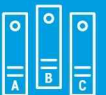


특허정보활용서비스 KIPRIS^{Plus}

Open API 이용 가이드

2020. 10



문 서 명		특허정보활용서비스(KIPRIS ^{Plus}) Open API 이용 가이드	
버 전	변 경 일	변 경 내 용	비 고
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

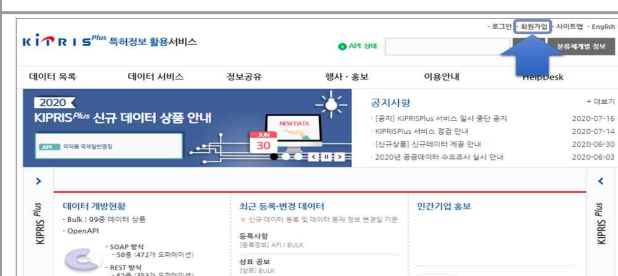
1 회원 가입 및 로그인

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 동의 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

① 오른쪽 상단의 검색 기능 이용	② 검색된 상품 확인
	

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

① 데이터 목록 탭의 서비스 유형별 선택	② 상품 접근 및 확인
	

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

① 데이터 목록 탭의 분류 체계별 선택	② 상품 접근 및 확인
	

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

① 데이터 목록 탭의 무료 상품정보 선택

② 상품 접근 및 확인

□ 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)

① 상품설명-01

② 상품설명-02

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.

서비스 유형

1 API종류 [REST] 선택

2 API분류 [서지정보] 선택

3 IPC정보 3 API 기능 선택

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

오퍼레이션 명 (Operation Name)

IPC정보 - patentIpcInfo

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/patUtilModInfoSearchService/patentIpcInfo?applicationNumber=1020060118886&access

출력값 (Response Parameter)

Variable (변수명)	Level	Variable Type (변수 타입)	Description (설명)
patentIpcInfo	1		
internationalpatentclassificationNumber	2	String	IPC번호
internationalpatentclassificationDate	2	String	IPC일자

출력값(샘플)

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



② 서비스 수수료안내 화면에서 수수료 정보 확인

Home > 이용안내 > 서비스 수수료안내

서비스 수수료안내

다운로드

KIPRIS^{Plus} 서비스 이용수수료 기준일: 2020년 6월 30일 (원단위, VAT 포함)

☐ 아래에 해당하는 경우 할인율을 추가로 적용합니다.
 가. 신청인이 개인, 중소기업, 대학, 공공기관 또는 비영리 법인 단체인 경우 **50%할인**
 나. 2개 이내 Open API 상품을 이용하는 경우 **50%할인**

☐ 특정 저장매체(CD, 하드디스크 등)를 통해 데이터를 제공받는 경우 저장매체 비용 및 배송비는 이용자 부담입니다.

☐ ㉓ 표기 : Bulk Data 방식으로만 제공되는 상품(37종)

☐ 서비스 이용 수수료에 관한 궁금하신 사항은 고객센터(☎ 02-6915-1495, ✉ hyekyung@kipi.or.kr)로 문의하여 주시기 바랍니다.

01 한국 산업재산권	데이터 상품명	Bulk Data		Open API
		당해연도	과거분 전체	
특허 실용	01 특허 실용 공개공보	3,273,463	18,004,044	API 이용요금 年 / 1,942,123원 ※ 월 1,000건 이내 무료 조회 가능
	02 특허 실용 등록공보	3,273,463	18,004,044	
	03 특허 실용 통합이력	682,516	3,753,836	
	04 특허 실용 인용문헌	84,304	463,674	
	05 특허 실용 피인용문헌	84,304	463,674	
	06 특허 실용 분류코드 이력	84,304	463,674	
	07 특허 파일리	84,304	463,674	
	08 청구항 변동이력	682,516	3,753,836	
	09 연속기간 연장등록 공보	84,304	463,674	
	10 디자인 공보	682,516	3,753,836	
	11 디자인 통합이력	682,516	3,753,836	
	12 디자인업 형태검색 DB	무료		
	13 디자인 분류코드 이력	84,304	463,674	
디자인				

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)

① 상품 화면 내 서비스 신청 선택	② 정보 입력 후 서비스 이용 신청

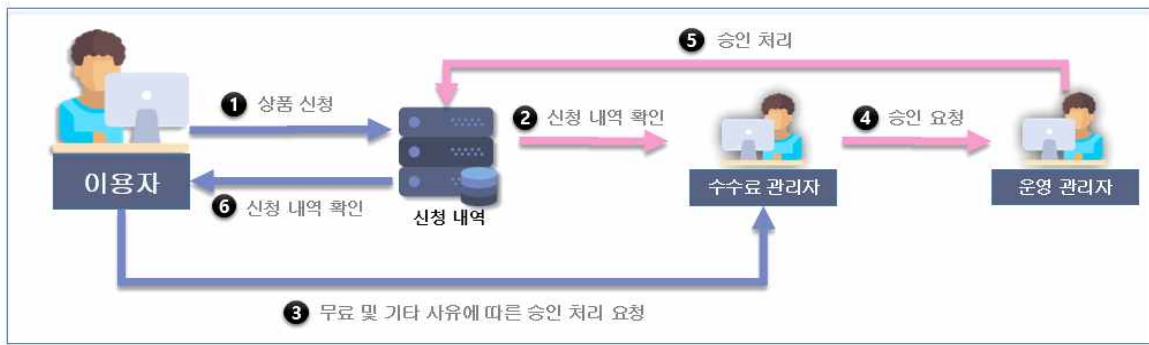
2) 서비스 일괄 신청(동시에 여러 개의 상품을 신청)

① 상품 화면 내 서비스 일괄 신청 선택	② 정보 입력 후 서비스 일괄 이용 신청

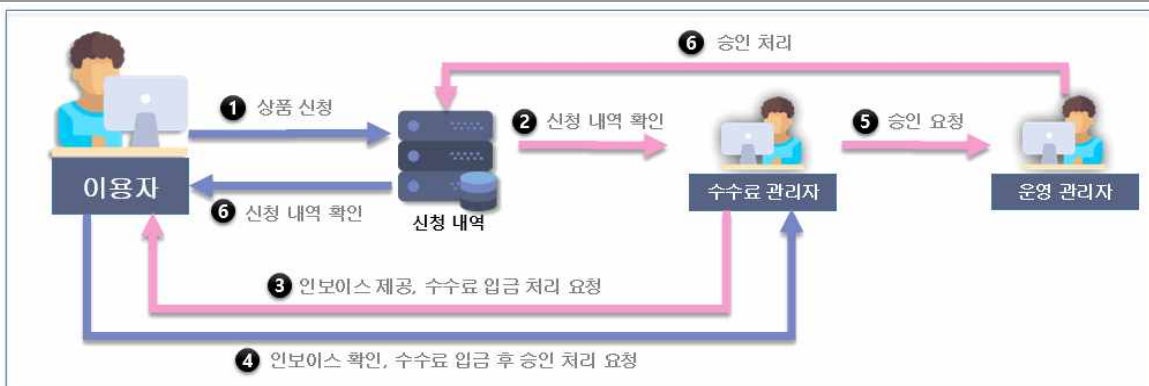
3.2. 상품 이용 승인

□ 결제 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

마이페이지 > 서비스 구매내역

서비스 구매내역

전체 SOAP REST BULK

최근 신청일 순 | 등록일 순

검색

번호	유형	서비스명	구분	신청일	시작일	종료일	승인여부
18	BULK	인보이스 자동발행명령 → 관리자 승인 불가	무료	2020-05-29	2020-01-01	2020-12-31	승인
17	REST	상표 출원 속보	무료	2019-11-29	2019-11-29	2019-12-31	승인
16	REST	특허 실용 공개 등록공보	무료	2019-11-29	2019-11-29	2019-12-31	승인
15	SOAP	디자인 공보	무료	2019-11-29	2019-11-29	2019-12-31	승인
14	SOAP	특허 실용 공개 등록공보	무료	2019-11-29	2019-11-29	2019-12-31	승인

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

① 웹사이트 오른쪽 상단의 마이페이지 선택



② API KEY 관리 확인



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.

① 웹사이트 오른쪽 상단의 마이페이지 선택



② 서비스 구매내역 확인



번호	유형	서비스명	구분	신청일	시작일	종료일	승인여부
18	BULK	의약품 국제일반명칭 ▶ 관리자 공지 보기	무료	2020-05-29	2020-01-01	2020-12-31	반려
17	REST	상표 출원 속보	무료	2019-11-29	2019-11-29	2019-12-31	승인
16	REST	특허-실용 공개-등록공보	무료	2019-11-29	2019-11-29	2019-12-31	승인

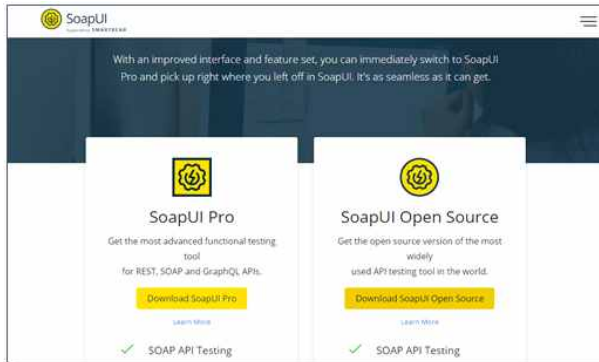
2) SOAP 방식의 API 호출 테스트를 통한 인증키 정상여부 확인

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

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

① SoapUI Open Souce 다운로드 및 설치

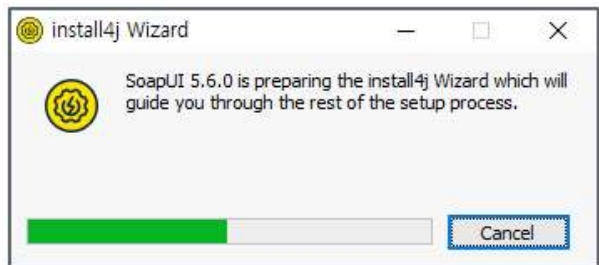
① SoapUI 다운로드 사이트 접근



② SoapUI 설치 프로그램 실행



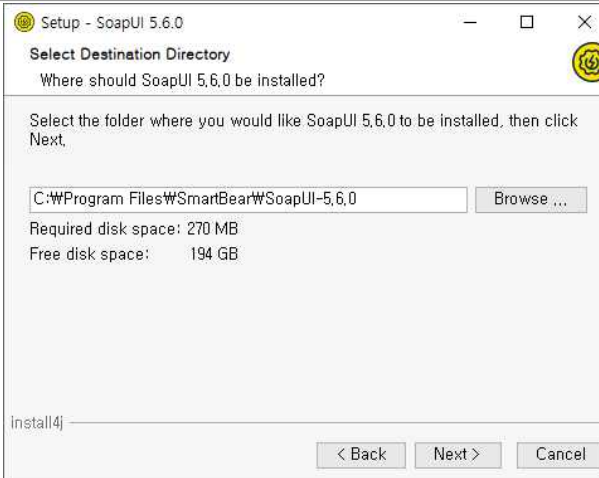
③ SoapUI 설치-01



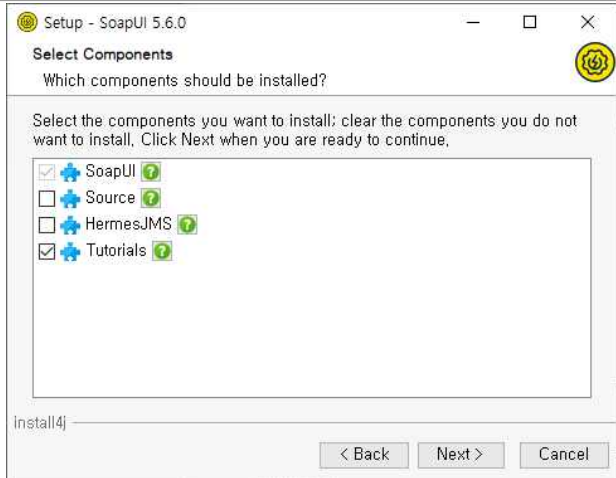
④ SoapUI 설치-02



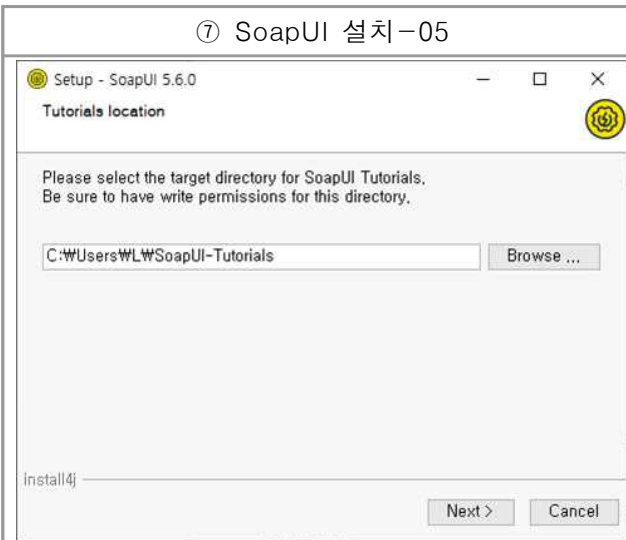
⑤ SoapUI 설치-03



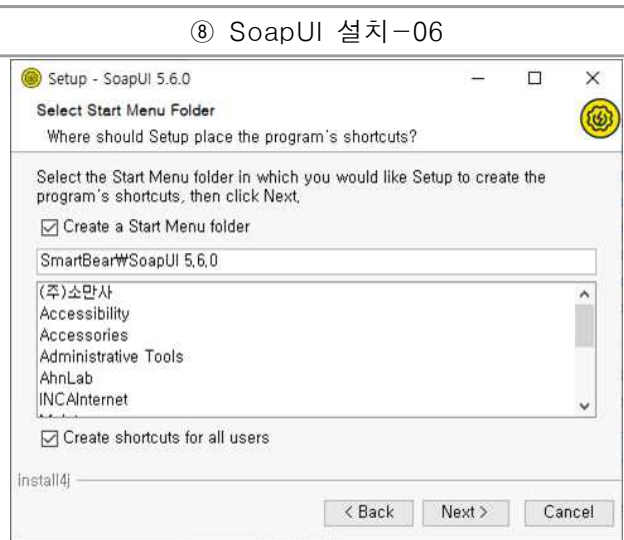
⑥ SoapUI 설치-04



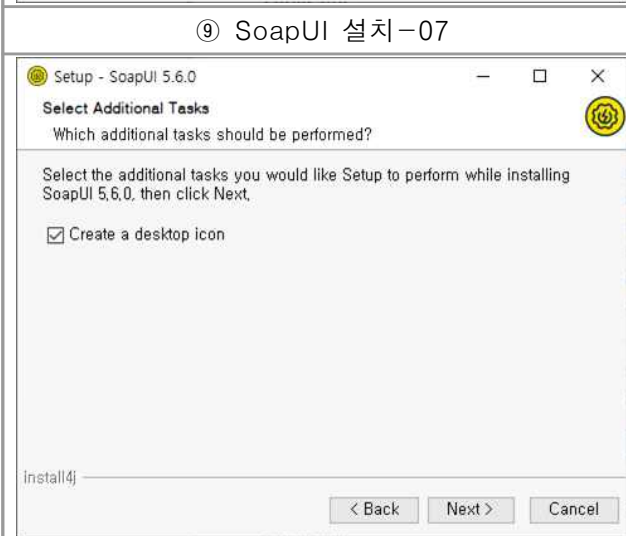
⑦ SoapUI 설치-05



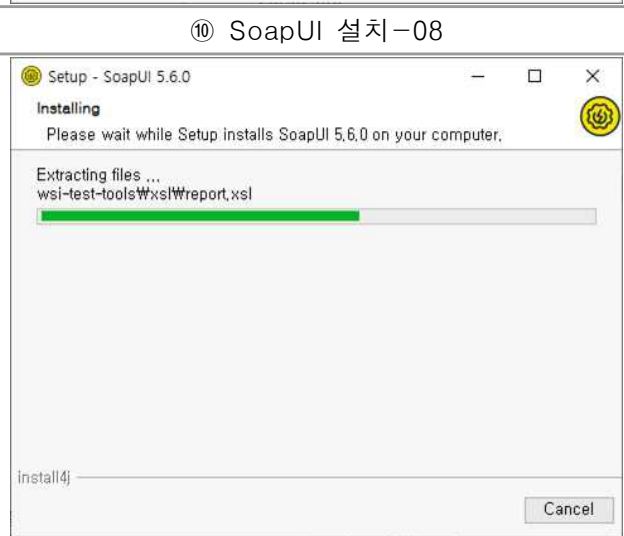
⑧ SoapUI 설치-06



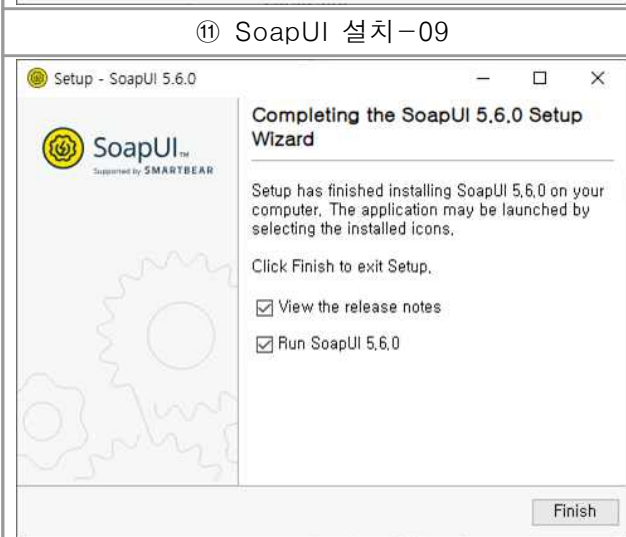
⑨ SoapUI 설치-07



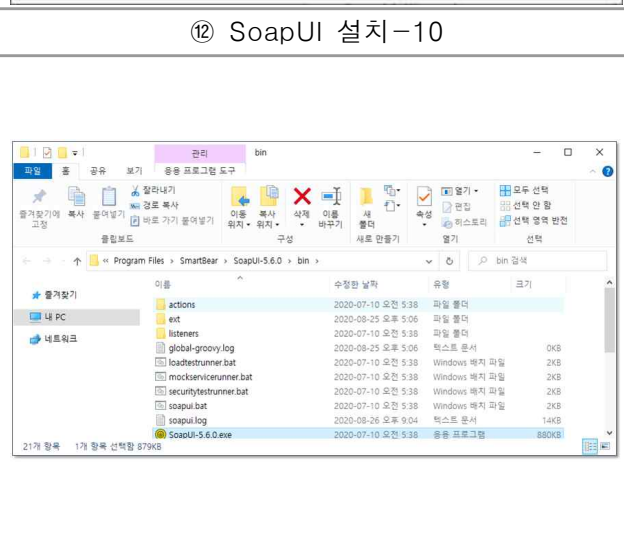
⑩ SoapUI 설치-08



⑪ SoapUI 설치-09



⑫ SoapUI 설치-10



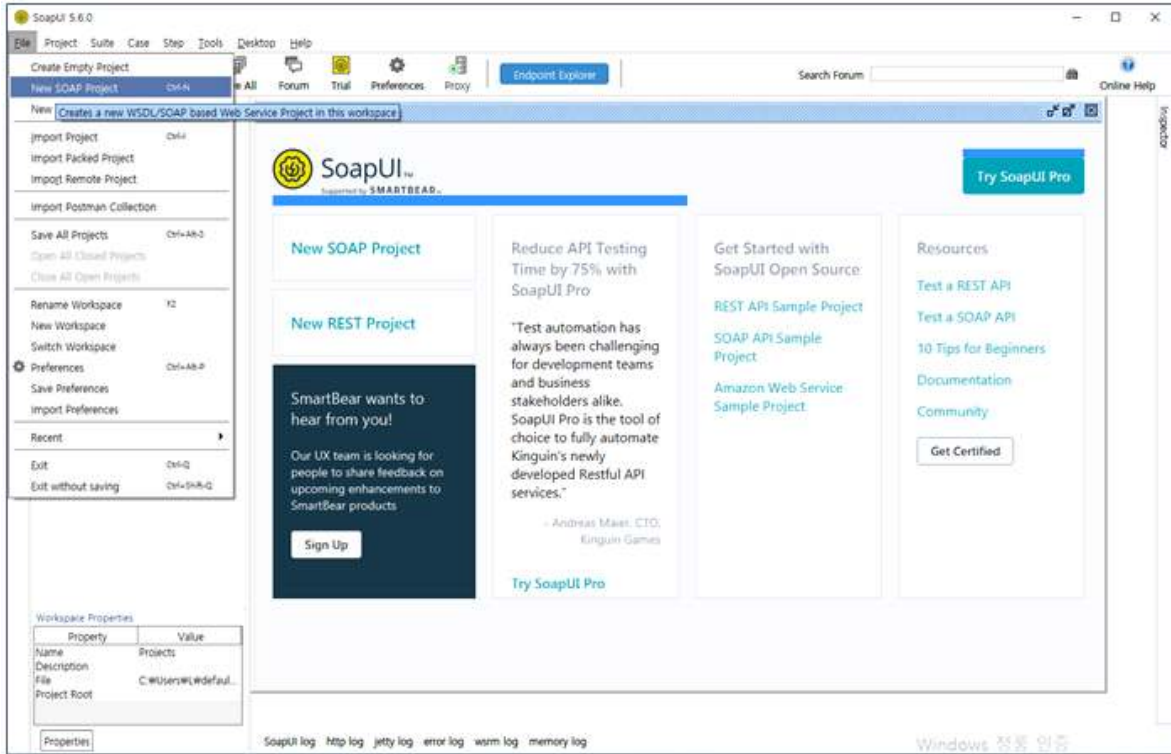
1.

서비스 이용 신청 단계

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 [Browse...]

Create Requests: ☒ Create sample requests for all operations?

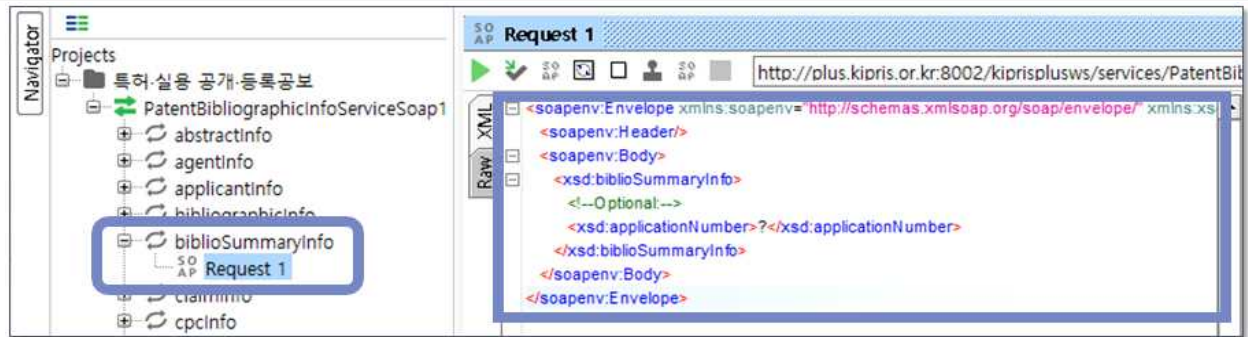
Create TestSuite: ☐ Creates a TestSuite for the imported WSDL

Relative Paths: ☐ Stores all file paths in project relatively to project file (requires save)

[OK] [Cancel]

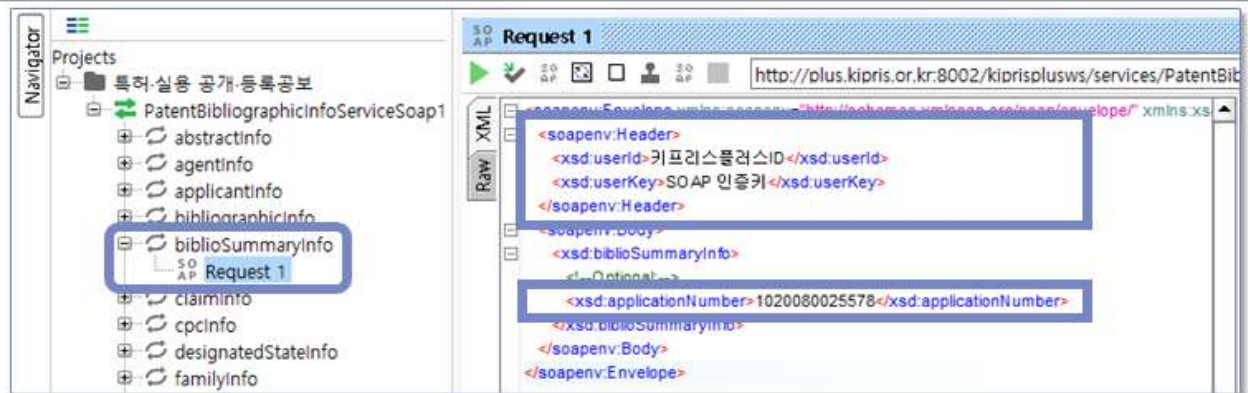
④ SOAP API 기능 선택

WSDL 등록 후 생성된 API 기능의 Request 정보 확인



⑤ SOAP API 호출

인증정보(아이디, 인증키) 및 기능별 입력정보(출원번호 등)를 작성

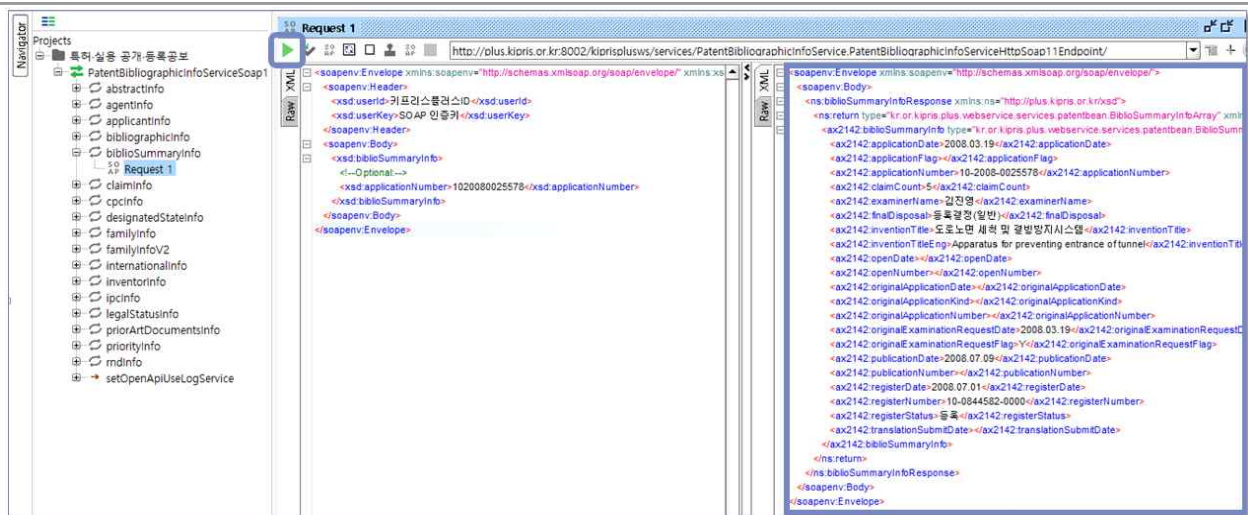


[참고] SOAP API 호출 시 Header 내 인증정보(아이디, 인증키) 작성 예시

```
<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



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/patUtiModInfoSearchSevice/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/>
    <responseTime>2020-08-26 18:02:38.238</responseTime>
    <responseMsgID/>
    <successYN>Y</successYN>
    <resultCode>00</resultCode>
    <resultMsg>NORMAL SERVICE.</resultMsg>
  </header>
  <body>
    <items>
      <item>
        <applicationDate>2005.06.10</applicationDate>
        <applicationNumber>10-2005-0050026</applicationNumber>
        <claimCount>4</claimCount>
        <finalDisposal>종록결정(일반)</finalDisposal>
        <inventionTitle>휴대용 단말기 및 그의 음향분석방법</inventionTitle>
        <inventionTitleEng>MOBILE TERMINAL AND METHOD FOR ANALYZING SOUND THEREOF</inventionTitleEng>
        <openDate>2006.12.14</openDate>
        <openNumber>10-2006-0128519</openNumber>
        <originalApplicationDate>
        <originalApplicationKind>
        <originalApplicationNumber>
        <originalExaminationRequestDate>2010.04.20</originalExaminationRequestDate>
        <originalExaminationRequestFlag>Y</originalExaminationRequestFlag>
        <publicationDate>2012.01.06</publicationDate>
        <publicationNumber>
        <registerDate>2012.01.02</registerDate>
        <registerNumber>10-1104063-0000</registerNumber>
        <registerStatus>소멸</registerStatus>
        <translationSubitDate>
      </item>
    </items>
  </body>
  <count>
    <numOfRows>1</numOfRows>
    <pageNo>1</pageNo>
    <totalCount>1</totalCount>
  </count>
</response>

```

[참고] REST API 호출 URL 작성 예시

http://plus.kipris.or.kr/kipo-api/kipi/patUtiModInfoSearchSevice/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

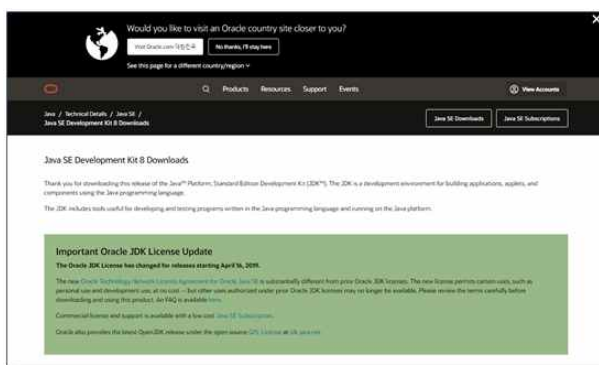
※ SOAP API 개발 환경 구축은 Java 8 이하 버전을 추천 드립니다.

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/javase-jdk8-downloads.html>)

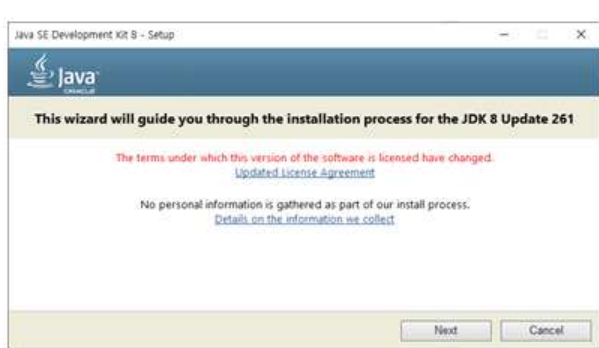
① Java 다운로드 사이트 접근



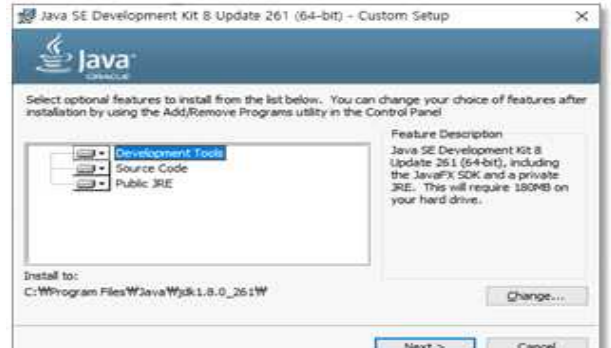
② 운영체제 호환 설치 파일 다운로드

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	75.4 MB	jdk-8u261-linux-arm32-hf.tar.gz
Linux ARM 64 Hard Float ABI	70.3 MB	jdk-8u261-linux-arm64-hf.tar.gz
Linux x86 RPM Package	121.92 MB	jdk-8u261-linux-i586.rpm
Linux x86 Compressed Archive	136.81 MB	jdk-8u261-linux-i586.tar.gz
Linux x64 RPM Package	121.53 MB	jdk-8u261-linux-x64.rpm
Linux x64 Compressed Archive	136.48 MB	jdk-8u261-linux-x64.tar.gz
macOS x64	205.94 MB	jdk-8u261-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	125.77 MB	jdk-8u261-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	88.72 MB	jdk-8u261-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	134.23 MB	jdk-8u261-solaris-x64.tar.Z
Solaris x64	92.47 MB	jdk-8u261-solaris-x64.tar.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



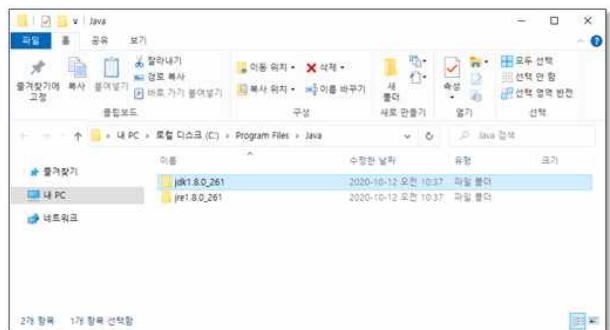
⑥ 다운로드 받은 설치파일 실행-04



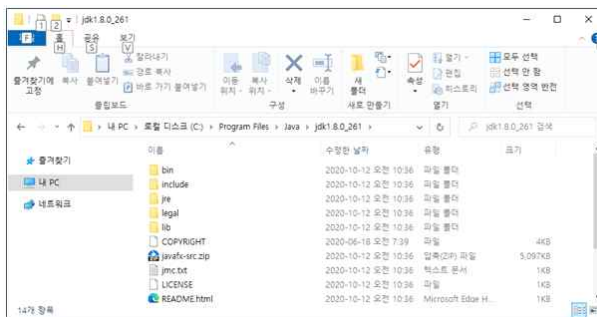
⑦ 다운로드 받은 설치파일 실행-05



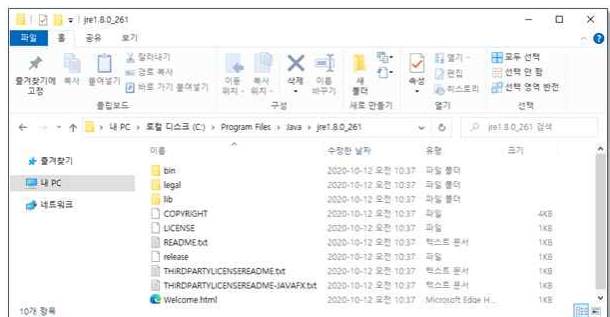
⑧ 설치경로 내 폴더 및 파일 생성 확인-01



⑨ 설치경로 내 폴더 및 파일 생성 확인-02



⑩ 설치경로 내 폴더 및 파일 생성 확인-03

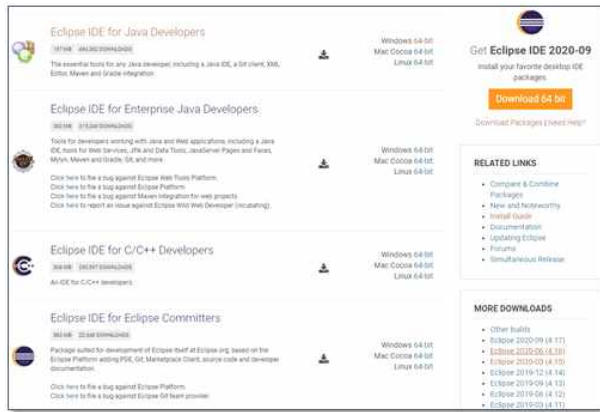


2) Eclipse IDE for Enterprise Java Developers(4.16) 다운로드 및 설치

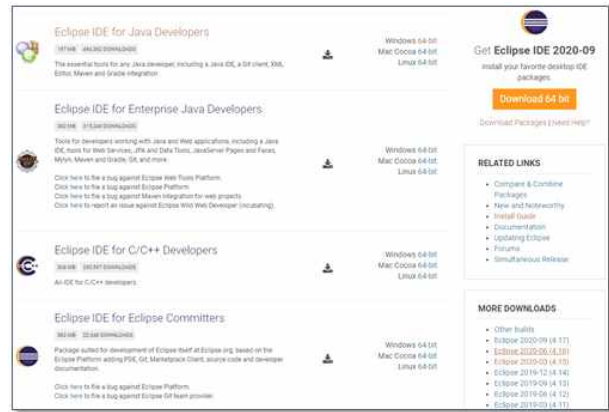
※ 이용 목적에 따라 진행 전 공식사이트 내 라이선스 정책을 확인 부탁드립니다.

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

① Eclipse 다운로드 사이트 접근



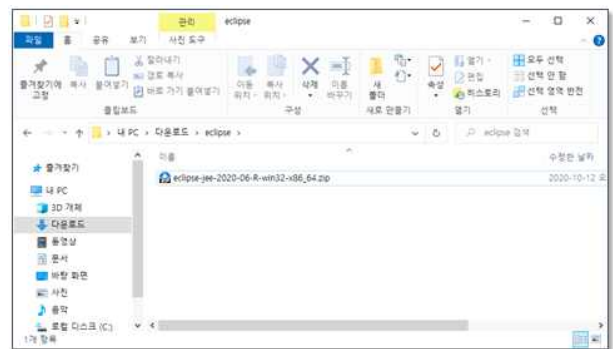
② Java 버전과 호환 가능한 4.16 버전 선택



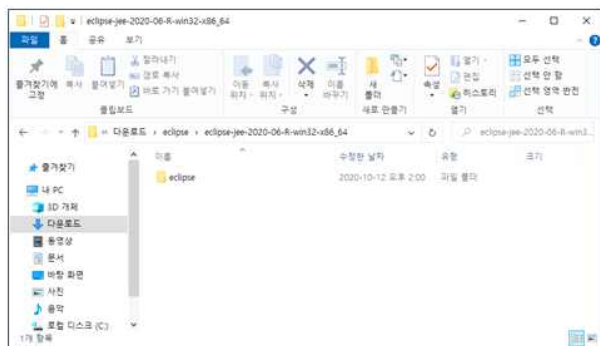
③ Enterprise Java Developers 다운로드



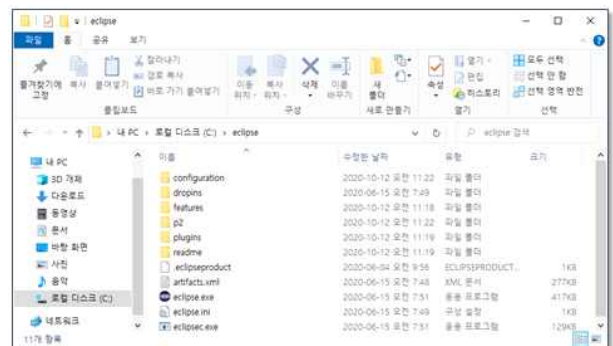
④ 다운로드 받은 ZIP 파일 확인



⑤ ZIP 파일 압축 해제 후 eclipse 폴더 확인



⑥ eclipse 폴더를 C드라이브 경로로 이동

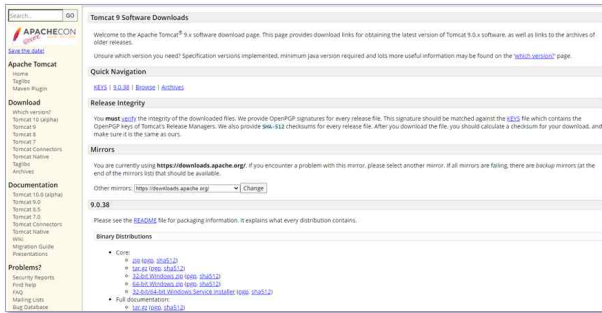


3) Apache Tomcat 9.0(9.0.38) 다운로드 및 설치

※ 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>)

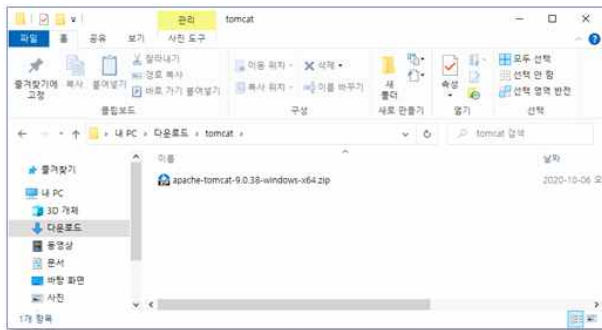
① Apache Tomcat 다운로드 사이트 접근



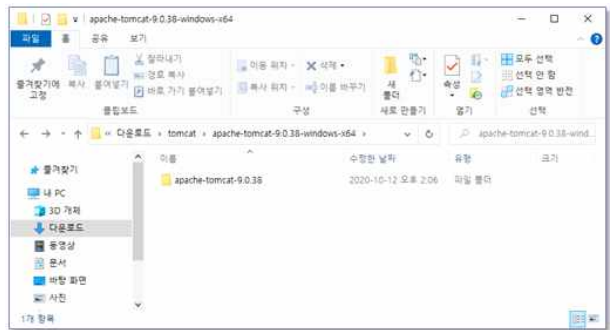
② Java 버전과 호환 가능한 9.0.38 버전 선택



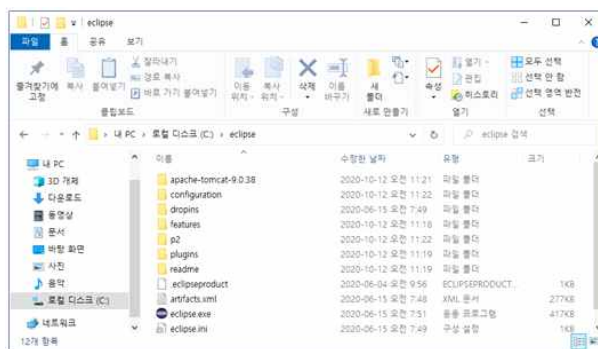
③ 다운로드 받은 ZIP 파일 확인



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

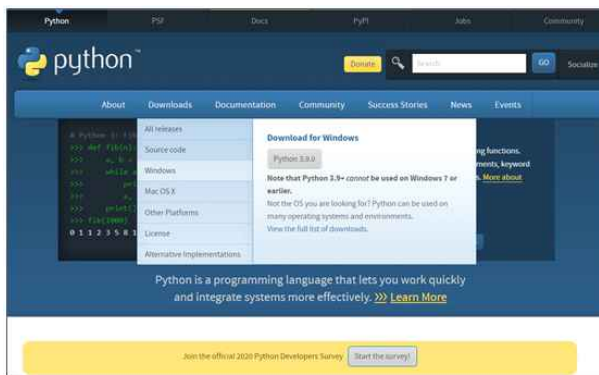
< 개발 환경 구축을 위한 프로그램별 버전 >

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/>)

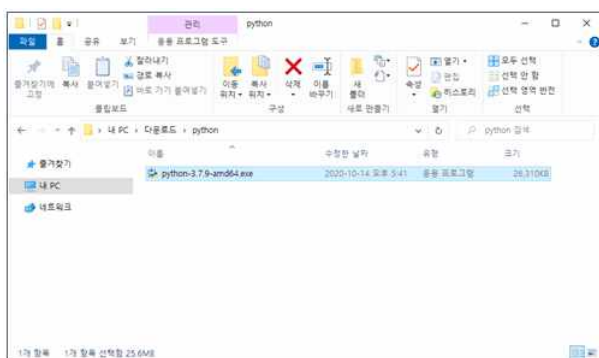
① Python 다운로드 사이트 접근



② 운영체제 호환 설치 파일 다운로드

- Python 3.7.9 - Aug. 17, 2020
- Note that Python 3.7.9 cannot be used on Windows XP or earlier.
- 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

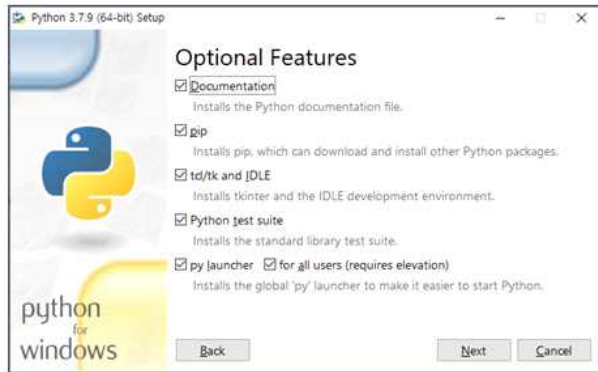
③ 다운로드 받은 설치파일 확인



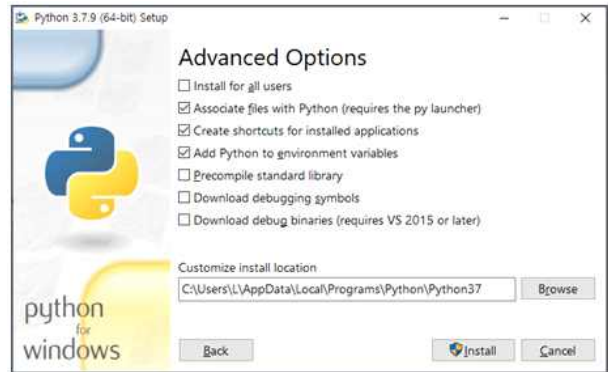
④ 다운로드 받은 설치파일 실행-01



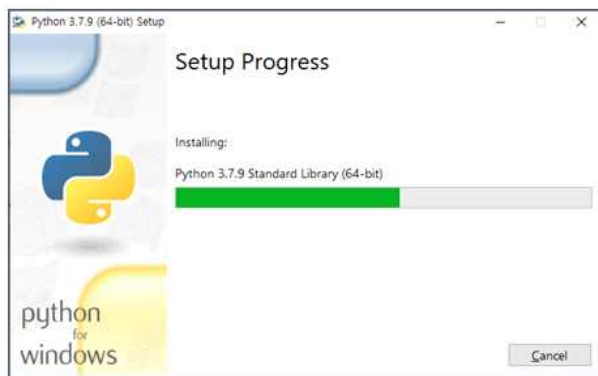
⑤ 다운로드 받은 설치파일 실행-02



⑥ 다운로드 받은 설치파일 실행-03



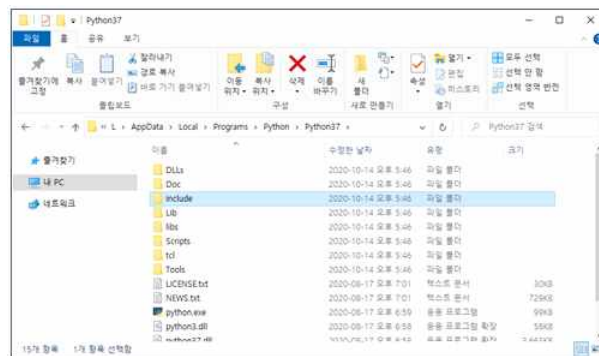
⑦ 다운로드 받은 설치파일 실행-04



⑧ 다운로드 받은 설치파일 실행-05



⑨ 설치경로 내 폴더 및 파일 생성 확인



2) Pycharm Community(2020.2.3) 다운로드 및 설치

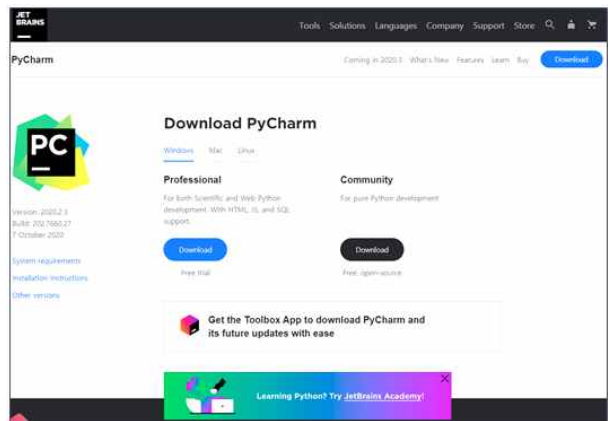
※ 이용 목적에 따라 진행 전 공식사이트 내 라이선스 정책을 확인 부탁드립니다.

(<https://jetbrains.com/pycharm/download/>)

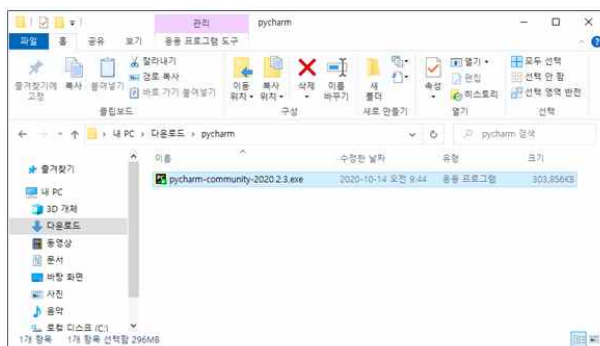
① Pycharm 다운로드 사이트 접근



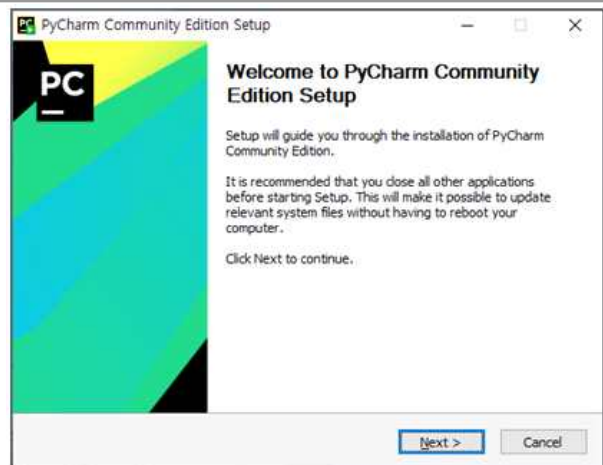
② Pycharm Community 다운로드



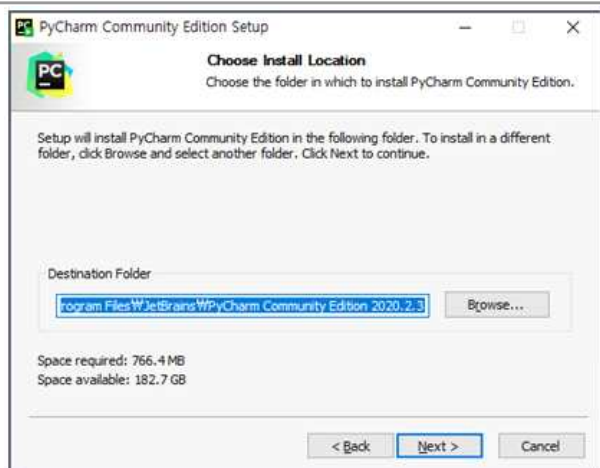
③ 다운로드 받은 설치파일 확인



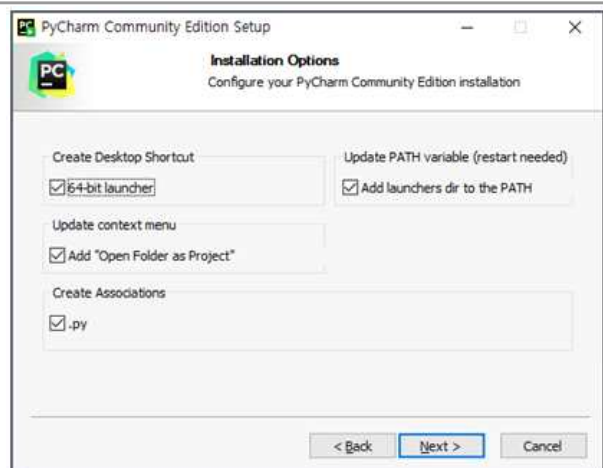
④ 다운로드 받은 설치파일 실행-01



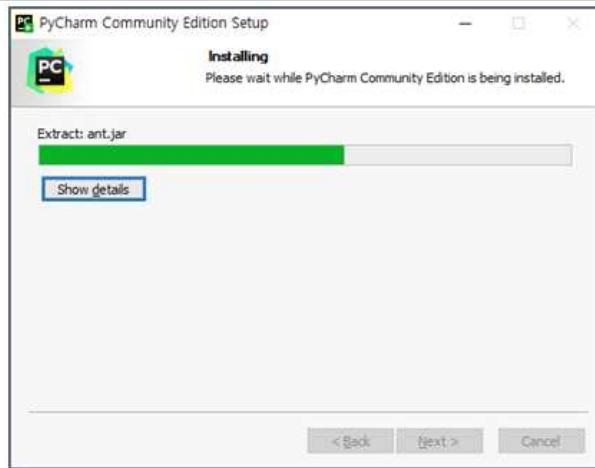
⑤ 다운로드 받은 설치파일 실행-02



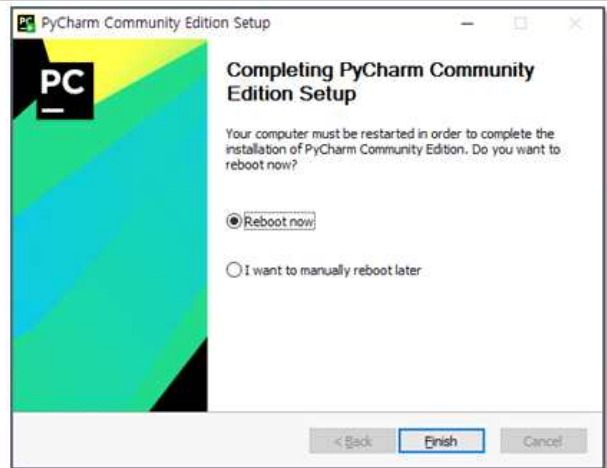
⑥ 다운로드 받은 설치파일 실행-03



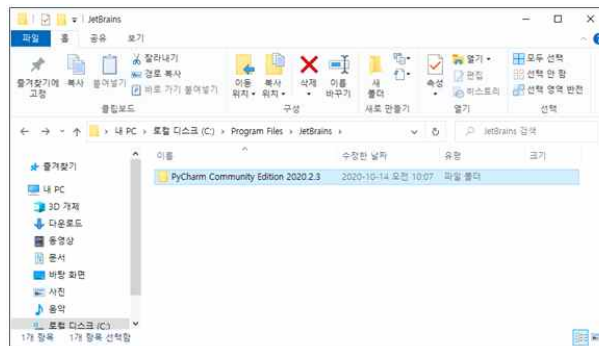
⑦ 다운로드 받은 설치파일 실행-04



⑧ 다운로드 받은 설치파일 실행-05



⑨ 설치경로 내 폴더 및 파일 생성 확인



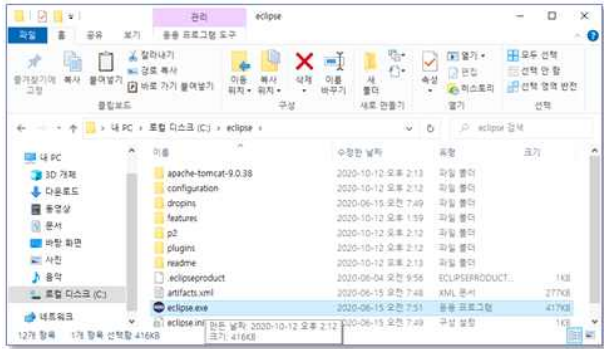
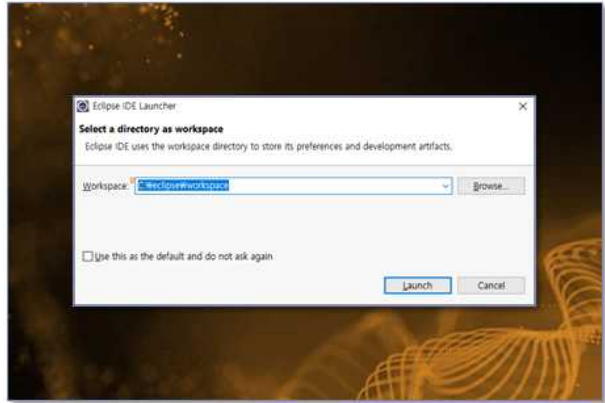
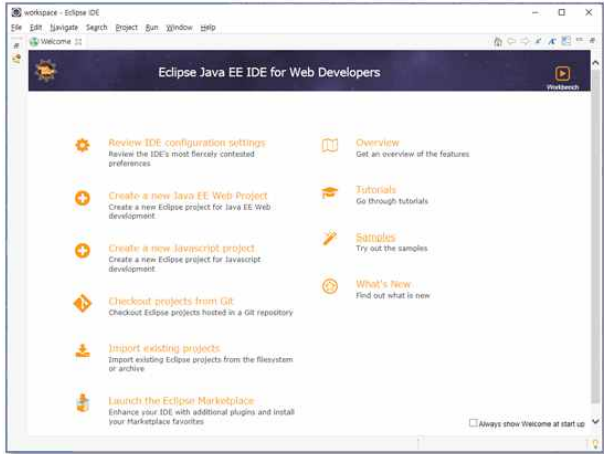
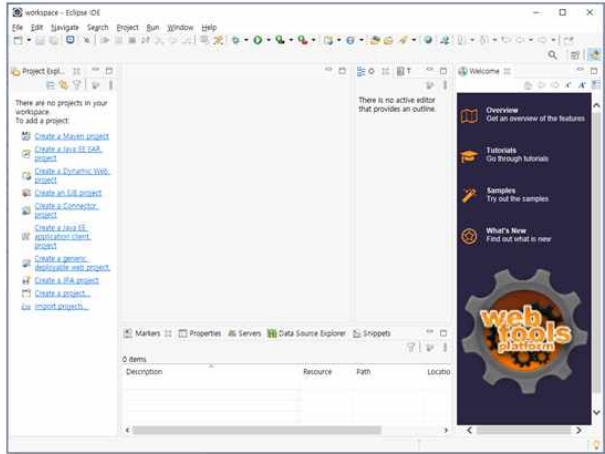
2 SOAP API 호출

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.

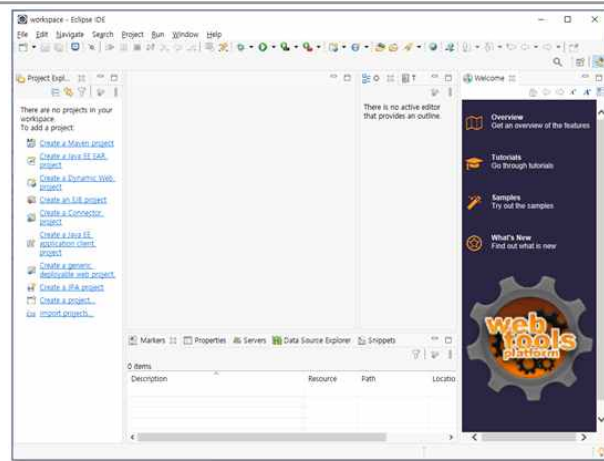
□ Eclipse 실행부터 프로그램 구현까지 SOAP API 호출 방법에 대한 가이드 문서이며, 기본적인 프로그램 구현 방식으로 작성되었습니다.

1) Eclipse 실행

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

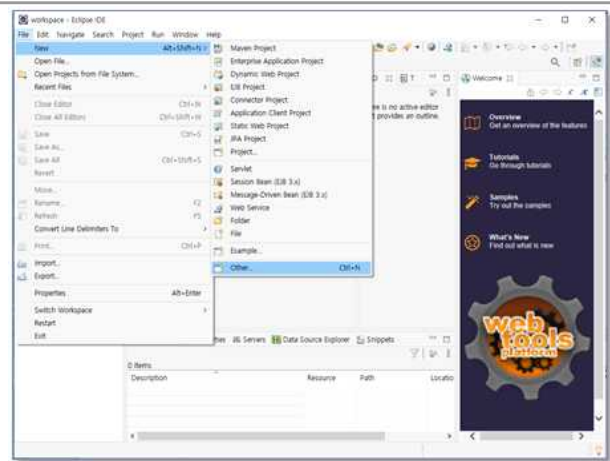
2) 신규 Project 생성

① Eclipse 초기 화면



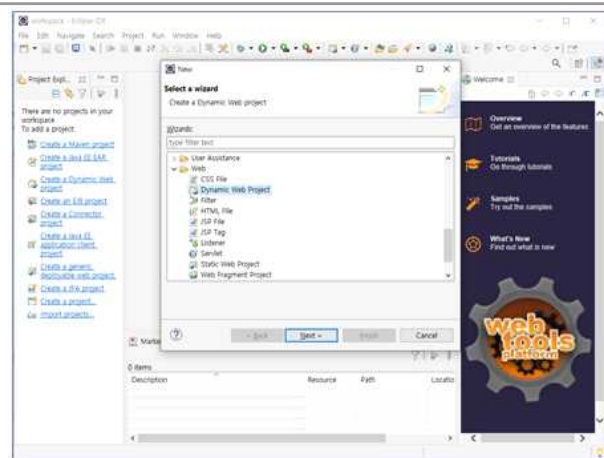
② 신규 프로젝트 생성-01

[메뉴] File → New → Other 선택



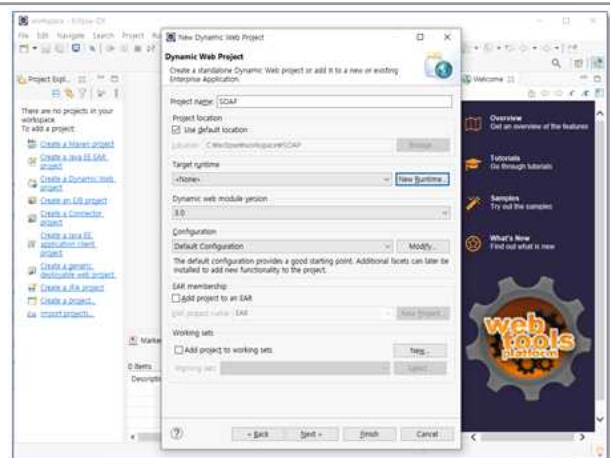
③ 신규 프로젝트 생성-02

[항목] Web → Dynamic Web Project 선택



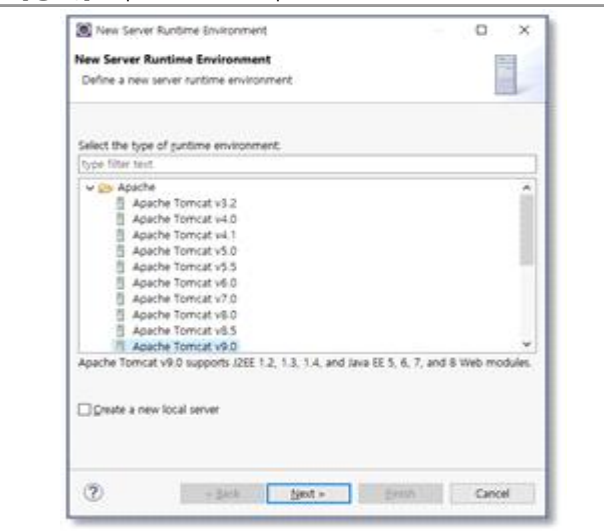
④ 신규 프로젝트 생성-03

Project name 작성 및 New Runtime 클릭



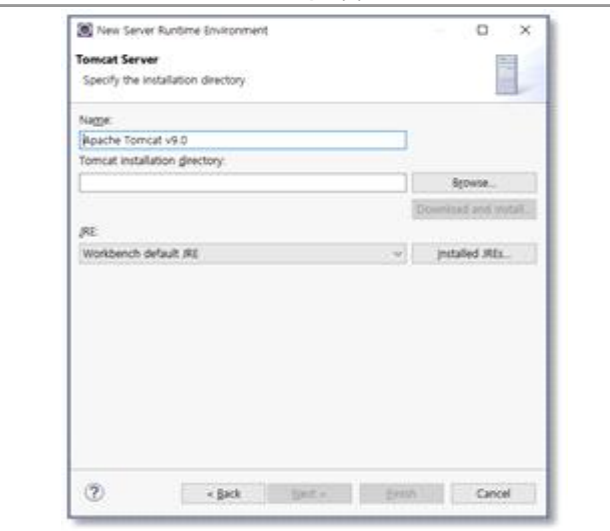
⑤ 신규 프로젝트 생성-04

[항목] Apache → Apache Tomcat v.9.0 선택

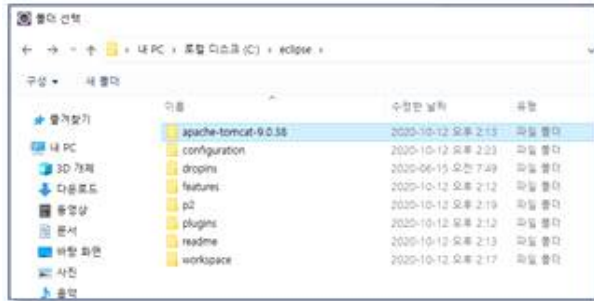


⑥ 신규 프로젝트 생성-05

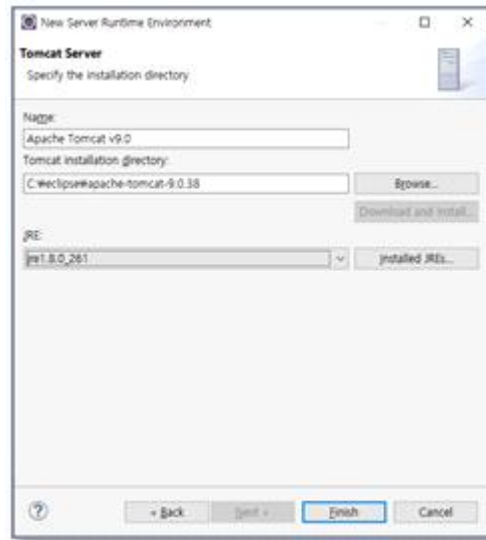
Server Name 작성 및 Browse 클릭



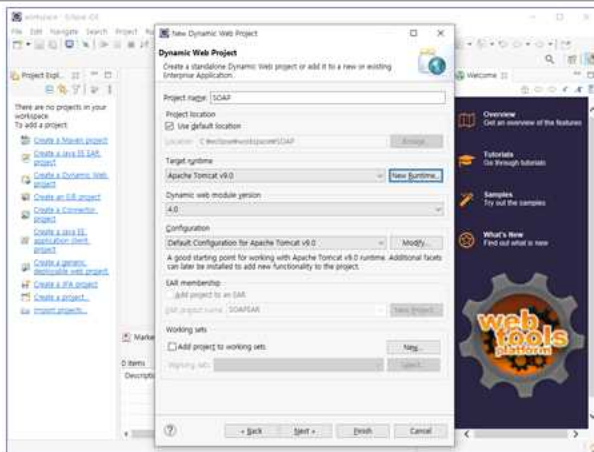
⑦ 신규 프로젝트 생성-06
기존에 생성한 tomcat 폴더 선택



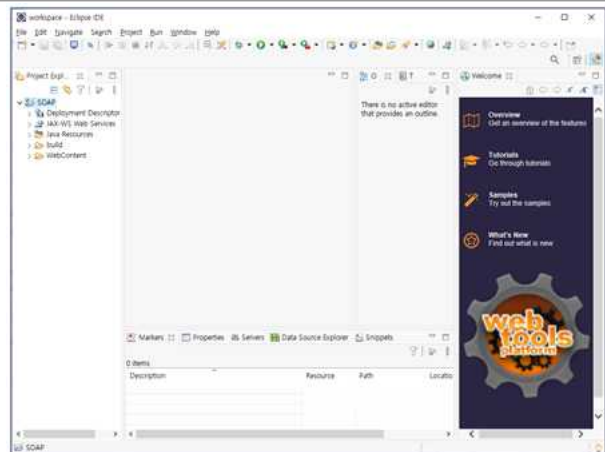
⑧ 신규 프로젝트 생성-07
기존에 생성한 JRE 선택



⑨ 신규 프로젝트 생성-08
Project 생성을 위한 기본 정보 입력 완료

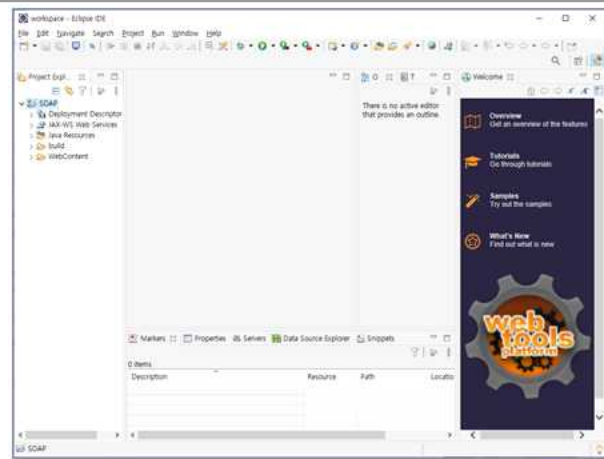


⑩ 신규 프로젝트 생성-09
생성된 신규 Project 확인

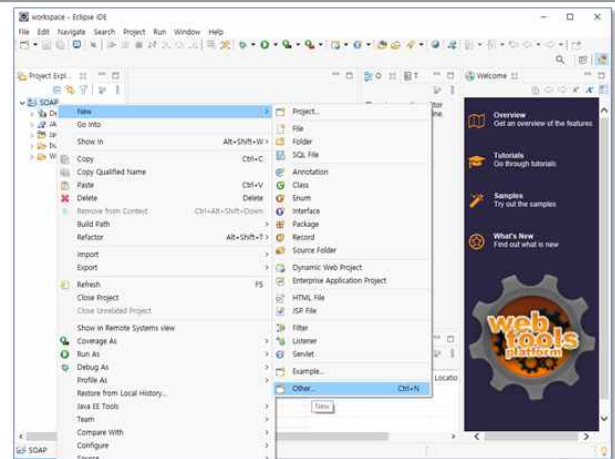


3) 신규 Project 내 Web Client 생성

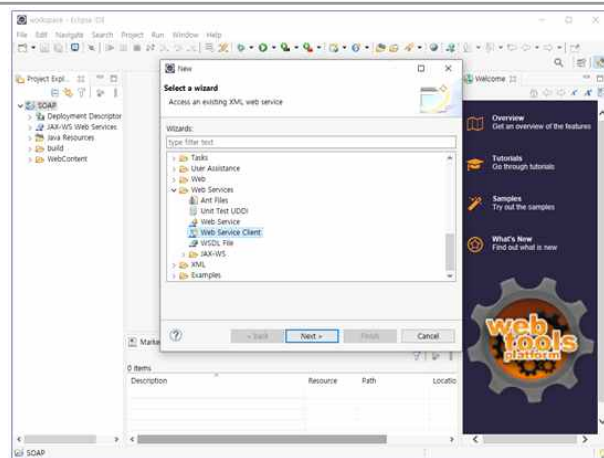
① 신규 생성된 Project 확인
신규 생성된 SOAP Project에 마우스 우 클릭



② Web Client 생성-01
[서브 메뉴] New → Other 선택



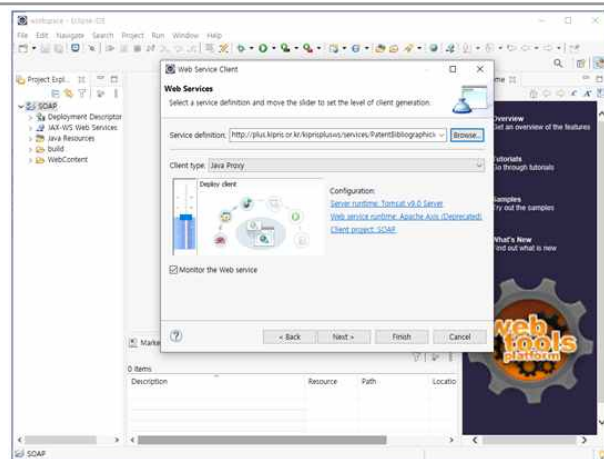
③ Web Client 생성-02
[항목] Web Service → Web Service Client 선택



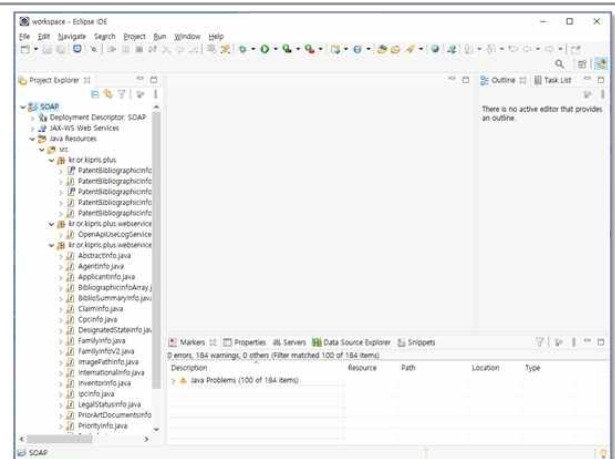
④ Web Client 생성-03
KIPRIS^{Plus} 상품화면 내 SOAP의 WSDL 확인



⑤ Web Client 생성-04
KIPRIS^{Plus}에서 제공하는 WSDL URL 입력

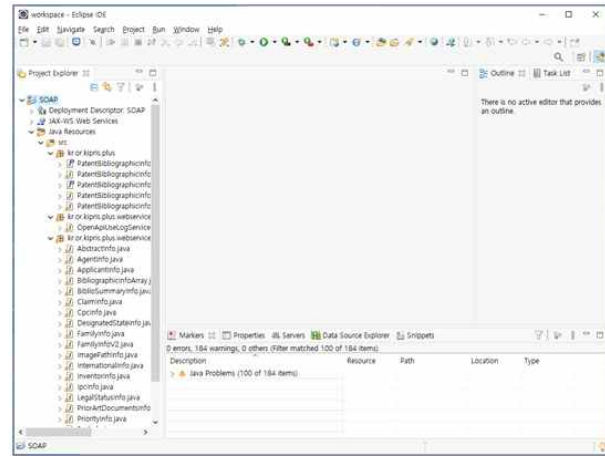


⑥ Web Client 생성-05
WSDL을 통해 Project 내 생성된 소스 정보 확인

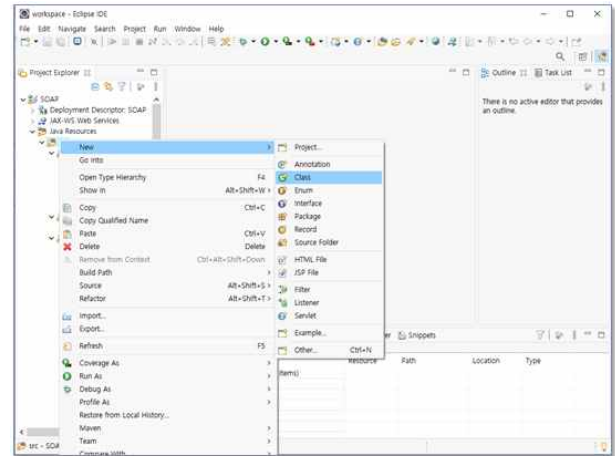


4) SOAP API 호출 프로그램 구현 및 테스트

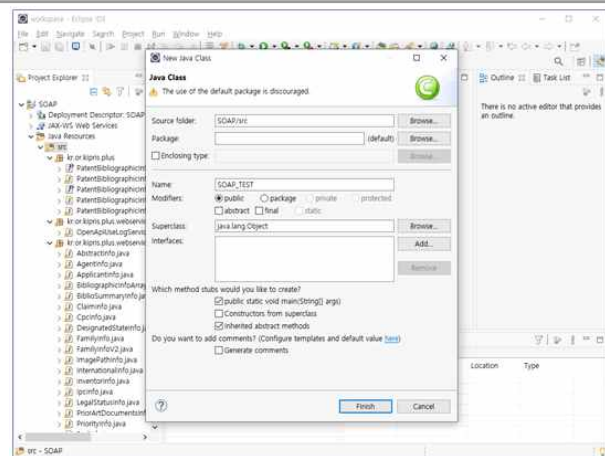
① Project 내 src 폴더 확인
src 폴더에 마우스 우 클릭



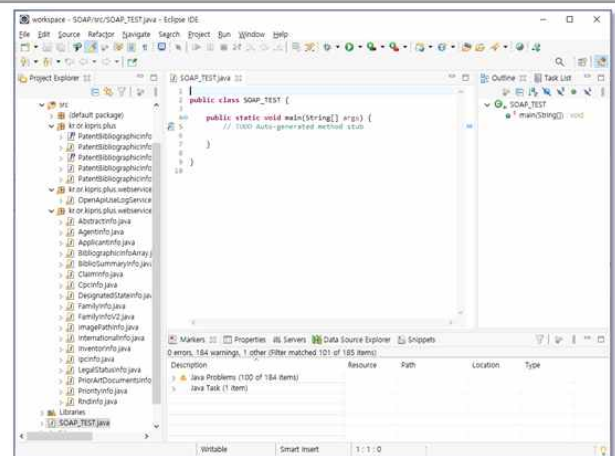
② Class 생성-01
[서브 메뉴] New → Class 선택



③ Class 생성-02
Name 입력, public static void main 체크



④ Class 생성-03
신규 생성된 Class 정보 확인



⑤ 샘플 소스 구현-01
SOAP API 호출 소스 선언(P.27 참고)

```
//wsdl를 통해 Client로 가져온 정보를 선언
PatentBibliographicInfoServicePortType proxy
= new PatentBibliographicInfoServicePortTypeProxy();

PatentBibliographicInfoServiceSoap11BindingStub stub
=(PatentBibliographicInfoServiceSoap11BindingStub)proxy.getPatentBibliographicInfoServicePortType();

//KIPRISPlus 에서 사용중인 본인의 ID를 입력
//KIPRISPlus 에서 제공받은 본인의 SOAP AccessKey 값을 입력
String _userId = " ";
String _userKey = " ";

//SOAP Header 에 id 정보 설정
SOAPHeaderElement id = new SOAPHeaderElement("http://plus.kipris.or.kr", "userId");
id.setValue(_userId);
stub.setHeader(id);

//SOAP Header 에 key 정보 설정
SOAPHeaderElement userKey = new SOAPHeaderElement("http://plus.kipris.or.kr", "userKey");
userKey.setValue(_userKey);
stub.setHeader(userKey);
```

⑥ 샘플 소스 구현-02
SOAP API 호출 소스 선언(P.27 참고)

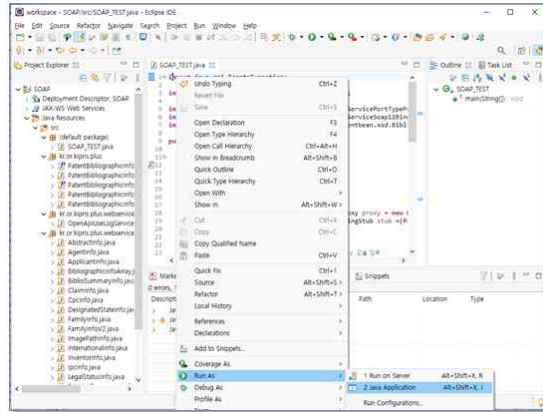
```
//결과(출력)값을 저장할 배열 정보 선언
BiblioSummaryInfo[] result = null;

try {
    //입력값(출력변수)를 통해 대상 호출
    result = stub.getBiblioSummaryInfo("1019950700550");
} catch (RemoteException e) {
    // T000 Auto-generated catch block
    e.printStackTrace();
}

//입력값(출력변수)를 통해 호출한 대상의 출력값을 확인
System.out.println("발명의명칭: [" + result[0].getInventionTitle()+"]");
System.out.println("출원번호: [" + result[0].getApplicationNumber()+"]");
System.out.println("출원일자: [" + result[0].getApplicationDate()+"]");
System.out.println("등록번호: [" + result[0].getRegistrationNumber()+"]");
System.out.println("등록일자: [" + result[0].getRegisterDate()+"]");
System.out.println("등록상태: [" + result[0].getRegisterStatus()+"]");
```

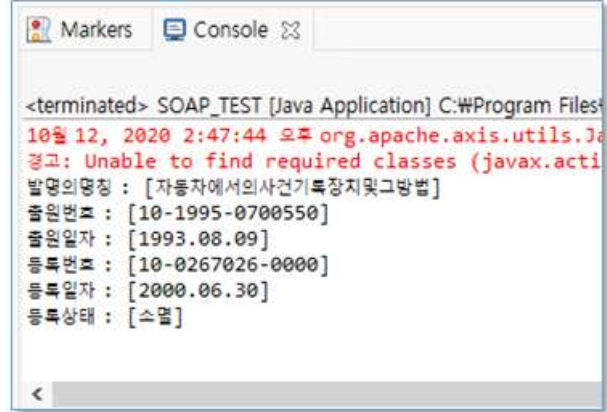

⑦ SOAP API 샘플 호출 결과 확인-01

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         /** SOAP API 기본 호출 소스 정보 */
15
16         //wsdl을 통해 Client로 가져온 정보를 선언
17         PatentBibliographicInfoServicePortTypeProxy proxy = new PatentBibliographicInfoServicePortTypeProxy();
18         PatentBibliographicInfoServiceSoap11BindingStub stub = (PatentBibliographicInfoServiceSoap11BindingStub)proxy.getPatentBibliographicInfoServicePortType();
19
20         //KIPRISplus 에서 사용중인 본인의 ID를 입력
21         //KIPRISplus 에서 제공받은 본인의 SOAP AccessKey 값을 입력
22         String _userId = "10-1995-0700550";
23         String _userKey = "10-0267026-0000";
24
25         //SOAP Header 에 id 정보 설정
26         SOAPHeaderElement id = new SOAPHeaderElement("http://plus.kipris.or.kr", "userId");
27         id.setValue(_userId);
28         stub.setHeader(id);
29
30         //SOAP Header 에 key 정보 설정
31         SOAPHeaderElement userKey = new SOAPHeaderElement("http://plus.kipris.or.kr", "userKey");
32         userKey.setValue(_userKey);
33         stub.setHeader(userKey);
34
35         //결과(출력)값을 저장할 배열 정보 선언
36         BiblioSummaryInfo[] result = null;
37
38         try {
39             //입력값(출력본드)를 통해 대상 호출
40             result = stub.getBiblioSummaryInfo("1019950700550");
41         } catch (RemoteException e) {
42             // TODO Auto-generated catch block
43             e.printStackTrace();
44         }
45
46         //입력값(출력본드)를 통해 호출한 대상의 출력값을 확인
47         System.out.println("발명의명칭 : [" + result[0].getInventionTitle()+""]");
48         System.out.println("출원번호 : [" + result[0].getApplicationNumber()+""]");
49         System.out.println("출원일자 : [" + result[0].getApplicationDate()+""]");
50         System.out.println("등록번호 : [" + result[0].getRegisterNumber()+""]");
51         System.out.println("등록일자 : [" + result[0].getRegisterDate()+""]");
52         System.out.println("등록상태 : [" + result[0].getRegisterStatus()+""]");
53     }
54 }

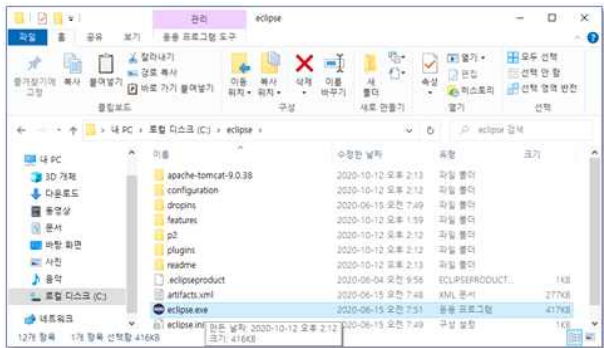
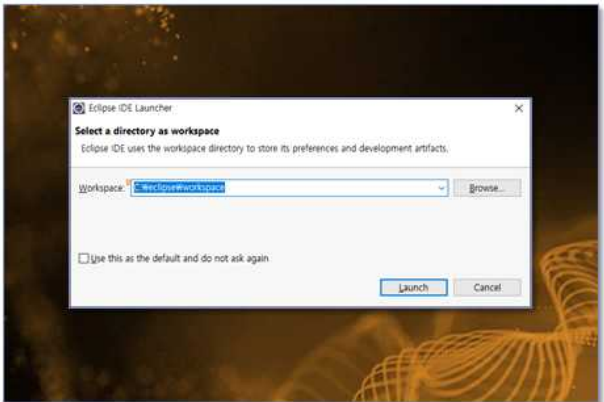
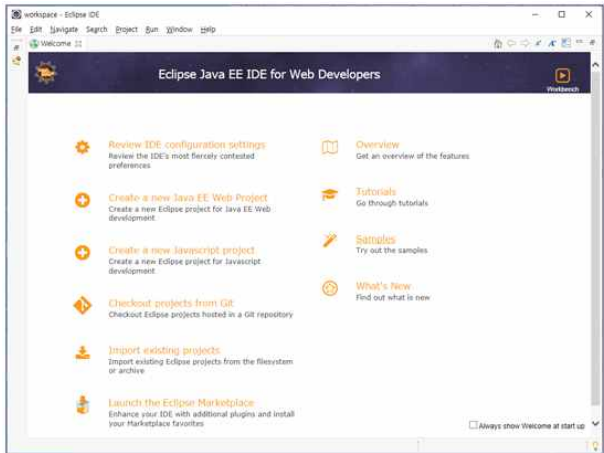
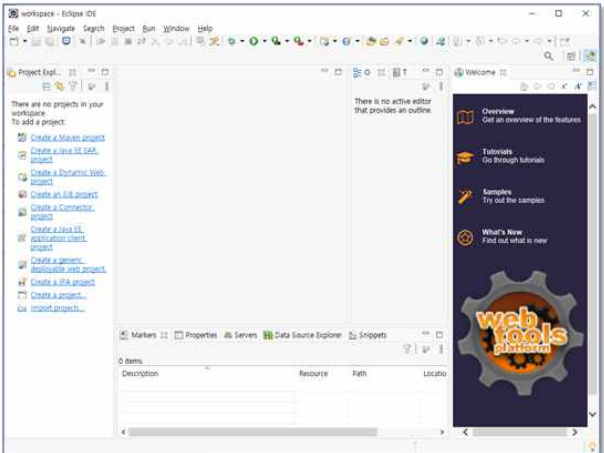
```

3 REST API 호출

3.1. Java를 이용한 REST API 호출 구현

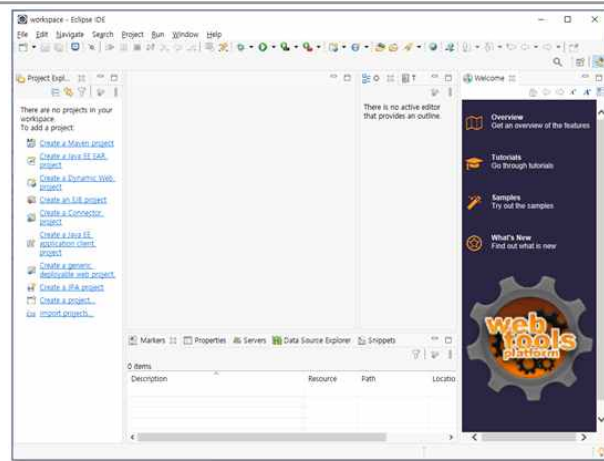
- Eclipse 실행부터 프로그램 구현까지 REST API 호출 방법에 대한 가이드 문서이며, 기본적인 프로그램 구현 방식으로 작성되었습니다.

1) Eclipse 실행

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

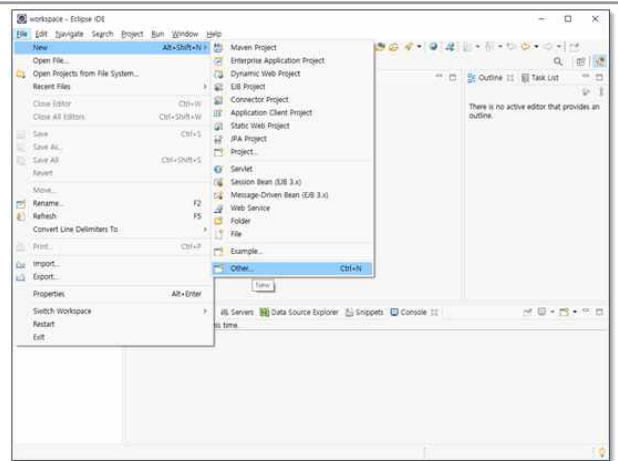
2) 신규 Project 생성

① Eclipse 초기 화면



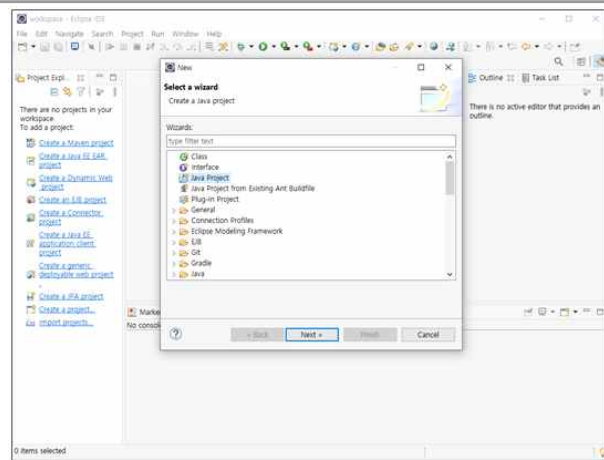
② 신규 프로젝트 생성-01

[메뉴] File → New → Other 선택



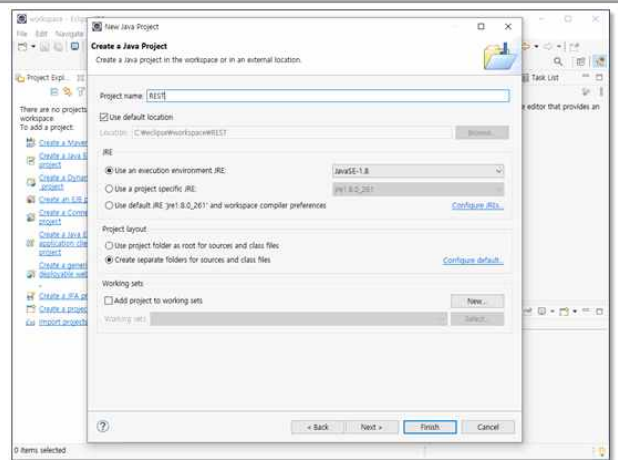
③ 신규 프로젝트 생성-02

[항목] Java Project 선택



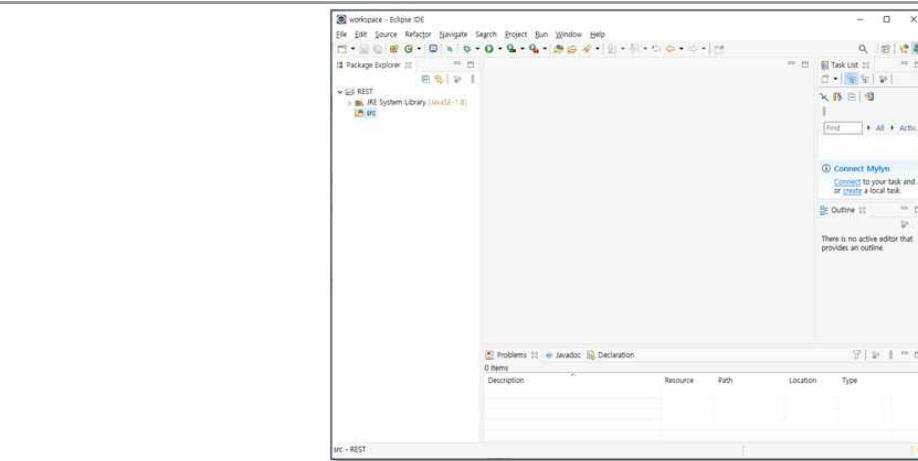
④ 신규 프로젝트 생성-03

Project 생성을 위한 기본 정보 입력 완료



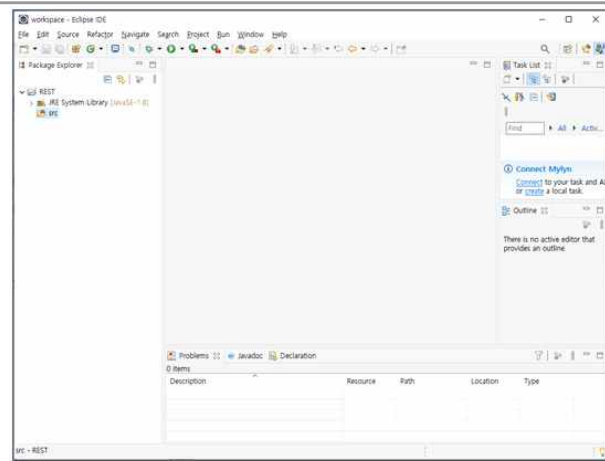
⑤ 신규 프로젝트 생성-04

생성된 신규 Project 확인

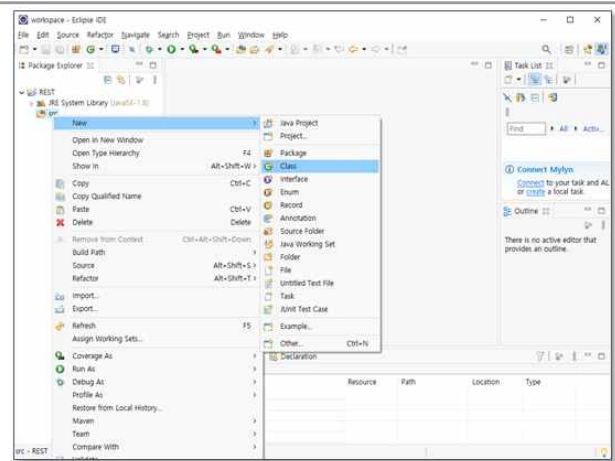


3) REST API 호출 프로그램 구현 및 테스트

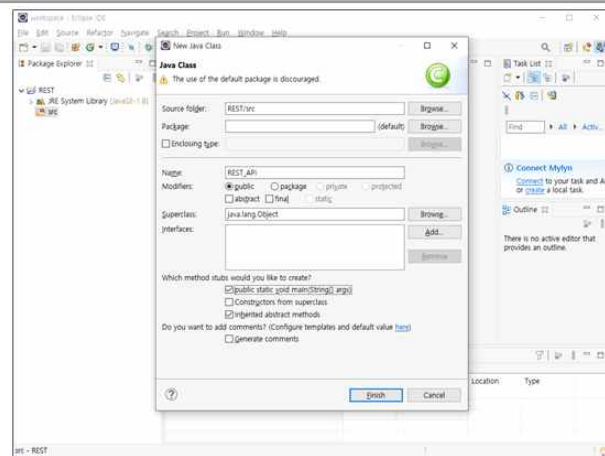
① Project 내 src 폴더 확인
src 폴더에 마우스 우 클릭



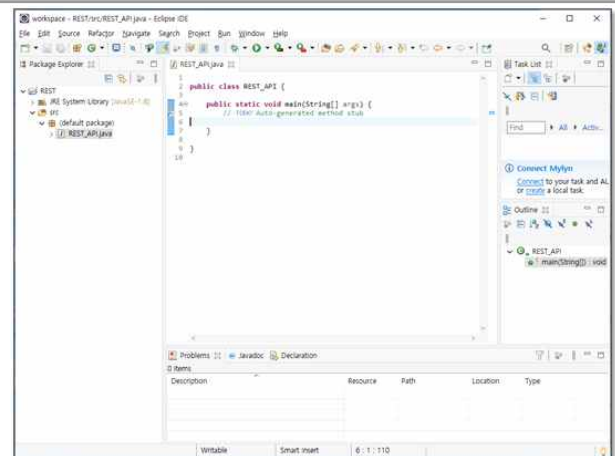
② Class 생성-01
[서브 메뉴] New → Class 선택



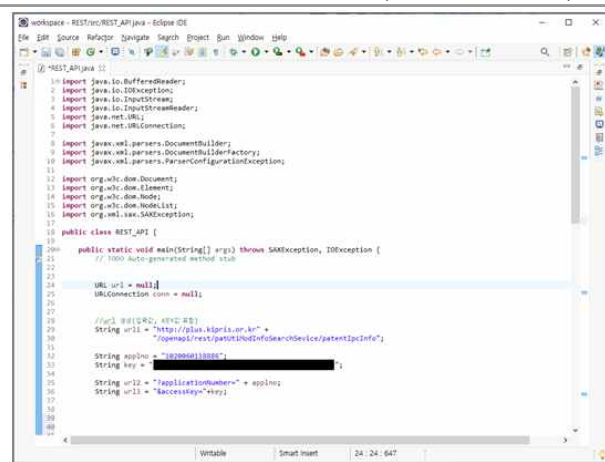
③ Class 생성-02
Name 입력, public static void main 체크



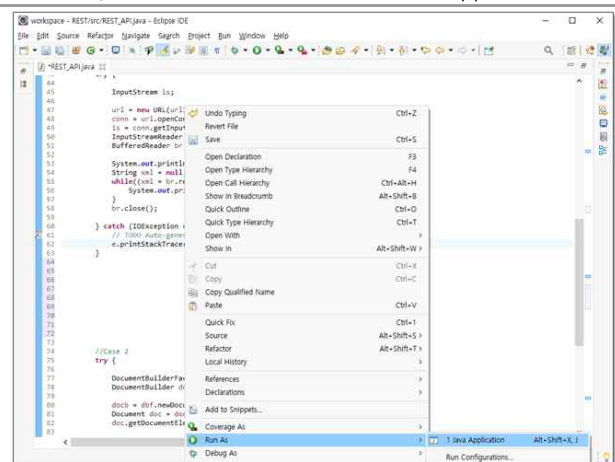
④ Class 생성-03
신규 생성된 Class 정보 확인



⑤ 샘플 소스 구현
REST API 호출 소스 선언(P.31~32 참고)



⑥ REST API 샘플 호출 결과 확인-01
Class 마우스 우 클릭 → Run As → Java Application 선택



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

```
*REST_API.java 32
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.net.URLConnection;
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 생성(주소, KEY값 포함)
29         String url1 = "http://plus.kipris.or.kr" +
30             "/openapi/rest/patUtilModInfoSearchService/patentIpcInfo";
31
32         String appno = "1020060116886";
33         String key = "XXXXXXXXXXXX";
34
35         String url2 = "?applicationNumber=" + appno;
36         String url3 = "&accessKey="+key;
```

[참고] 샘플 소스 프로그램 구현 예시-02

```

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).getTextCo
91         String date = ((Element)n).getElementsByTagName("InternationalpatentclassificationDate").item(0).getTextCon
92         System.out.println("\n\n결과 Case 2");
93         System.out.println("IPC : "+ipc);
94         System.out.println("일자 : "+date);
95
96     }
97
98
99 } catch (ParserConfigurationException e) {
100     // TODO Auto-generated catch block
101     e.printStackTrace();
102 }
103
104
105
106
107 }
108
109 }

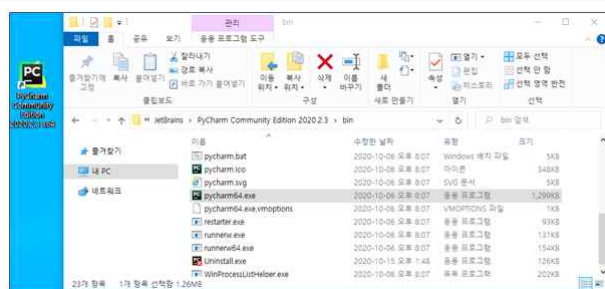
```

3.2. Python을 이용한 REST API 호출 구현

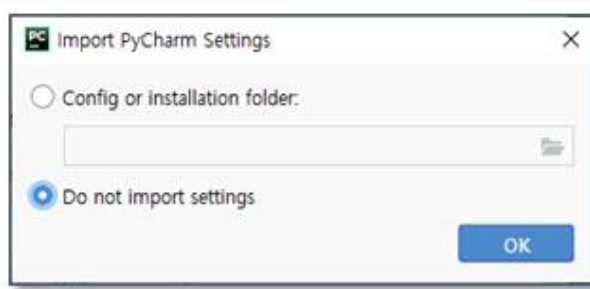
- Pycharm 실행부터 프로그램 구현까지 REST API 호출 방법에 대한 가이드 문서이며, 기본적인 프로그램 구현 방식으로 작성되었습니다.

1) Pycharm Community 실행

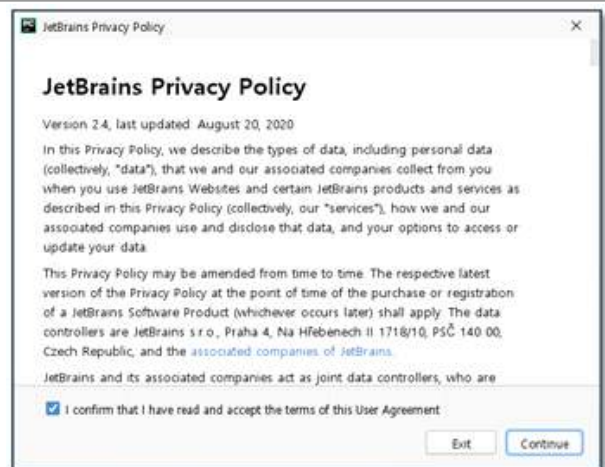
① Pycharm Community 실행



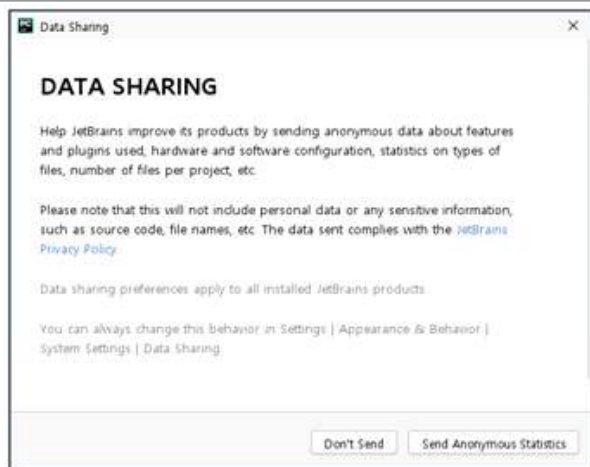
② 최초 실행 시 확인-1



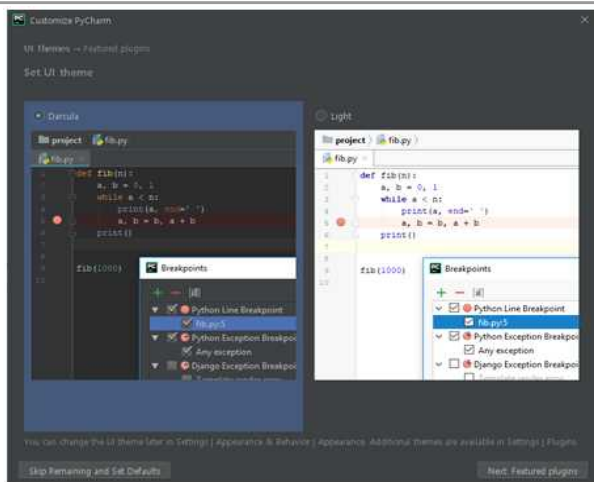
③ 최초 실행 시 확인-2



④ 최초 실행 시 확인-3

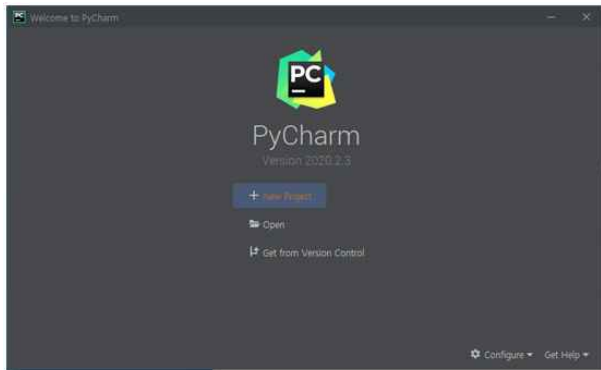


⑤ 최초 실행 시 테마 확인

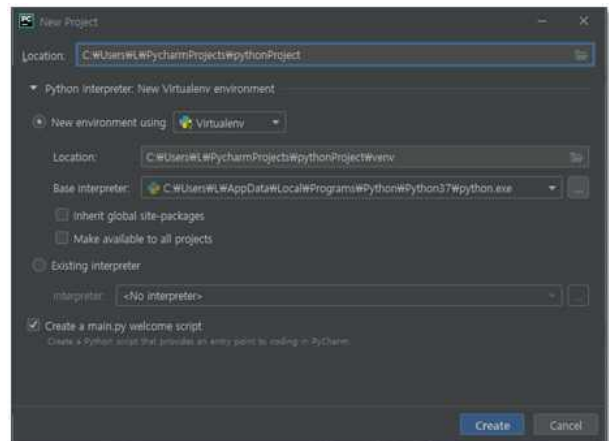


2) 신규 Project 생성

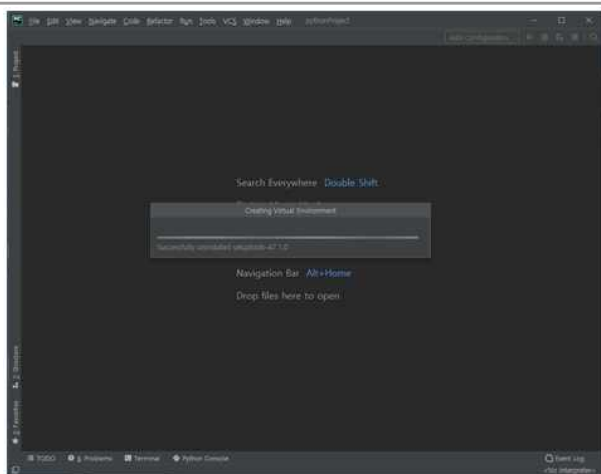
⑦ 최초 실행 시 프로젝트 생성 및 설정



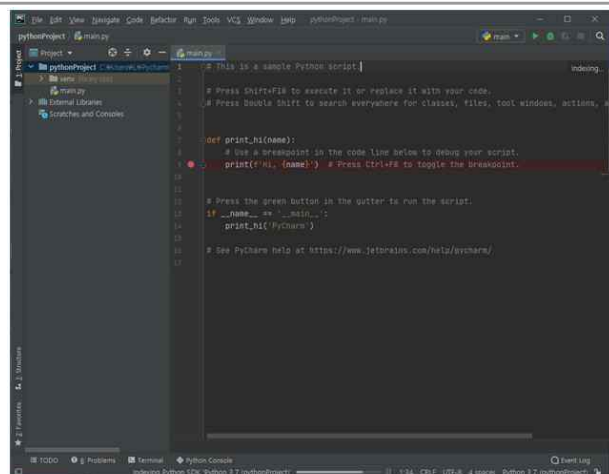
⑧ 신규 프로젝트 생성-01



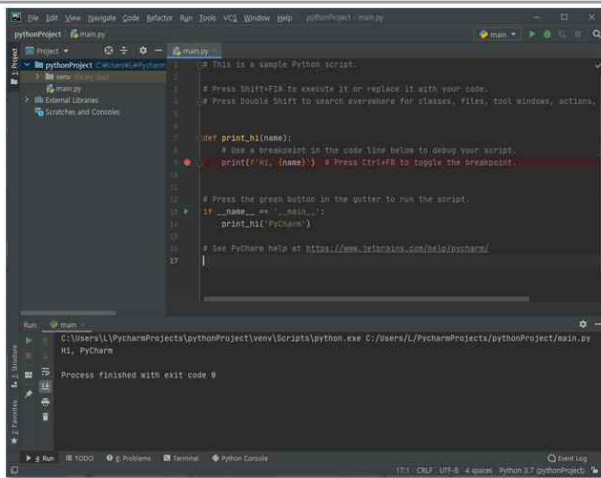
③ 신규 프로젝트 생성-02 프로젝트 생성 중



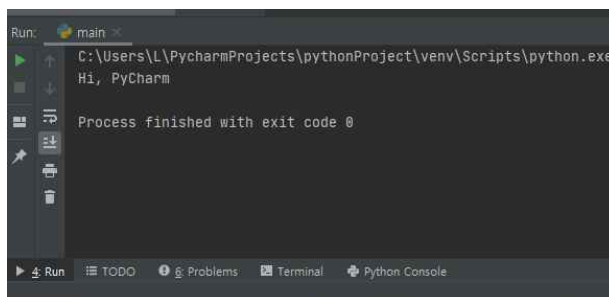
④ 신규 프로젝트 생성-03 기본 구현된 프로그램 확인



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



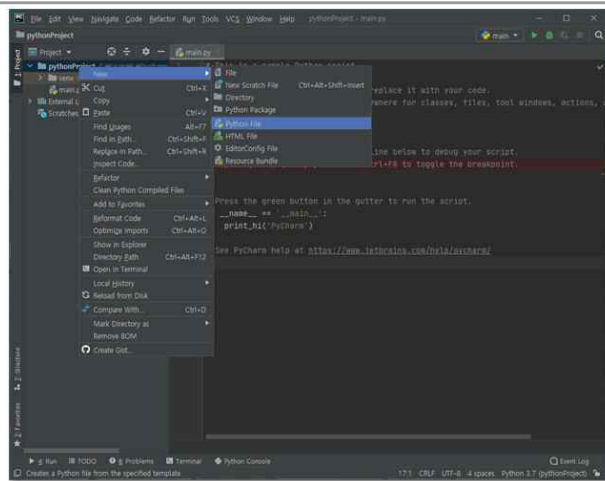
⑥ 신규 프로젝트 생성-05 하단 Window창을 통해 실행 결과 확인



3) REST API 호출 프로그램 구현 및 테스트

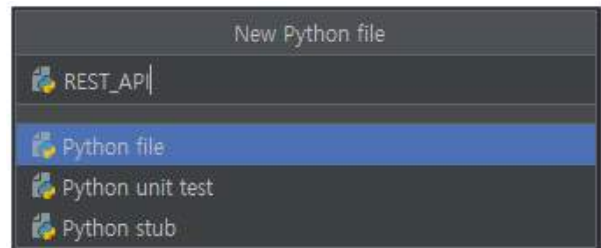
① Python File 생성-01

프로젝트 마우스 우 클릭 → New → Python File 선택



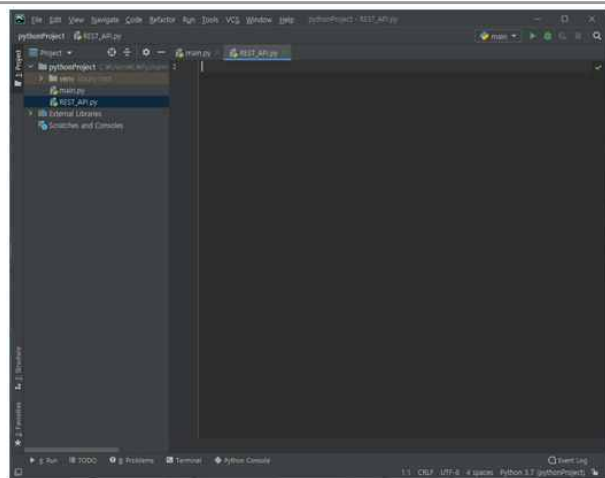
② Python File 생성-02

파일명 입력 후 신규 .py 파일 생성



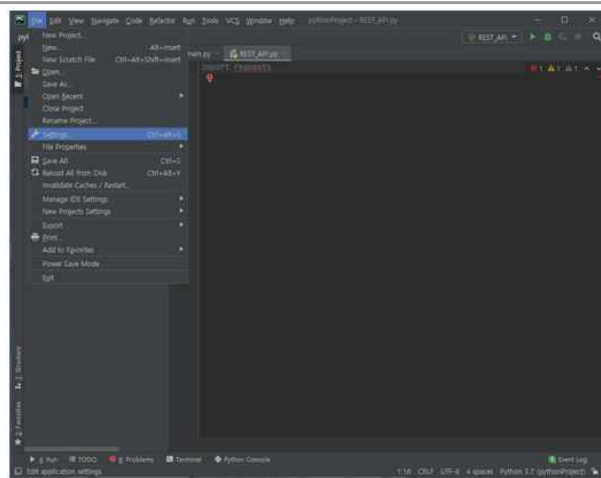
③ Python File 생성-02

생성된 파일 확인



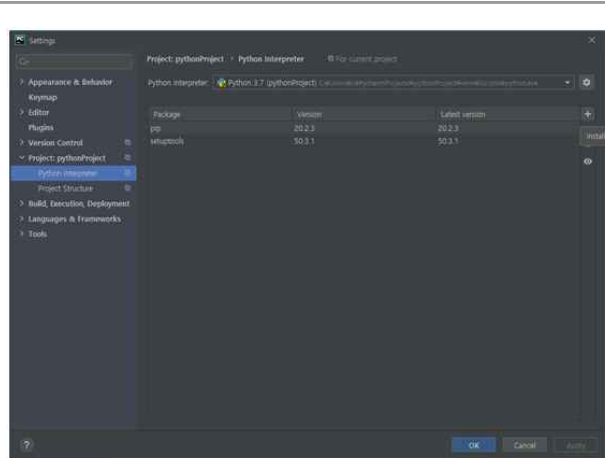
④ 신규 Package 설치 - 01

[메뉴] File → Setting 선택



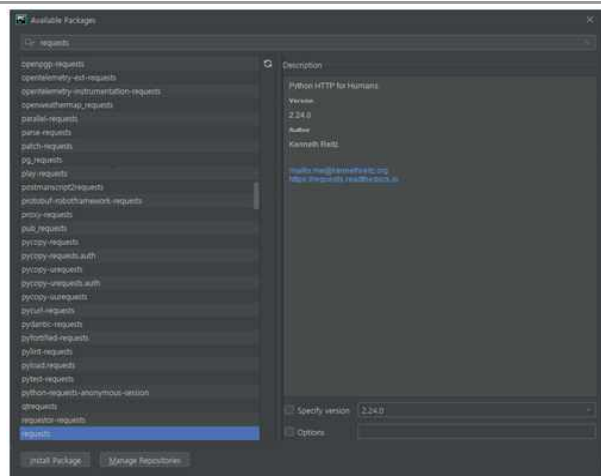
⑤ 신규 Package 설치 - 02

Project python Interpreter 화면의 + 선택

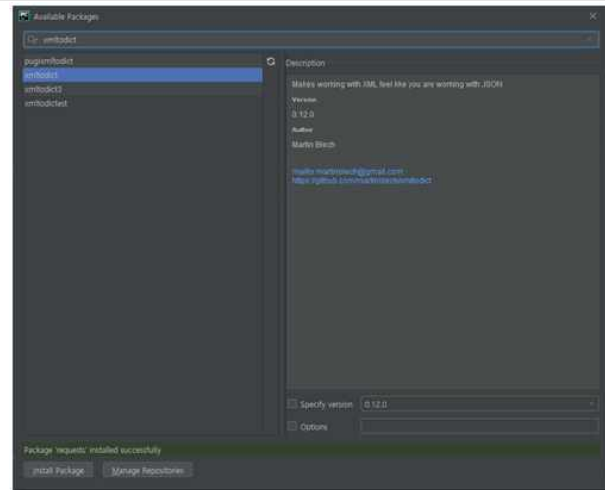


⑥ 신규 Package 설치 - 03

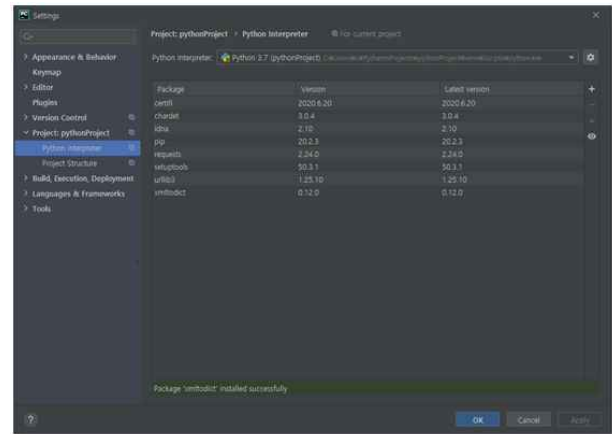
requests 검색 후 하단 install 진행



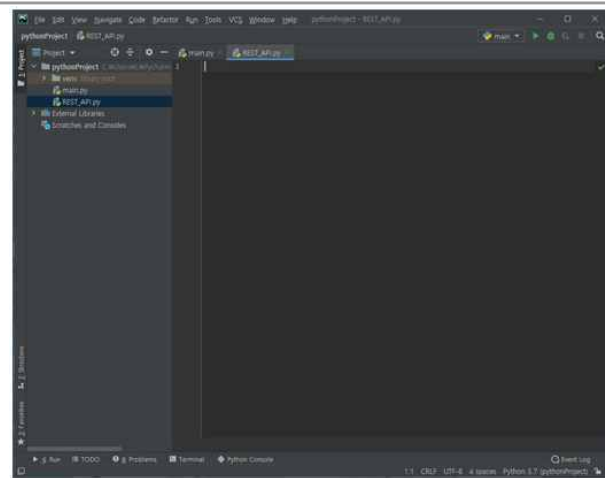
⑦ 신규 Package 설치 - 04
xmltodict 검색 후 하단 install 진행



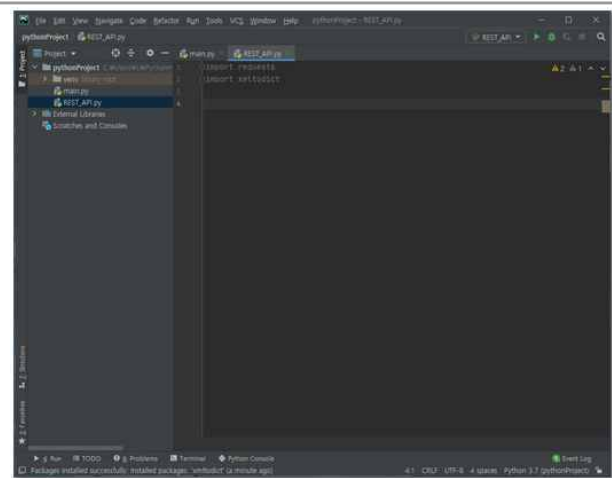
⑧ 신규 Package 설치 - 05
install 완료 정보 확인



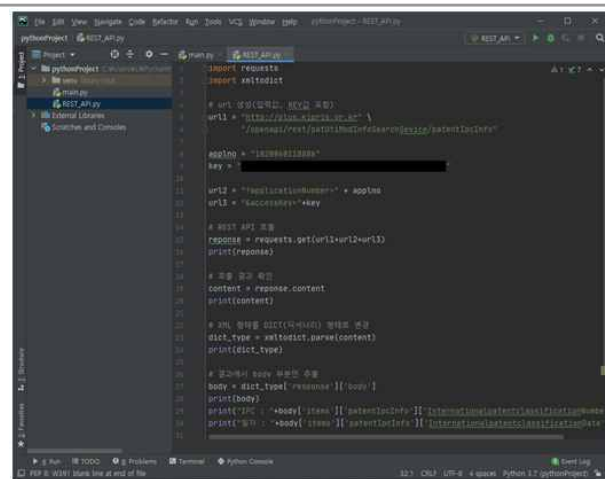
⑨ 샘플 소스 구현-01
신규 생성한 파일 확인



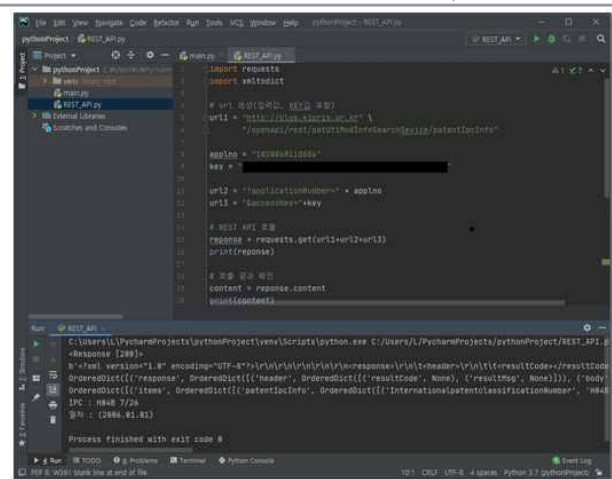
⑩ 샘플 소스 구현-02
import 시 앞 단계 requests, xmltodict 설치 필요



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



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



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 상태 화면(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 공개 · 등록공보 상품에 대한 데이터 수집에 대한 예시입니다.</p> <p>Case 1. 변동정보 기능을 이용한 데이터 수집</p> <ol 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> <ol 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> <ol 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 부탁드립니다. 인터넷에 공개된 게시 글, 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>