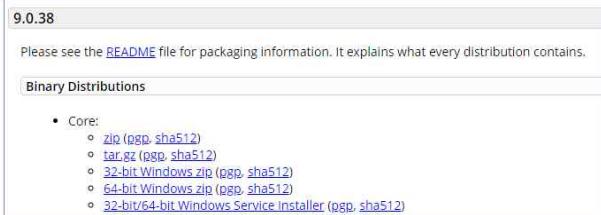
* 본 가이드에서 Apache Tomcat 관련 부분은 참고용으로 안내드리고 있습니다.
이용 목적에 따라 진행 전 공식사이트 내 라이선스 정책을 확인 부탁드립니다.

(<https://tomcat.apache.org/download-90.cgi>)

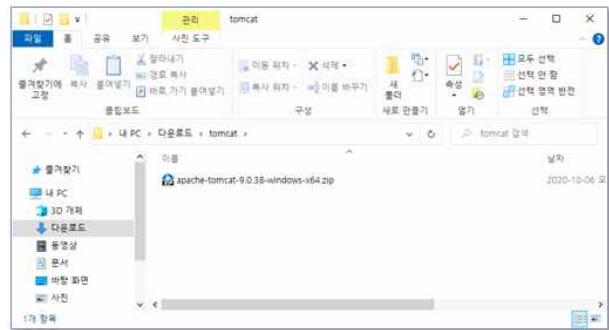
① Apache Tomcat 다운로드 사이트 접근



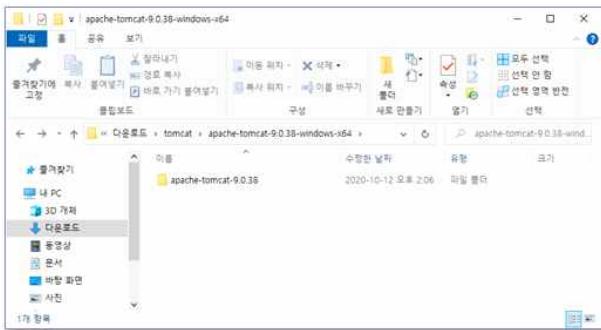
② Java 버전과 호환 가능한 9.0.38 버전 선택



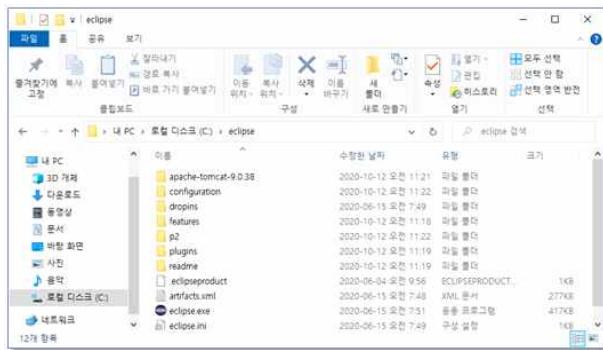
③ 다운로드 받은 ZIP 파일 확인



④ 파일 압축 해제 후 tomcat 폴더 확인



⑤ tomcat 폴더를 앞 단계에서 생성한 eclipse 경로로 이동



2.2. Python 환경 구축

□ 프로그램 및 버전

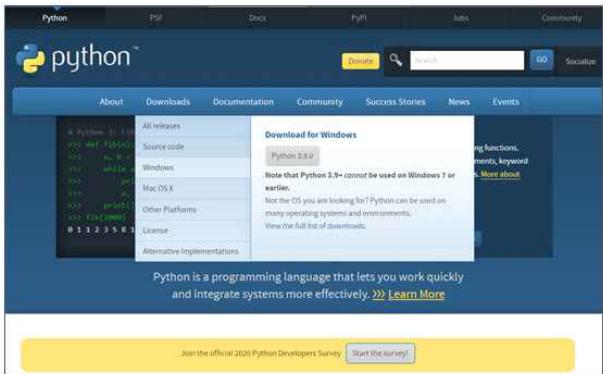
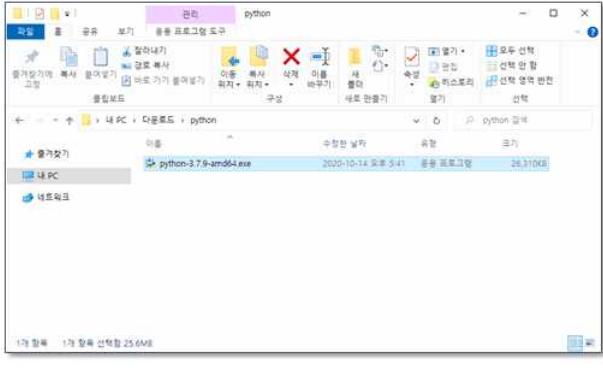
< 개발 환경 구축을 위한 프로그램별 버전 >

프로그램 종류	프로그램 상세 명칭	버전
Python	Python	3.7.9
PyCharm	PyCharm Community	2020.2.3

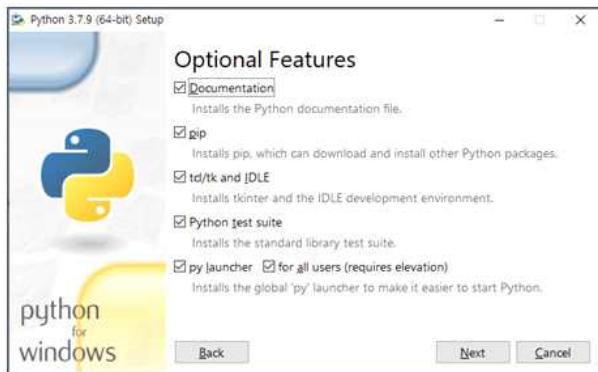
1) Python(3.7.9) 다운로드 및 설치

※ 이용 목적에 따라 진행 전 공식사이트 내 라이선스 정책을 확인 부탁드립니다.

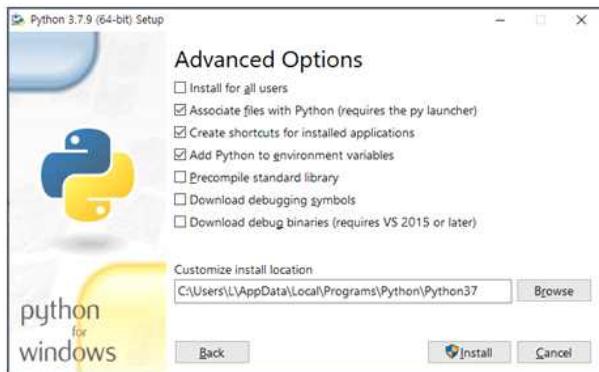
(<https://www.python.org/downloads/>)

① Python 다운로드 사이트 접근	② 운영체제 호환 설치 파일 다운로드
	<p>▪ Python 3.7.9 - Aug. 17, 2020 Note that Python 3.7.9 cannot be used on Windows XP or earlier.</p> <ul style="list-style-type: none"> ▪ Download Windows help file ▪ Download Windows x86-64 embeddable zip file ▪ Download Windows x86-64 executable installer ▪ Download Windows x86-64 web-based installer ▪ Download Windows x86 embeddable zip file ▪ Download Windows x86 executable installer ▪ Download Windows x86 web-based installer
③ 다운로드 받은 설치파일 확인	④ 다운로드 받은 설치파일 실행 -01
	

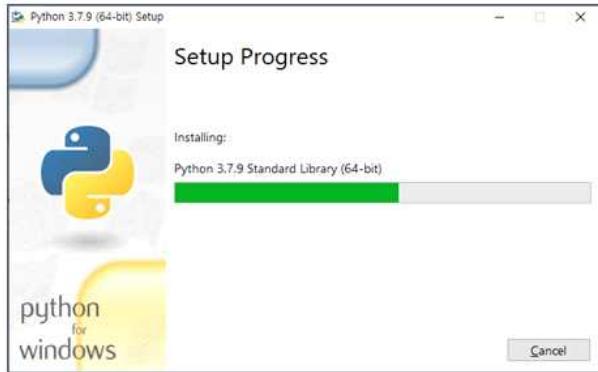
⑤ 다운로드 받은 설치파일 실행 -02



⑥ 다운로드 받은 설치파일 실행 -03



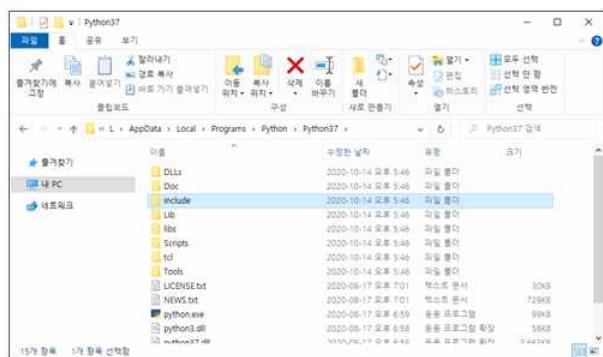
⑦ 다운로드 받은 설치파일 실행 -04



⑧ 다운로드 받은 설치파일 실행 -05



⑨ 설치경로 내 폴더 및 파일 생성 확인



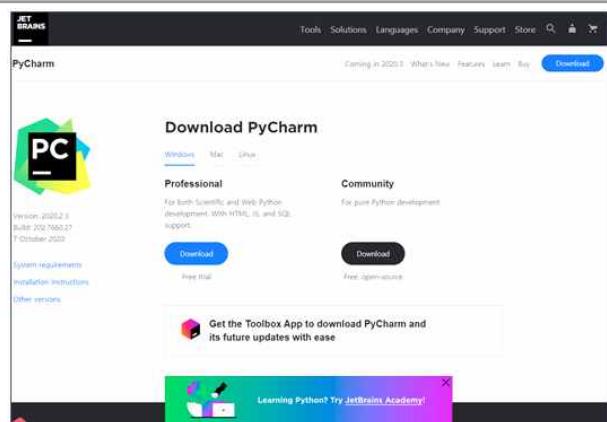
2) Pycharm Community(2020.2.3) 다운로드 및 설치

* 이용 목적에 따라 진행 전 공식사이트 내 라이선스 정책을 확인 부탁드립니다.
[\(https://jetbrains.com/pycharm/download/\)](https://jetbrains.com/pycharm/download/)

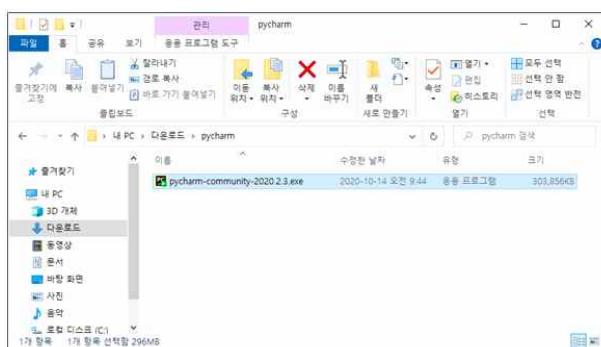
① Pycharm 다운로드 사이트 접근



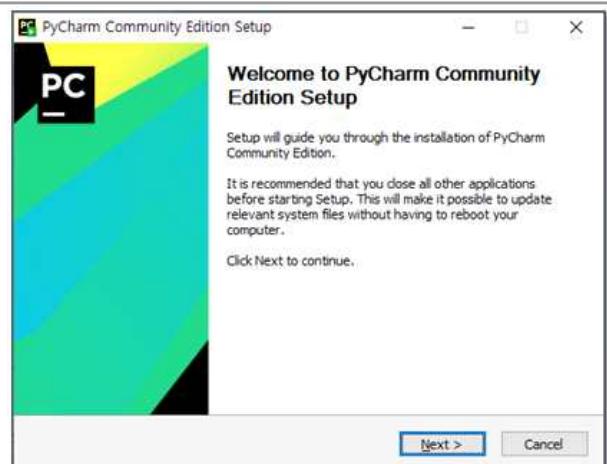
② Pycharm Community 다운로드



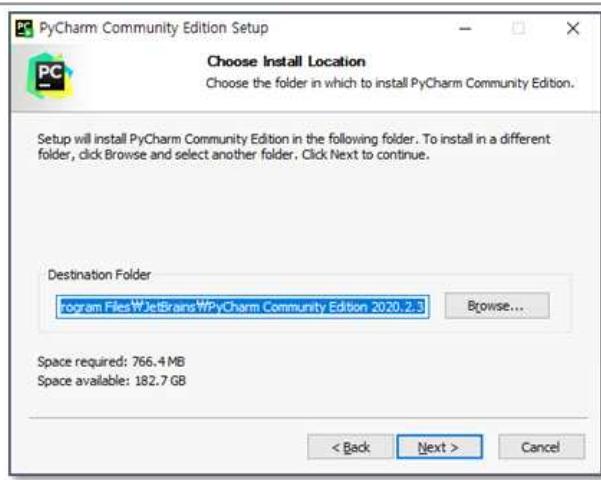
③ 다운로드 받은 설치파일 확인



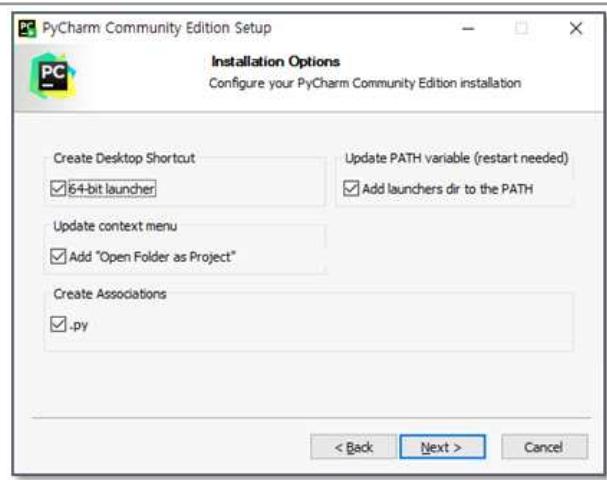
④ 다운로드 받은 설치파일 실행 -01



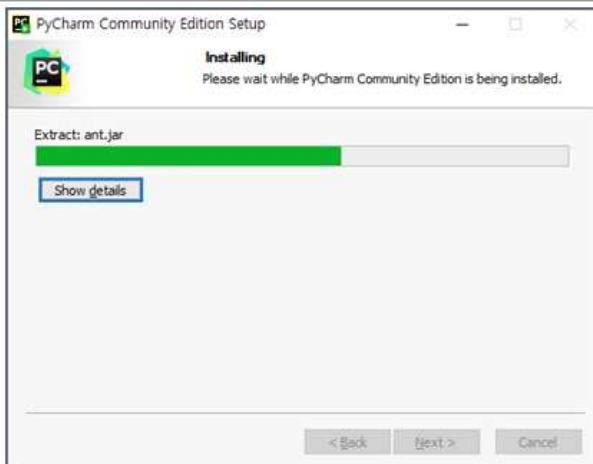
⑤ 다운로드 받은 설치파일 실행 -02



⑥ 다운로드 받은 설치파일 실행 -03



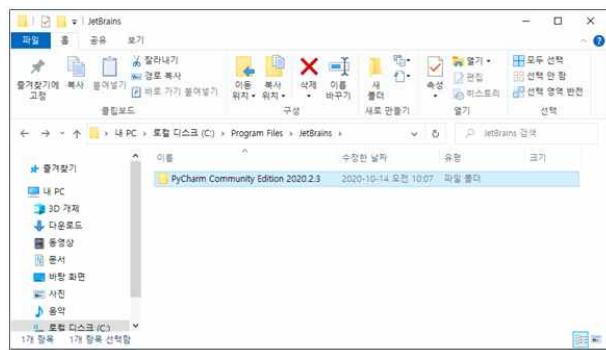
⑦ 다운로드 받은 설치파일 실행 - 04



⑧ 다운로드 받은 설치파일 실행 - 05



⑨ 설치경로 내 폴더 및 파일 생성 확인



2 SOAP API 호출

2.1. Java를 이용한 SOAP API 호출 구현

※ KIPRIS^{Plus}에서는 신규 API 기능 구현 시 SOAP API 기능으로 제공하지 않고 REST API 기능으로만 제공합니다. 가능하면 REST API로 이용 부탁드립니다.

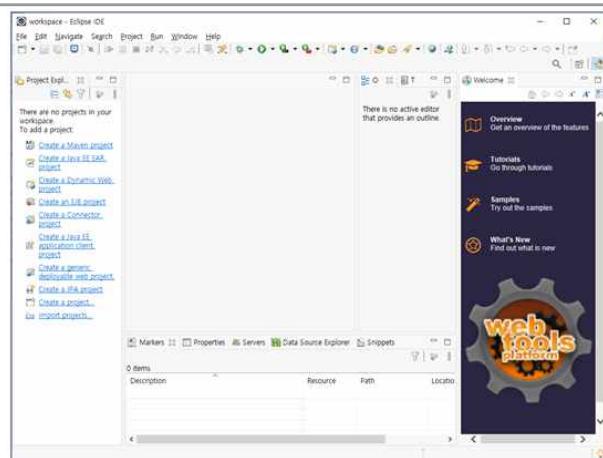
- Eclipse 실행부터 프로그램 구현까지 SOAP API 호출 방법에 대한 가이드 문서이며, 기본적인 프로그램 구현 방식으로 작성되었습니다.

1) Eclipse 실행

① Eclipse 실행	② 작업경로 지정
③ Eclipse 최초 실행 시 화면	④ Eclipse 초기 화면

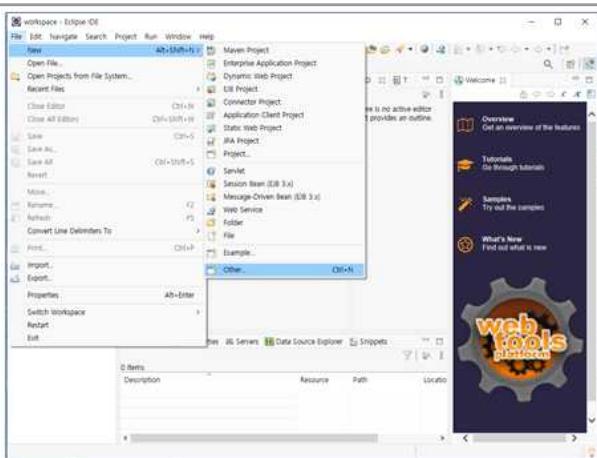
2) 신규 Project 생성

① Eclipse 초기 화면



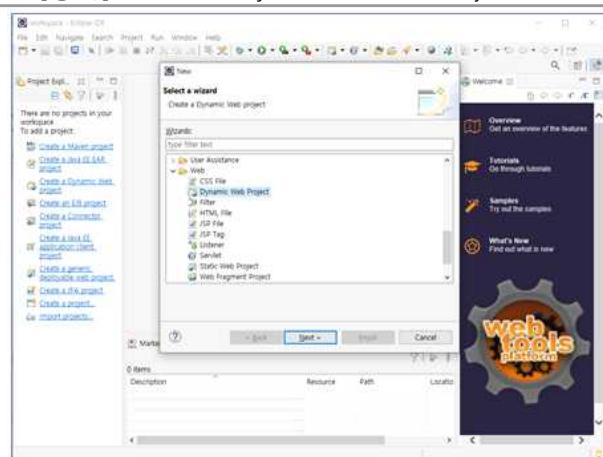
② 신규 프로젝트 생성-01

[메뉴] File → New → Other 선택



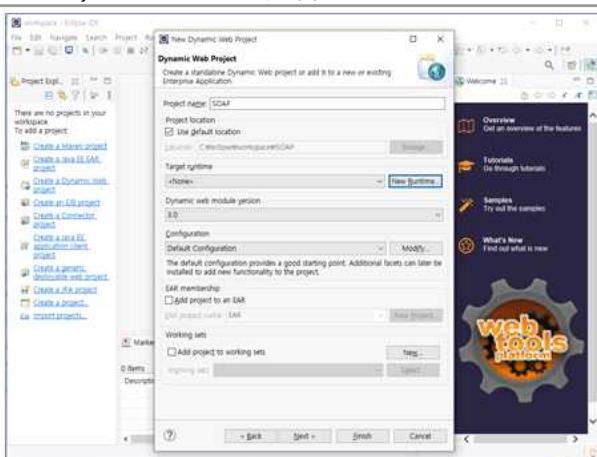
③ 신규 프로젝트 생성-02

[항목] Web → Dynamic Web Project 선택



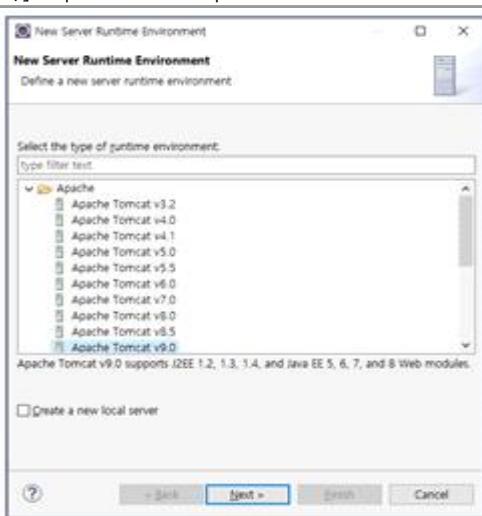
④ 신규 프로젝트 생성-03

Project name 작성 및 New Runtime 클릭



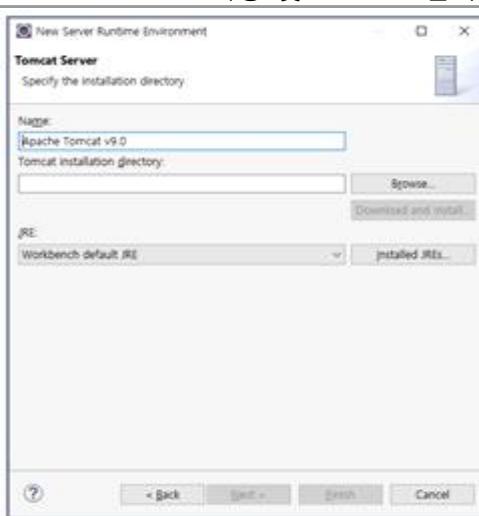
⑤ 신규 프로젝트 생성-04

[항목] Apache → Apache Tomcat v.9.0 선택



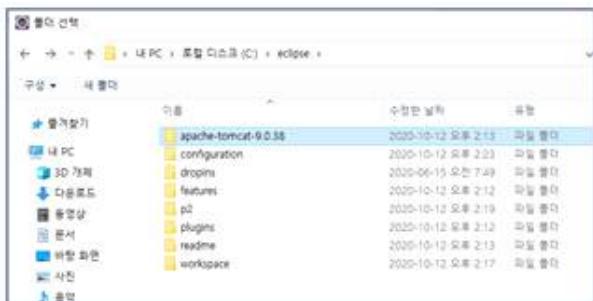
⑥ 신규 프로젝트 생성-05

Server Name 작성 및 Browse 클릭



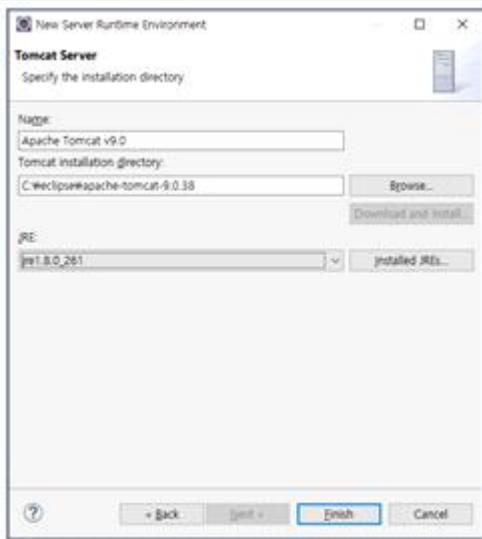
⑦ 신규 프로젝트 생성-06

기존에 생성한 tomcat 폴더 선택



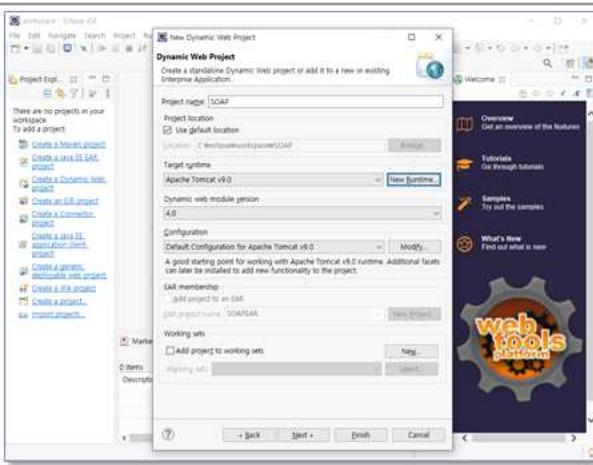
⑧ 신규 프로젝트 생성-07

기존에 생성한 JRE 선택



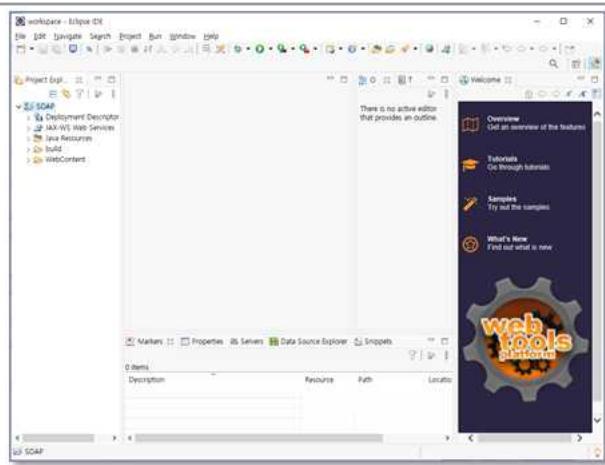
⑨ 신규 프로젝트 생성-08

Project 생성을 위한 기본 정보 입력 완료



⑩ 신규 프로젝트 생성-09

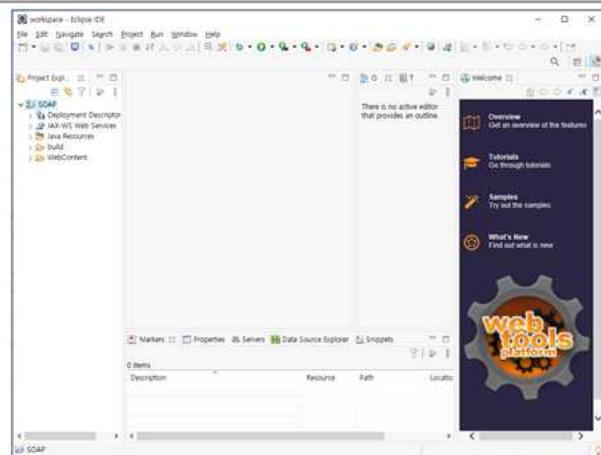
생성된 신규 Project 확인



3) 신규 Project 내 Web Client 생성

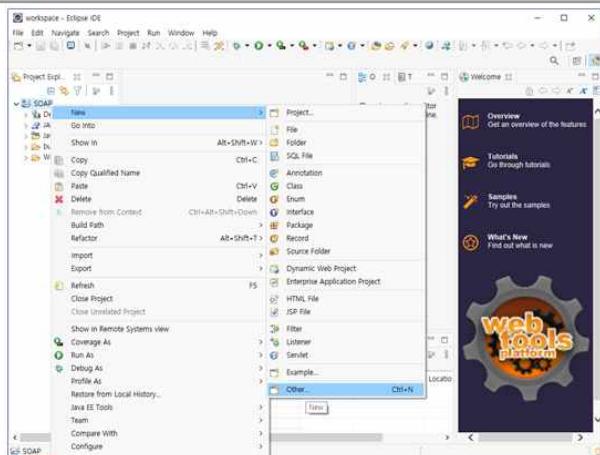
① 신규 생성된 Project 확인

신규 생성된 SOAP Project에 마우스 우 클릭



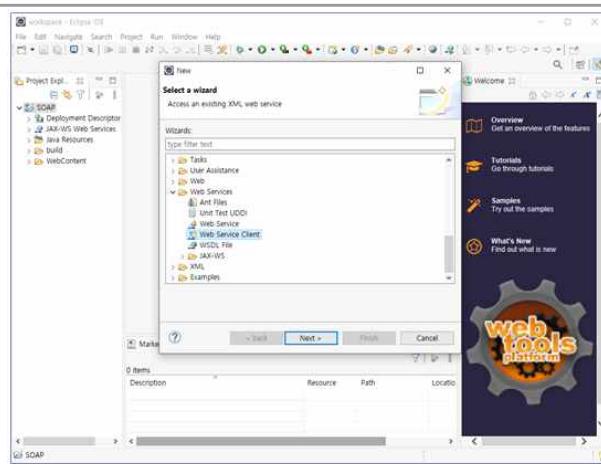
② Web Client 생성-01

[서브 메뉴] New → Other 선택



③ Web Client 생성-02

[항목] Web Service → Web Service Client 선택



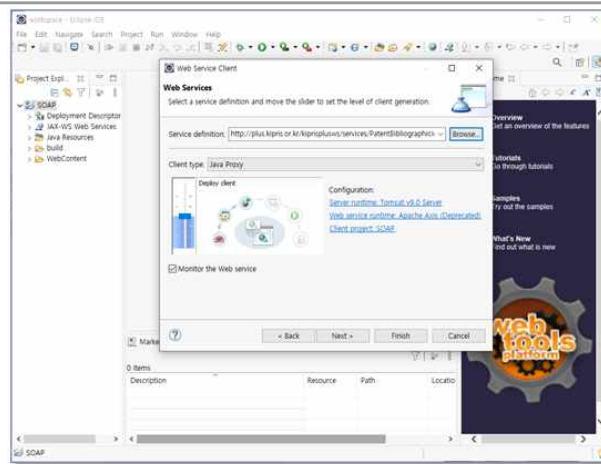
④ Web Client 생성-03

KIPRIS^{Plus} 상품화면 내 SOAP의 WSDL 확인



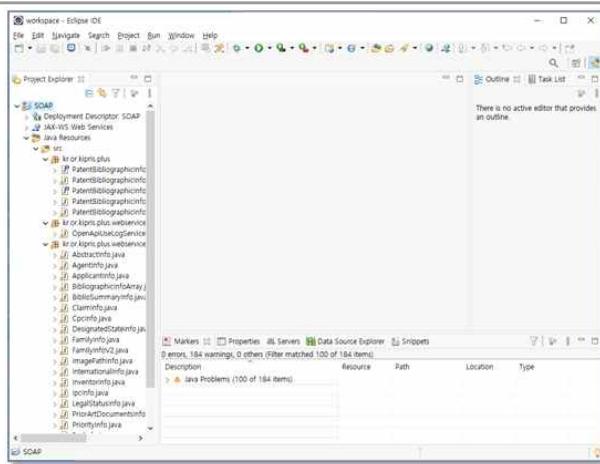
⑤ Web Client 생성-04

KIPRIS^{Plus}에서 제공하는 WSDL URL 입력

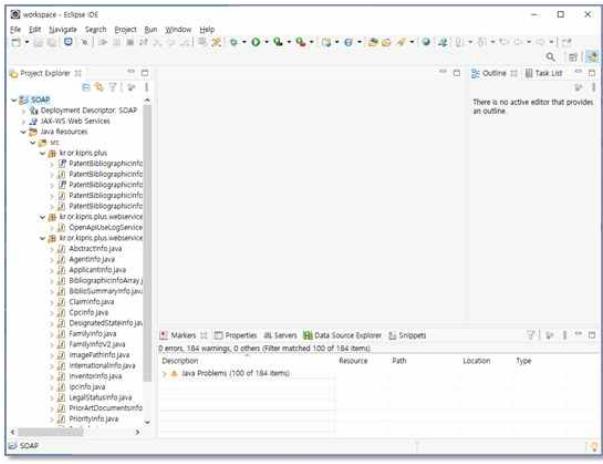
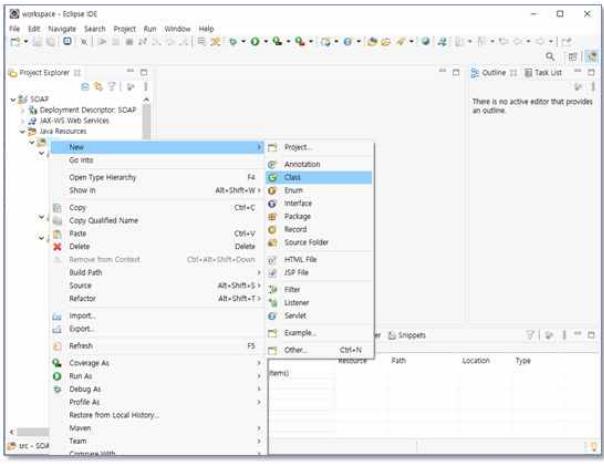
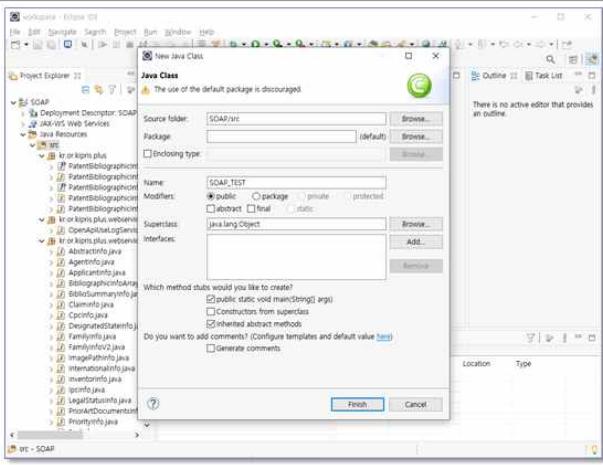
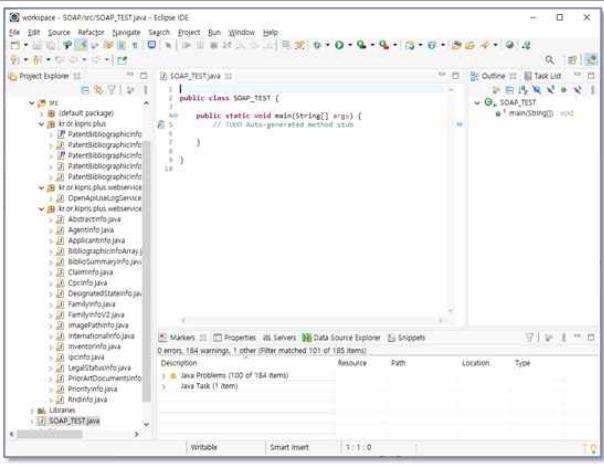


⑥ Web Client 생성-05

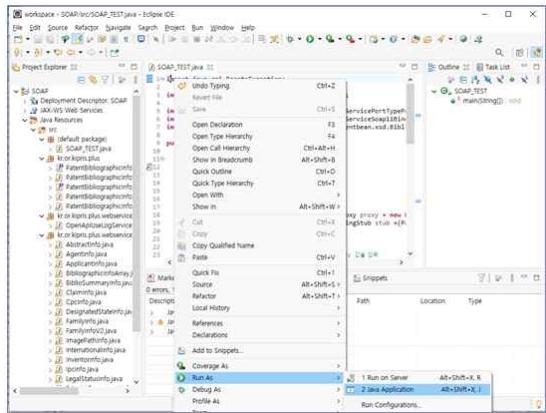
WSDL을 통해 Project 내 생성된 소스 정보 확인



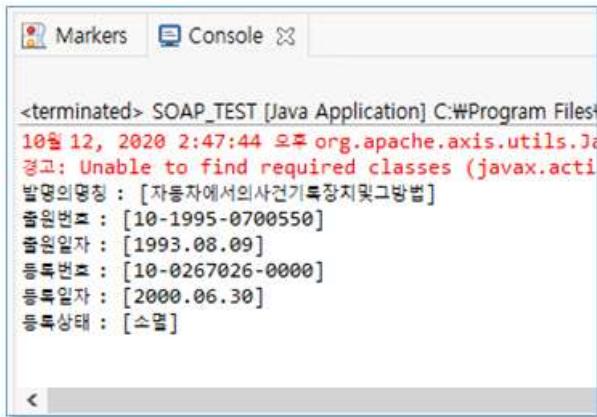
4) SOAP API 호출 프로그램 구현 및 테스트

<p>① Project 내 src 폴더 확인 src 폴더에 마우스 우 클릭</p> 	<p>② Class 생성-01 [서브 메뉴] New → Class 선택</p> 
<p>③ Class 생성-02 Name 입력, public static void main 체크</p> 	<p>④ Class 생성-03 신규 생성된 Class 정보 확인</p> 
<p>⑤ 샘플 소스 구현-01 SOAP API 호출 소스 선언(P.27 참고)</p> <pre>//wsdl을 통해 Client로 가져온 정보를 선언 PatentBibliographicInfoServicePortTypeProxy proxy = new PatentBibliographicInfoServicePortTypeProxy(); PatentBibliographicInfoServiceSoap11BindingStub stub = (PatentBibliographicInfoServiceSoap11BindingStub)proxy.getPatentBibliographicInfoServicePortType(); //KIPRISPlus에서 사용중인 본인의 ID를 입력 //KIPRISPlus에서 제공받은 본인의 SOAP AccessKey 값을 입력 String _userId = " "; String _userKey = " "; //SOAP Header에 id 정보 셋팅 SOAPHeaderElement id = new SOAPHeaderElement("http://plus.kipris.or.kr","userId"); id.setValue(_userId); stub.setHeader(id); //SOAP Header에 key 정보 셋팅 SOAPHeaderElement userKey = new SOAPHeaderElement("http://plus.kipris.or.kr","userKey"); userKey.setValue(_userKey); stub.setHeader(userKey);</pre>	<p>⑥ 샘플 소스 구현-02 SOAP API 호출 소스 선언(P.27 참고)</p> <pre>//결과(출력)값을 저장할 배열 정보 선언 BiblioSummaryInfo[] result = null; try { //입력값(출원번호)을 통해 대상 조회 result = stub.biblioSummaryInfo("1019950700550"); } catch (RemoteException e) { // TODO Auto-generated catch block e.printStackTrace(); } //입력값(출원번호)을 통해 조회한 대상의 출력값을 확인 System.out.println("발명의 명칭 : [" + result[0].getInventionTitle()+"']"); System.out.println("출원번호 : [" + result[0].getApplicationNumber()+"']"); System.out.println("출원일자 : [" + result[0].getApplicationDate()+"']"); System.out.println("등록번호 : [" + result[0].getRegisterNumber()+"']"); System.out.println("등록일자 : [" + result[0].getRegisterDate()+"']"); System.out.println("등록상태 : [" + result[0].getRegisterStatus()+"']);</pre>

⑦ SOAP API 샘플 호출 결과 확인-01
Class 마우스 우 클릭 → Run As → Java Application 선택



⑧ SOAP API 샘플 호출 결과 확인-02
하단 Console 창을 통해 API 호출 결과 확인



[참고] 샘플 소스 프로그램 구현 예시

```

1= import java.rmi.RemoteException;
2
3 import org.apache.axis.message.SOAPHeaderElement;
4
5 import kr.or.kipris.plus.PatentBibliographicInfoServicePortTypeProxy;
6 import kr.or.kipris.plus.PatentBibliographicInfoServiceSoap11BindingStub;
7 import kr.or.kipris.plus.webservice.services.patentbean.xsd.BiblioSummaryInfo;
8
9 public class SOAP_TEST {
10
11    public static void main(String[] args) {
12        // TODO Auto-generated method stub
13
14
15        /** SOAP API 기본 코드 소스 정보 */
16
17        //wsdl을 통해 Client로 가져온 정보를 선언
18        PatentBibliographicInfoServicePortTypeProxy proxy = new PatentBibliographicInfoServicePortTypeProxy();
19        PatentBibliographicInfoServiceSoap11BindingStub stub =(PatentBibliographicInfoServiceSoap11BindingStub)proxy.getPatentBibliographicInfoServicePortType();
20
21
22        //KIPRISPlus에서 사용할 브랜드 ID를 입력
23        //KIPRISPlus에서 제공받은 본인의 SOAP AccessKey 값을 입력
24        String _userId = "_____";
25        String _userKey = "_____";
26
27        //SOAP Header에 id 정보 설정
28        SOAPHeaderElement id = new SOAPHeaderElement("http://plus.kipris.or.kr","userId");
29        id.setValue(_userId);
30        stub.setHeader(id);
31
32        //SOAP Header에 key 정보 설정
33        SOAPHeaderElement userKey = new SOAPHeaderElement("http://plus.kipris.or.kr","userKey");
34        userKey.setValue(_userKey);
35        stub.setHeader(userKey);
36
37
38        //결과(출력)값을 저장할 배열 정체 선언
39        BiblioSummaryInfo[] result = null;
40
41    try {
42        //입력값(출원번호)을 통해 대상 코드
43        result = stub.biblioSummaryInfo("1019950700550");
44    } catch (RemoteException e) {
45        // TODO Auto-generated catch block
46        e.printStackTrace();
47    }
48
49
50
51        //입력값(출원번호)을 통해 대상 대상의 출력값을 확인
52        System.out.println("제명의정 : [" + result[0].getInventionTitle()+""]');
53        System.out.println("출원번호 : [" + result[0].getApplicationNumber()+"']");
54        System.out.println("출원일자 : [" + result[0].getApplicationDate()+"']");
55        System.out.println("등록번호 : [" + result[0].getRegisterNumber()+"']");
56        System.out.println("등록일자 : [" + result[0].getRegisterDate()+"']");
57        System.out.println("등록상태 : [" + result[0].getRegisterStatus()+"']");
58
59
60    }
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
89
90
91
92
93
94
95
96
97
98
99
99
100
101
102
103
104
105
106
107
108
109
109
110
111
112
113
114
115
116
117
118
119
119
120
121
122
123
124
125
126
127
128
129
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
159
160
161
162
163
164
165
166
167
168
169
169
170
171
172
173
174
175
176
177
178
179
179
180
181
182
183
184
185
186
187
188
189
189
190
191
192
193
194
195
196
197
198
199
199
200
201
202
203
204
205
206
207
208
209
209
210
211
212
213
214
215
216
217
218
219
219
220
221
222
223
224
225
226
227
227
228
229
229
230
231
232
233
234
235
236
237
237
238
239
239
240
241
242
243
244
245
246
247
248
249
249
250
251
252
253
254
255
256
257
257
258
259
259
260
261
262
263
264
265
266
267
267
268
269
269
270
271
272
273
274
275
275
276
277
277
278
279
279
280
281
282
283
284
285
285
286
287
287
288
289
289
290
291
292
293
294
295
295
296
297
297
298
299
299
300
301
302
303
304
305
305
306
307
307
308
309
309
310
311
312
313
313
314
315
315
316
317
317
318
319
319
320
321
321
322
323
323
324
325
325
326
327
327
328
329
329
330
331
331
332
333
333
334
335
335
336
337
337
338
339
339
340
341
341
342
343
343
344
345
345
346
347
347
348
349
349
350
351
351
352
353
353
354
355
355
356
357
357
358
359
359
360
361
361
362
363
363
364
365
365
366
367
367
368
369
369
370
371
371
372
373
373
374
375
375
376
377
377
378
379
379
380
381
381
382
383
383
384
385
385
386
387
387
388
389
389
390
391
391
392
393
393
394
395
395
396
397
397
398
399
399
400
401
401
402
403
403
404
405
405
406
407
407
408
409
409
410
411
411
412
413
413
414
415
415
416
417
417
418
419
419
420
421
421
422
423
423
424
425
425
426
427
427
428
429
429
430
431
431
432
433
433
434
435
435
436
437
437
438
439
439
440
441
441
442
443
443
444
445
445
446
447
447
448
449
449
450
451
451
452
453
453
454
455
455
456
457
457
458
459
459
460
461
461
462
463
463
464
465
465
466
467
467
468
469
469
470
471
471
472
473
473
474
475
475
476
477
477
478
479
479
480
481
481
482
483
483
484
485
485
486
487
487
488
489
489
490
491
491
492
493
493
494
495
495
496
497
497
498
499
499
500
501
501
502
503
503
504
505
505
506
507
507
508
509
509
510
511
511
512
513
513
514
515
515
516
517
517
518
519
519
520
521
521
522
523
523
524
525
525
526
527
527
528
529
529
530
531
531
532
533
533
534
535
535
536
537
537
538
539
539
540
541
541
542
543
543
544
545
545
546
547
547
548
549
549
550
551
551
552
553
553
554
555
555
556
557
557
558
559
559
560
561
561
562
563
563
564
565
565
566
567
567
568
569
569
570
571
571
572
573
573
574
575
575
576
577
577
578
579
579
580
581
581
582
583
583
584
585
585
586
587
587
588
589
589
590
591
591
592
593
593
594
595
595
596
597
597
598
599
599
600
601
601
602
603
603
604
605
605
606
607
607
608
609
609
610
611
611
612
613
613
614
615
615
616
617
617
618
619
619
620
621
621
622
623
623
624
625
625
626
627
627
628
629
629
630
631
631
632
633
633
634
635
635
636
637
637
638
639
639
640
641
641
642
643
643
644
645
645
646
647
647
648
649
649
650
651
651
652
653
653
654
655
655
656
657
657
658
659
659
660
661
661
662
663
663
664
665
665
666
667
667
668
669
669
670
671
671
672
673
673
674
675
675
676
677
677
678
679
679
680
681
681
682
683
683
684
685
685
686
687
687
688
689
689
690
691
691
692
693
693
694
695
695
696
697
697
698
699
699
700
701
701
702
703
703
704
705
705
706
707
707
708
709
709
710
711
711
712
713
713
714
715
715
716
717
717
718
719
719
720
721
721
722
723
723
724
725
725
726
727
727
728
729
729
730
731
731
732
733
733
734
735
735
736
737
737
738
739
739
740
741
741
742
743
743
744
745
745
746
747
747
748
749
749
750
751
751
752
753
753
754
755
755
756
757
757
758
759
759
760
761
761
762
763
763
764
765
765
766
767
767
768
769
769
770
771
771
772
773
773
774
775
775
776
777
777
778
779
779
780
781
781
782
783
783
784
785
785
786
787
787
788
789
789
790
791
791
792
793
793
794
795
795
796
797
797
798
799
799
800
801
801
802
803
803
804
805
805
806
807
807
808
809
809
810
811
811
812
813
813
814
815
815
816
817
817
818
819
819
820
821
821
822
823
823
824
825
825
826
827
827
828
829
829
830
831
831
832
833
833
834
835
835
836
837
837
838
839
839
840
841
841
842
843
843
844
845
845
846
847
847
848
849
849
850
851
851
852
853
853
854
855
855
856
857
857
858
859
859
860
861
861
862
863
863
864
865
865
866
867
867
868
869
869
870
871
871
872
873
873
874
875
875
876
877
877
878
879
879
880
881
881
882
883
883
884
885
885
886
887
887
888
889
889
890
891
891
892
893
893
894
895
895
896
897
897
898
899
899
900
901
901
902
903
903
904
905
905
906
907
907
908
909
909
910
911
911
912
913
913
914
915
915
916
917
917
918
919
919
920
921
921
922
923
923
924
925
925
926
927
927
928
929
929
930
931
931
932
933
933
934
935
935
936
937
937
938
939
939
940
941
941
942
943
943
944
945
945
946
947
947
948
949
949
950
951
951
952
953
953
954
955
955
956
957
957
958
959
959
960
961
961
962
963
963
964
965
965
966
967
967
968
969
969
970
971
971
972
973
973
974
975
975
976
977
977
978
979
979
980
981
981
982
983
983
984
985
985
986
987
987
988
989
989
990
991
991
992
993
993
994
995
995
996
997
997
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484

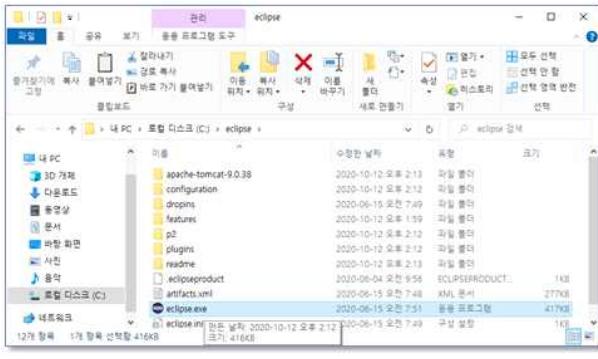
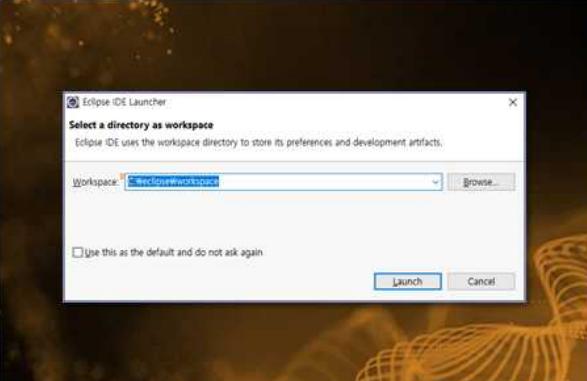
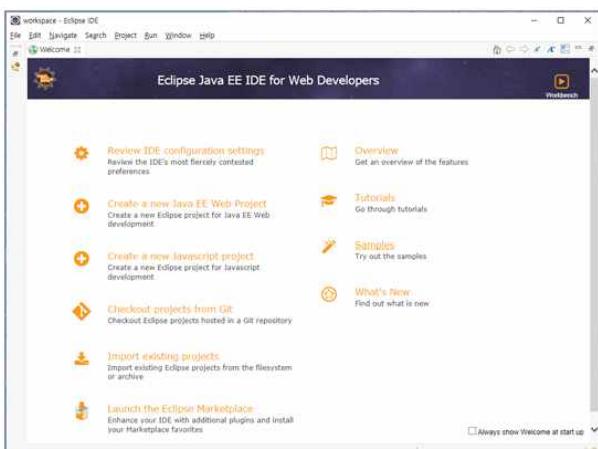
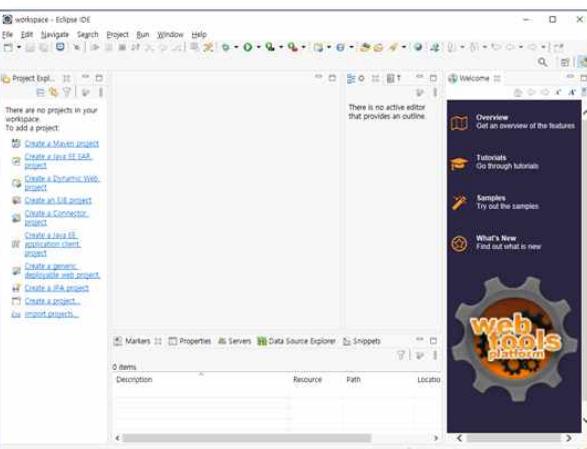
```

3 REST API 호출

3.1. Java를 이용한 REST API 호출 구현

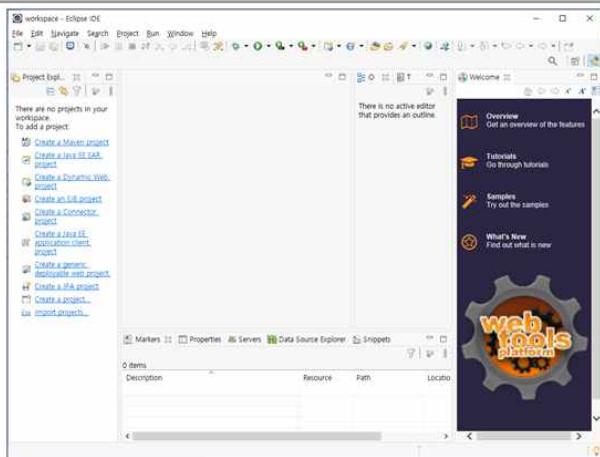
- Eclipse 실행부터 프로그램 구현까지 REST API 호출 방법에 대한 가이드 문서이며, 기본적인 프로그램 구현 방식으로 작성되었습니다.

1) Eclipse 실행

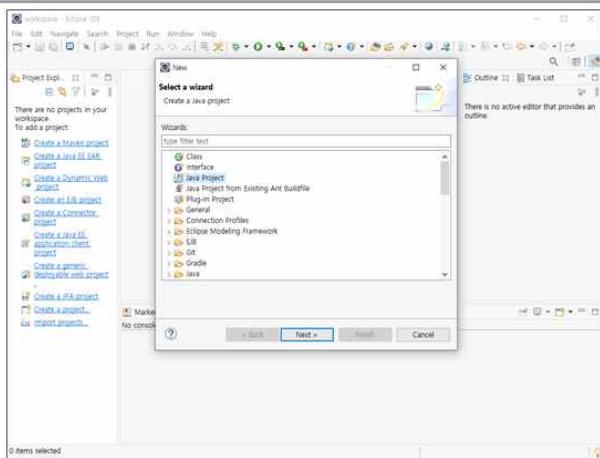
① Eclipse 실행	② 작업경로 지정(관리 편한 경로로 추천)
	
③ Eclipse 최초 실행 시 화면	④ Eclipse 초기 화면
	

2) 신규 Project 생성

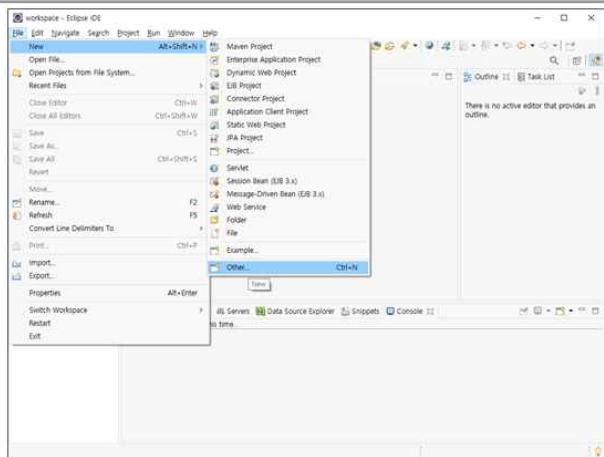
① Eclipse 초기 화면



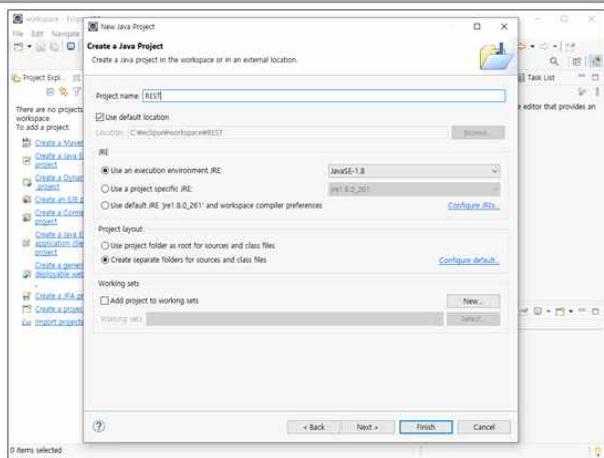
③ 신규 프로젝트 생성-02
[항목] Java Project 선택



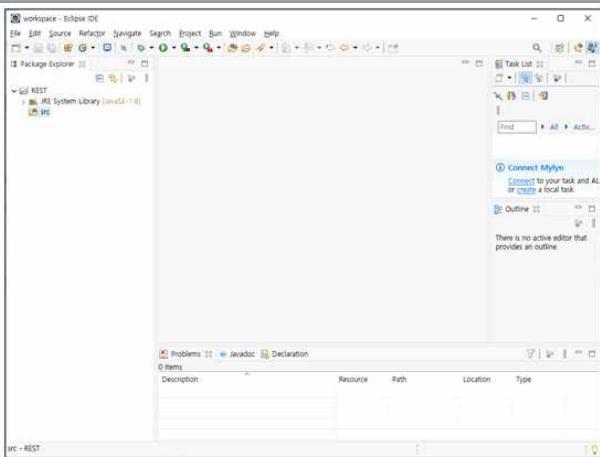
② 신규 프로젝트 생성-01
[메뉴] File → New → Other 선택



④ 신규 프로젝트 생성-03
Project 생성을 위한 기본 정보 입력 완료

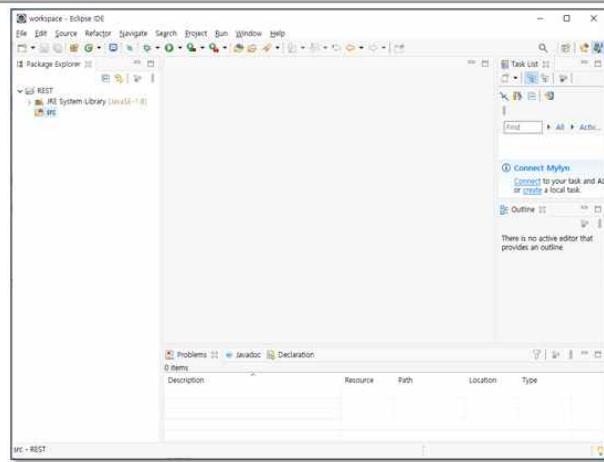


⑤ 신규 프로젝트 생성-04
생성된 신규 Project 확인



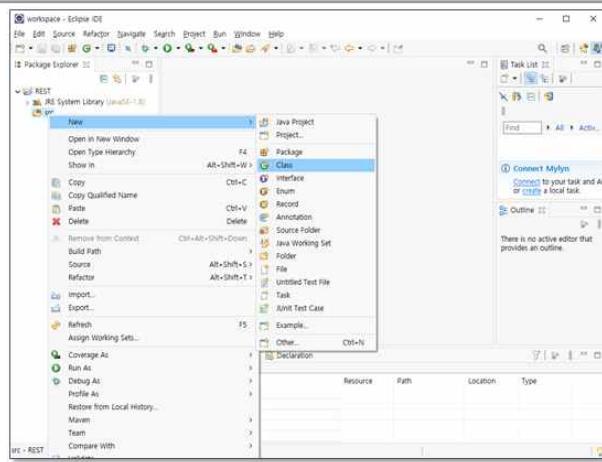
3) REST API 호출 프로그램 구현 및 테스트

- ① Project 내 src 폴더 확인
src 폴더에 마우스 우 클릭



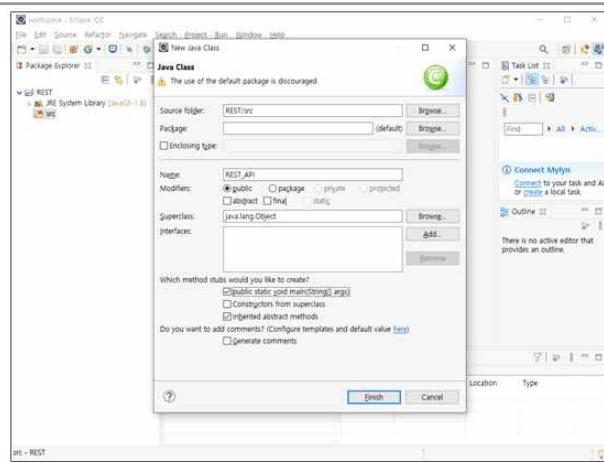
- ② Class 생성-01

[서브 메뉴] New → Class 선택



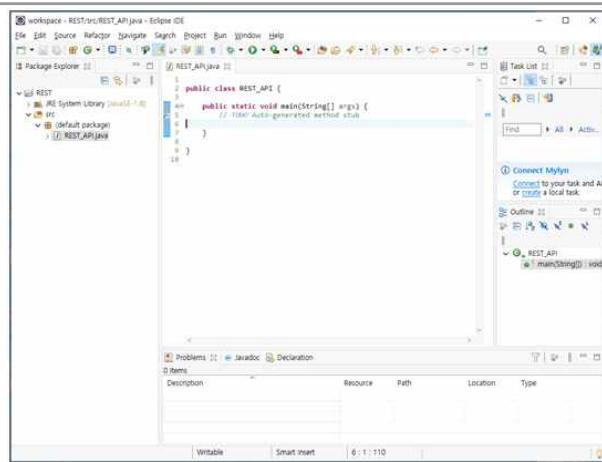
- ③ Class 생성-02

Name 입력, public static void main 체크



- ④ Class 생성-03

신규 생성된 Class 정보 확인



- ⑤ 샘플 소스 구현

REST API 호출 소스 선언(P.31~32 참고)

```

1 import java.io.BufferedReader;
2 import java.io.InputStream;
3 import java.io.InputStreamReader;
4 import java.net.HttpURLConnection;
5 import java.net.URL;
6 import java.net.URLConnection;
7 import java.util.List;
8 import javax.xml.parsers.DocumentBuilder;
9 import javax.xml.parsers.DocumentBuilderFactory;
10 import javax.xml.parsers.ParserConfigurationException;
11 import org.w3c.dom.Document;
12 import org.w3c.dom.Element;
13 import org.w3c.dom.Node;
14 import org.w3c.dom.NodeList;
15 import org.xml.sax.SAXException;
16
17 public class REST_API {
18
19     public static void main(String[] args) throws SAXException, IOException {
20         // TODO Auto-generated method stub
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
287
288
289
289
290
291
292
293
294
295
296
297
297
298
299
299
300
300
301
301
302
302
303
303
304
304
305
305
306
306
307
307
308
308
309
309
310
310
311
311
312
312
313
313
314
314
315
315
316
316
317
317
318
318
319
319
320
320
321
321
322
322
323
323
324
324
325
325
326
326
327
327
328
328
329
329
330
330
331
331
332
332
333
333
334
334
335
335
336
336
337
337
338
338
339
339
340
340
341
341
342
342
343
343
344
344
345
345
346
346
347
347
348
348
349
349
350
350
351
351
352
352
353
353
354
354
355
355
356
356
357
357
358
358
359
359
360
360
361
361
362
362
363
363
364
364
365
365
366
366
367
367
368
368
369
369
370
370
371
371
372
372
373
373
374
374
375
375
376
376
377
377
378
378
379
379
380
380
381
381
382
382
383
383
384
384
385
385
386
386
387
387
388
388
389
389
390
390
391
391
392
392
393
393
394
394
395
395
396
396
397
397
398
398
399
399
400
400
401
401
402
402
403
403
404
404
405
405
406
406
407
407
408
408
409
409
410
410
411
411
412
412
413
413
414
414
415
415
416
416
417
417
418
418
419
419
420
420
421
421
422
422
423
423
424
424
425
425
426
426
427
427
428
428
429
429
430
430
431
431
432
432
433
433
434
434
435
435
436
436
437
437
438
438
439
439
440
440
441
441
442
442
443
443
444
444
445
445
446
446
447
447
448
448
449
449
450
450
451
451
452
452
453
453
454
454
455
455
456
456
457
457
458
458
459
459
460
460
461
461
462
462
463
463
464
464
465
465
466
466
467
467
468
468
469
469
470
470
471
471
472
472
473
473
474
474
475
475
476
476
477
477
478
478
479
479
480
480
481
481
482
482
483
483
484
484
485
485
486
486
487
487
488
488
489
489
490
490
491
491
492
492
493
493
494
494
495
495
496
496
497
497
498
498
499
499
500
500
501
501
502
502
503
503
504
504
505
505
506
506
507
507
508
508
509
509
510
510
511
511
512
512
513
513
514
514
515
515
516
516
517
517
518
518
519
519
520
520
521
521
522
522
523
523
524
524
525
525
526
526
527
527
528
528
529
529
530
530
531
531
532
532
533
533
534
534
535
535
536
536
537
537
538
538
539
539
540
540
541
541
542
542
543
543
544
544
545
545
546
546
547
547
548
548
549
549
550
550
551
551
552
552
553
553
554
554
555
555
556
556
557
557
558
558
559
559
560
560
561
561
562
562
563
563
564
564
565
565
566
566
567
567
568
568
569
569
570
570
571
571
572
572
573
573
574
574
575
575
576
576
577
577
578
578
579
579
580
580
581
581
582
582
583
583
584
584
585
585
586
586
587
587
588
588
589
589
590
590
591
591
592
592
593
593
594
594
595
595
596
596
597
597
598
598
599
599
600
600
601
601
602
602
603
603
604
604
605
605
606
606
607
607
608
608
609
609
610
610
611
611
612
612
613
613
614
614
615
615
616
616
617
617
618
618
619
619
620
620
621
621
622
622
623
623
624
624
625
625
626
626
627
627
628
628
629
629
630
630
631
631
632
632
633
633
634
634
635
635
636
636
637
637
638
638
639
639
640
640
641
641
642
642
643
643
644
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
13
```

⑦ REST API 샘플 호출 결과 확인 – 02
하단 Console 창을 통해 API 호출 결과 확인

```
결과 Case 1
<?xml version="1.0" encoding="UTF-8"?>

<response>
    <header>
        <resultCode></resultCode>
        <resultMsg></resultMsg>
    </header>
    <body>
        <items>
            <patentIpcInfo>
                <InternationalpatentclassificationNumber>H04B 7/26</InternationalpatentclassificationNumber>
                <InternationalpatentclassificationDate>(2006.01.01)</InternationalpatentclassificationDate>
            </patentIpcInfo>
        </items>
    </body>
</response>

결과 Case 2
IPC : H04B 7/26
일자 : (2006.01.01)
```

[참고] 샘플 소스 프로그램 구현 예시 – 01

```
① REST_API.java 03
1 import java.io.BufferedReader;
2 import java.io.IOException;
3 import java.io.InputStream;
4 import java.io.InputStreamReader;
5 import java.net.URL;
6 import java.netURLConnection;
7
8 import javax.xml.parsers.DocumentBuilder;
9 import javax.xml.parsers.DocumentBuilderFactory;
10 import javax.xml.parsers.ParserConfigurationException;
11
12 import org.w3c.dom.Document;
13 import org.w3c.dom.Element;
14 import org.w3c.dom.Node;
15 import org.w3c.dom.NodeList;
16 import org.xml.sax.SAXException;
17
18 public class REST_API {
19
20     public static void main(String[] args) throws SAXException, IOException {
21         // TODO Auto-generated method stub
22
23
24         URL url = null;
25         URLConnection conn = null;
26
27
28         //url 설정(입력값, KEY값 포함)
29         String url1 = "http://plus.kipris.or.kr" +
30             "/openapi/rest/patentModInfoSearchService/patentIpcInfo";
31
32         String applno = "1020060116886";
33         String key = "[REDACTED]";
34
35         String url2 = "?applicationNumber=" + applno;
36         String url3 = "&accessKey=" + key;
```

[참고] 샘플 소스 프로그램 구현 예시-02

```
39  
40  
41  
42     //Case 1  
43     try {  
44  
45         InputStream is;  
46  
47         url = new URL(url1+url2+url3);  
48         conn = url.openConnection();  
49         is = conn.getInputStream();  
50         InputStreamReader isr = new InputStreamReader(is);  
51         BufferedReader br = new BufferedReader(isr);  
52  
53         System.out.println("결과 Case 1");  
54         String xml = null;  
55         while((xml = br.readLine()) != null) {  
56             System.out.println(xml);  
57         }  
58         br.close();  
59  
60     } catch (IOException e) {  
61         // TODO Auto-generated catch block  
62         e.printStackTrace();  
63     }  
64
```

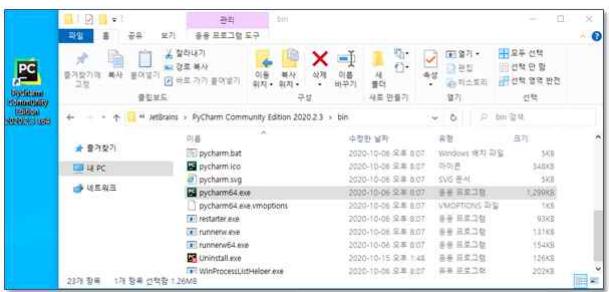
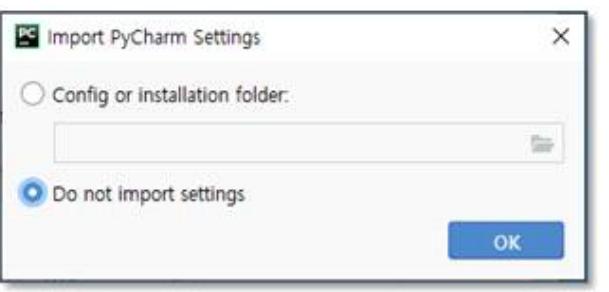
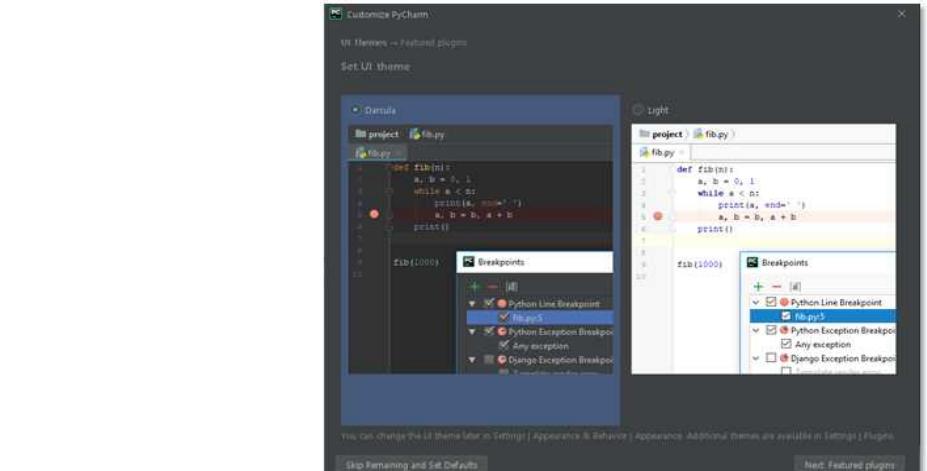
[참고] 샘플 소스 프로그램 구현 예시-03

```
74     //Case 2  
75     try {  
76  
77         DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();  
78         DocumentBuilder docb;  
79  
80         docb = dbf.newDocumentBuilder();  
81         Document doc = docb.parse(url1+url2+url3);  
82         doc.getDocumentElement().normalize();  
83  
84         NodeList nlist = doc.getElementsByTagName("patentIpcInfo");  
85  
86         for(int idx=0; idx<nList.getLength();++idx) {  
87  
88             Node n= nList.item(idx);  
89  
90             String ipc = ((Element)n).getElementsByTagName("InternationalpatentclassificationNumber").item(0).getTextContent();  
91             String date = ((Element)n).getElementsByTagName("InternationalpatentclassificationDate").item(0).getTextContent();  
92             System.out.println("\n\n결과 Case 2");  
93             System.out.println("IPC : "+ipc);  
94             System.out.println("일자 : "+date);  
95         }  
96  
97  
98     } catch (ParserConfigurationException e) {  
99         // TODO Auto-generated catch block  
100         e.printStackTrace();  
101     }  
102  
103  
104  
105  
106  
107     }  
108  
109 }
```

3.2. Python을 이용한 REST API 호출 구현

- Pycharm 실행부터 프로그램 구현까지 REST API 호출 방법에 대한 가이드 문서이며, 기본적인 프로그램 구현 방식으로 작성되었습니다.

1) Pycharm Community 실행

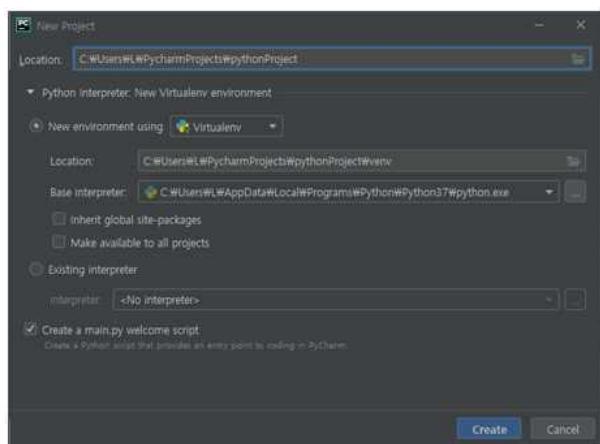
① Pycharm Community 실행 	② 최초 실행 시 확인-1 
③ 최초 실행 시 확인-2 	④ 최초 실행 시 확인-3 
⑤ 최초 실행 시 테마 확인 	

2) 신규 Project 생성

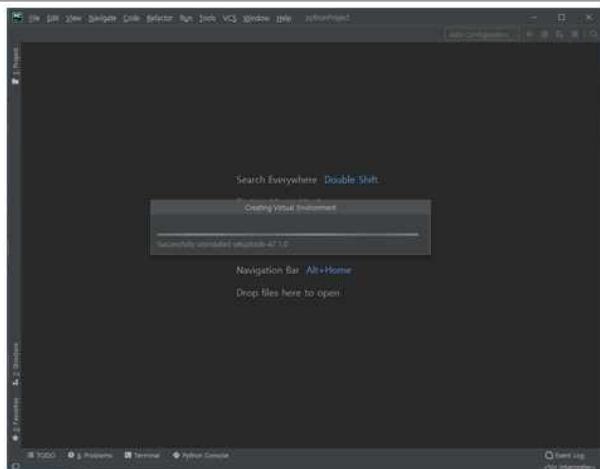
⑦ 최초 실행 시 프로젝트 생성 및 설정



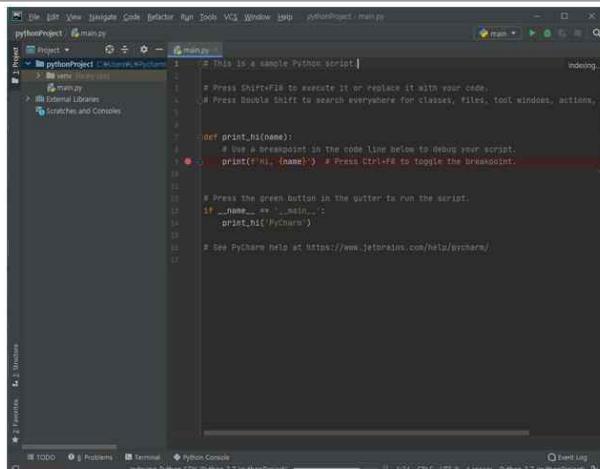
⑧ 신규 프로젝트 생성-01



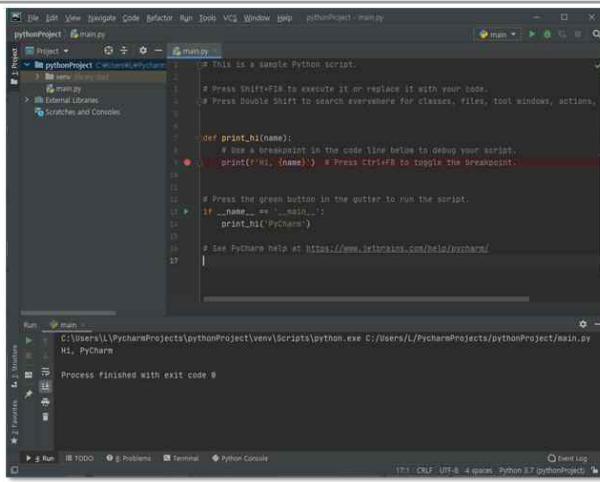
③ 신규 프로젝트 생성-02
프로젝트 생성 중



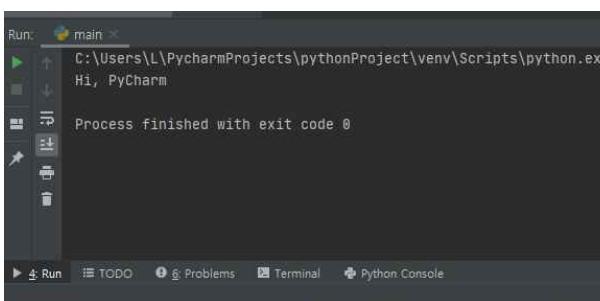
④ 신규 프로젝트 생성-03
기본 구현된 프로그램 확인



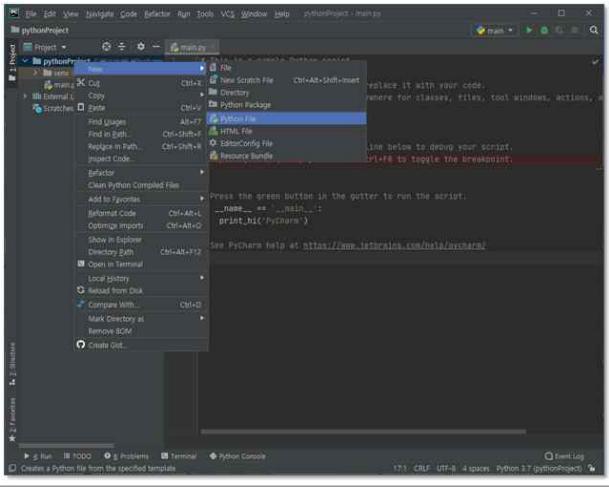
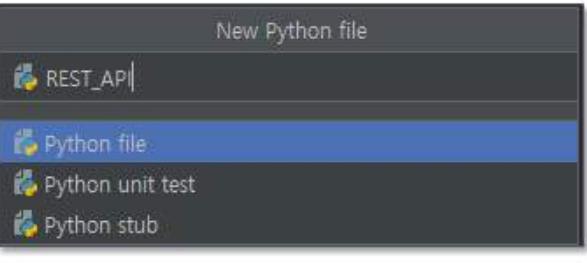
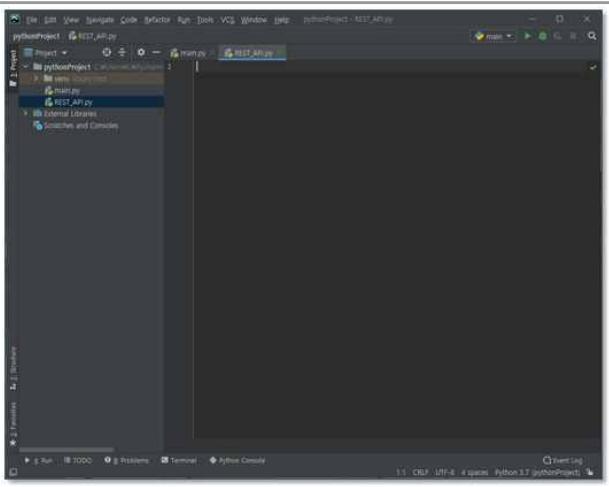
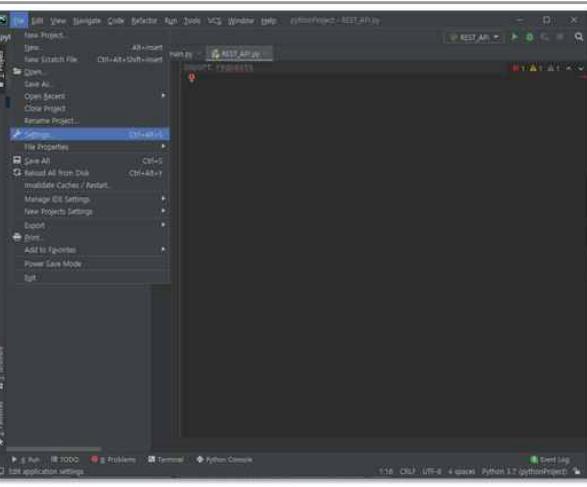
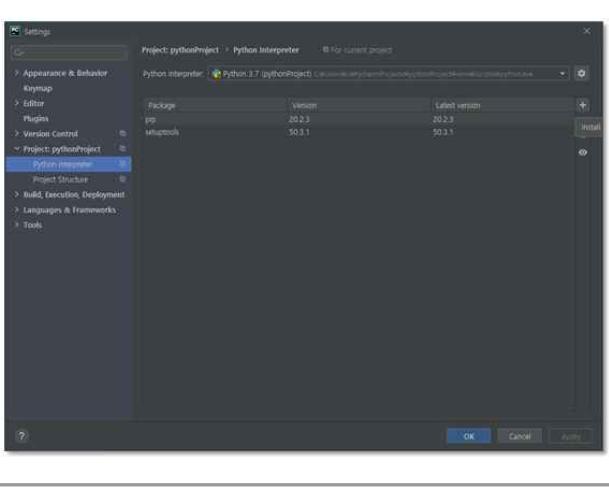
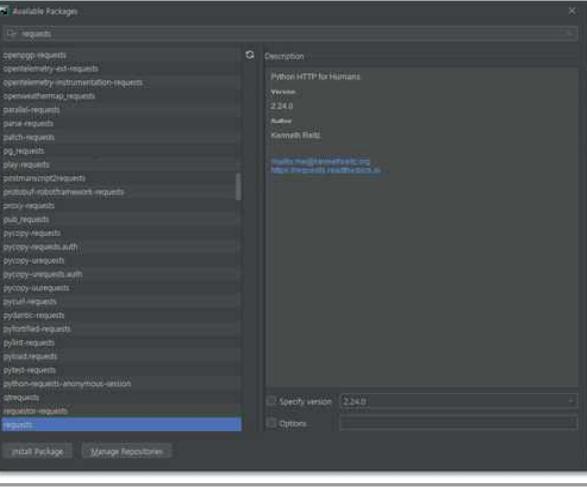
⑤ 신규 프로젝트 생성-04
프로그램 실행(Ctrl+Shift+F10, ▶버튼)



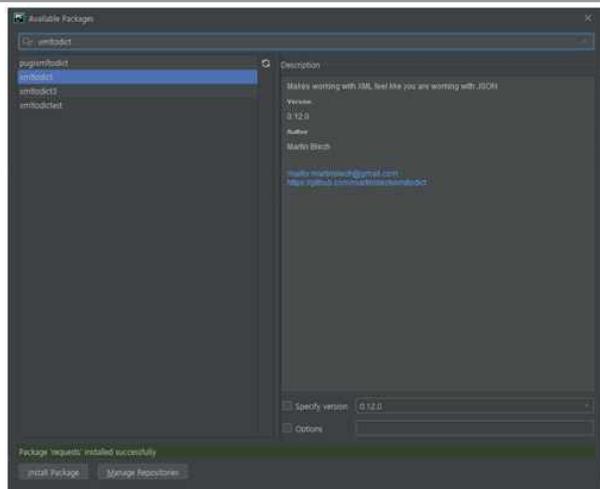
⑥ 신규 프로젝트 생성-05
하단 Window창을 통해 실행 결과 확인



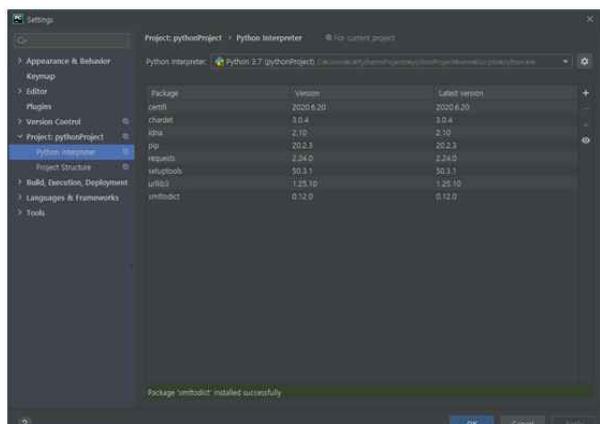
3) REST API 호출 프로그램 구현 및 테스트

<p>① Python File 생성-01 프로젝트 마우스 우 클릭 → New → Python File 선택</p> 	<p>② Python File 생성-02 파일명 입력 후 신규 .py 파일 생성</p> 
<p>③ Python File 생성-02 생성된 파일 확인</p> 	<p>④ 신규 Pakage 설치 – 01 [메뉴] File → Setting 선택</p> 
<p>⑤ 신규 Pakage 설치 – 02 Project python Interpreter 화면의 + 선택</p> 	<p>⑥ 신규 Pakage 설치 – 03 requests 검색 후 하단 install 진행</p> 

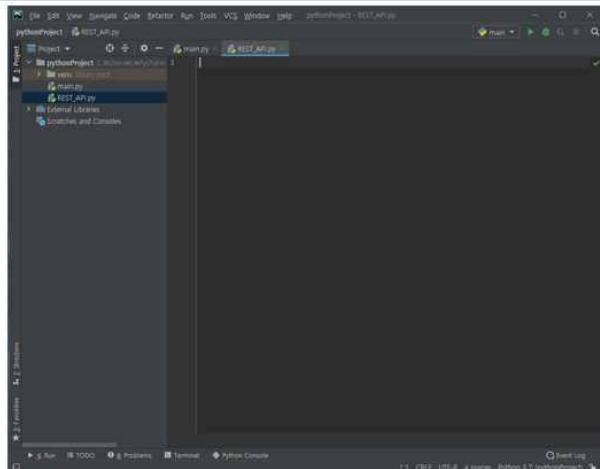
⑦ 신규 Pakage 설치 – 04
xmltodict 검색 후 하단 install 진행



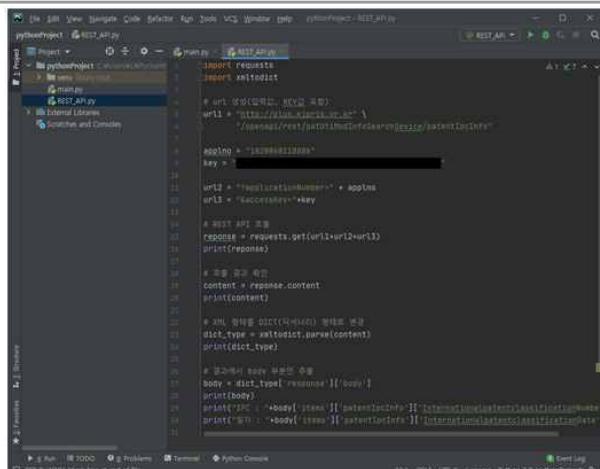
⑧ 신규 Pakage 설치 – 05
install 완료 정보 확인



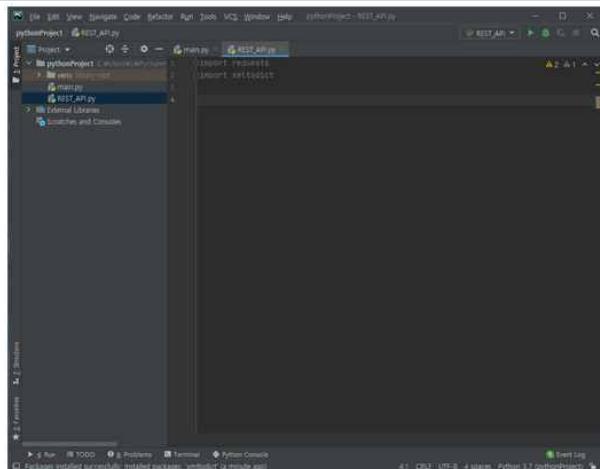
⑨ 샘플 소스 구현–01
신규 생성한 파일 확인



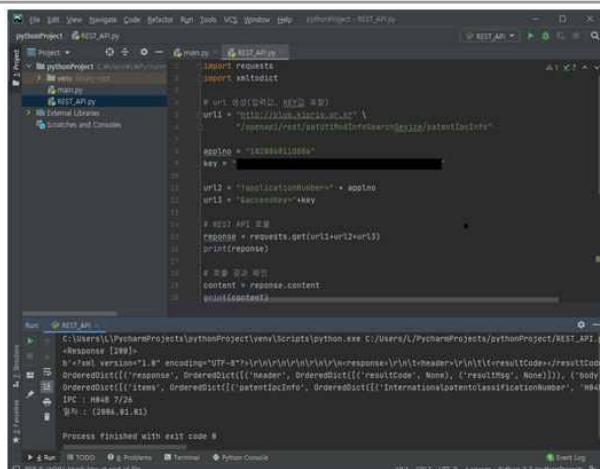
⑪ 샘플 소스 구현–03
REST API 호출 소스 선언(P.37 참고)



⑩ 샘플 소스 구현–02
import 시 앞 단계 requests, xmltodict 설치 필요



⑫ REST API 샘플 호출 결과 확인–01
프로그램 실행(Ctrl+Shift+F10, ▶ 버튼)



⑬ REST API 샘플 호출 결과 확인 – 02
하단 Console 창을 통해 API 호출 결과 확인



[참고] 샘플 소스 프로그램 구현 예시

```

main.py x REST_API.py x
1 import requests
2 import xmltodict
3
4 # url 생성(입력값, KEY값 포함)
5 url1 = "http://plus.kipris.or.kr" \
6         "/openapi/rest/patentInfoSearchService/patentIpcInfo"
7
8 applno = "1020060118886"
9 key = "REDACTED"
10
11 url2 = "?applicationNumber=" + applno
12 url3 = "&accessKey=" + key
13
14 # REST API 호출
15 reponse = requests.get(url1+url2+url3)
16 print(reponse)
17
18 # 호출 결과 확인
19 content = reponse.content
20 print(content)
21
22 # XML 형태를 DICT(딕셔너리) 형태로 변경
23 dict_type = xmltodict.parse(content)
24 print(dict_type)
25
26 # 결과에서 body 부분만 추출
27 body = dict_type['response']['body']
28 print(body)
29 print("IPC : " + body['items'][0]['patentIpcInfo']['InternationalPatentClassificationNumber'])
30 print("일자 : " + body['items'][0]['patentIpcInfo']['InternationalPatentClassificationDate'])
    
```

III

FAQ

자주 질문하는 문의 및 답변 안내

 API 호출 문의

질문	API 이용 시 접근 오류 발생 문의
답변	<p>외부 인터넷 접속이 불가능한 환경에서는 API 서비스 이용이 불가능합니다. 인터넷 접속이 가능한지 확인 부탁드립니다.</p> <p>추가로 KIPRIS^{Plus} 웹사이트(http://plus.kipris.or.kr) 접근이 가능한지 확인 부탁드립니다. 이용자의 컴퓨터 또는 네트워크 환경에서 보안상의 사유로 특정 사이트에 대한 접근 제한이 있는지 확인 후 제한 해제 부탁드립니다.</p> <p>※ 참고로 비정상적으로 API 호출 시 시스템 공격으로 판단되어 이용에 제한이 발생하며, 이를 해제하기 위해서는 소명 등 확인이 필요합니다.</p>

질문	API 응답시간 문의
답변	<p>KIPRIS^{Plus} 웹사이트 내 시간별로 API 응답시간을 안내하는 API 상태화면(http://plus.kipris.or.kr/portal/main/apiStatus.do?menuNo=210157)을 제공하고 있으므로 해당 화면을 확인 부탁드립니다.</p> <p>참고로 API 응답시간의 경우 네트워크, 서버 자원 등 여러 요인이 존재합니다. 관련하여 내부적인 요인도 확인 부탁드립니다.</p>

질문	초당 API 호출건수 제한 문의
답변	<p>서버 및 네트워크 부하 해소를 위하여 회원계정별 초당 50건 이상 호출 발생 시 서비스 이용이 제한됩니다.</p> <p>서버 및 네트워크 통신 상황을 고려하여 초당 40건 정도로 호출 부탁드립니다.</p>

질문	무료 이용 시 API 호출 가능 건수 문의
답변	<p>무료 이용자는 월 1천 건 이하로 API 서비스 이용 가능합니다.</p> <p>상품별로 API 호출 시 전체 합산수치가 월 1천 건이 넘게 되면 서비스 이용에 제한이 발생되며 매월 1일 초기화 되어 다시 서비스 이용이 가능합니다.</p>

□ API를 이용한 데이터 수집 문의

질문	데이터 수집 방법 문의
답변	<p>특허 · 실용 · 등록공보 상품에 대한 데이터 수집에 대한 예시입니다.</p> <p>Case 1. 변동정보 기능을 이용한 데이터 수집</p> <ul style="list-style-type: none"> ① 변동정보2 기능을 이용하여 날짜별 출원번호 정보 확인 ② 변동정보2 결과 내 출원번호를 이용하여 필요한 API 기능 재 호출 <p>Case 2. 전체검색 기능을 이용한 데이터 수집</p> <ul style="list-style-type: none"> ① 전체검색 기능을 이용하여 출원번호 및 관련 정보 확인 ② 전체검색을 통해 제공받은 정보보다 더 필요한 정보가 존재할 경우 전체검색 결과 내 출원번호를 이용하여 필요한 API 기능 재 호출 <p>Case 3. 자유검색 기능을 이용한 데이터 수집</p> <ul style="list-style-type: none"> ① 자유검색 기능을 이용하여 출원번호 정보 확인 ② 자유검색을 통해 제공받은 정보보다 더 필요한 정보가 존재할 경우 자유검색 결과 내 출원번호를 이용하여 필요한 API 기능 재 호출 <p>변동정보2, 자유검색, 전체검색 등의 기능으로 출원번호를 제공받을 수 있으며, 해당 출원번호를 통해 다른 API를 호출할 경우 필요한 정보를 확인 가능합니다.</p>

□ 데이터 문의

질문	누락 및 오류 데이터 확인 문의
답변	<p>특허청에서 생성된 DB데이터 및 파일데이터(PDF, XML 및 이미지 등)를 매일 제공받아 서비스하고 있습니다. 내부 데이터 검증 및 외부 데이터 문의 요청으로 데이터 누락 및 오류가 확인 될 경우 해당 부분에 대한 로직 또는 데이터를 보완하여 재 서비스 하고 있습니다. 일반적으로 데이터는 특허정보검색서비스 (www.kipris.or.kr)와 비교하여 확인이 가능합니다. 문제되는 데이터 중 일부는 대전 특허청의 원천데이터에 대한 확인 및 검증이 필요하며, 관련 부서에 확인 요청 후 데이터 확인 및 보완 완료까지 시간이 소요될 수도 있습니다.</p>

□ 프로그램 개발 문의

질문	개발 언어별(Java, C#, C, Python 등) 프로그램 구현 방법 문의
답변	<p>Open API 이용 시 단순 호출 오류에 대해서만 가이드가 가능하며 Java 외 언어에 대해서는 확인이 어렵습니다. 이용자가 개별적으로 구현한 프로그램에 대한 소스 수정 및 보완 등의 개발 공수가 필요한 경우에 대해서는 가이드를 제공해드리기 어려운 점이 존재하므로 이용에 참고 부탁드리며 인터넷에 공개된 게시글, 동영상 및 교육 자료 등을 통해 프로그램 언어에 대해 숙지하신 후 이용하시는 것을 추천 드립니다.</p>

KIPRIS^{Plus}

Open API 이용 가이드



특허청



한국특허정보원

서울특별시 강남구 테헤란로 131(역삼동 647-9) 한국지식재산센터 한국특허정보원
Tel. 02-6915-1432(1495) Fax. 02-6915-1350 Website. <http://plus.kipris.or.kr>