
KOSIS 공유서비스(OpenAPI) 개발가이드



통계청

목 차

1. KOSIS 공유서비스(OpenAPI) 개요	5
1.1 제공 콘텐츠	6
1.1.1 통계목록	6
1.1.2 통계자료	7
1.1.3 대용량 통계자료	8
1.1.4 통계설명	9
1.1.5 메타자료	10
1.1.6 KOSIS통합검색	11
1.1.7 통계주요지표	12
1.2 회원가입	13
1.3 인증키 발급	14
1.3.1 활용신청	14
1.3.2 신청현황	15
1.4 에러메시지	16
1.4.1 오류메시지 형식	16
1.4.2 오류메시지 유형	16
2. KOSIS 공유서비스(OpenAPI) 주요기능 및 활용방법	17
2.1 통계목록	17
2.1.1 특징	17
2.1.2 서비스활용	18
2.1.1.1 URL생성	18
2.1.3 활용방법	19
2.1.1.1 JSON	19
2.1.1.1 SDMX(Category)	26
2.2 통계자료	33
2.2.1 특징	33
2.2.2 서비스 활용	33
2.2.2.1 URL 생성	33
2.2.2.2 자료등록 URL 생성	35
2.2.2.2.1 자료등록	35
2.2.2.2.2 등록된 자료	36
2.2.2.2.3 URL생성	37
2.2.3 활용방법	39
2.2.3.1 JSON	39
2.2.3.2 SDMX(DSD)	54
2.2.3.3 SDMX(Generic)	70

2.2.3.4 SDMX(StructureSpecific)	86
2.3 대용량 통계자료	102
2.3.1 특징	102
2.3.2 서비스활용	102
2.3.2.1 자료등록	102
2.3.2.2 등록된자료	103
2.3.2.3 URL 생성	104
2.3.3 활용방법	106
2.3.3.1 SDMX(DSD)	106
2.3.3.2 SDMX(Generic)	107
2.3.3.3 SDMX(StructureSpecific)	108
2.3.3.4 XLS	109
2.4 통계설명	110
2.4.1 특징	110
2.4.2 서비스활용	111
2.4.2.1 URL생성	111
2.4.3 활용방법	112
2.4.3.1 JSON	112
2.4.3.2 XML	123
2.5 메타자료	136
2.5.1 특징	136
2.5.2 서비스활용	136
2.5.2.1 URL생성	136
2.5.3 활용방법	137
2.5.3.1 JSON(통계표 명칭)	137
2.5.3.2 JSON(기관 명칭)	137
2.5.3.3 JSON(수록정보)	138
2.5.3.4 JSON(분류/항목)	138
2.5.3.5 JSON(주석)	139
2.5.3.6 JSON(단위)	139
2.5.3.7 JSON(출처)	140
2.5.3.8 JSON(가중치)	140
2.5.3.9 JSON(자료갱신일)	141
2.5.3.10 XML(통계표 명칭)	141
2.5.3.11 XML(기관 명칭)	142
2.5.3.12 XML(수록정보)	142
2.5.3.13 XML(분류/항목)	143
2.5.3.14 XML(주석)	143
2.5.3.15 XML(단위)	144
2.5.3.16 XML(출처)	144
2.5.3.17 XML(가중치)	145

2.5.3.18 XML(자료갱신일)	145
2.6 KOSIS통합검색	146
2.6.1 특징	146
2.6.2 서비스활용	146
2.6.2.1 URL생성	146
2.6.3 활용방법	147
2.6.3.1 JSON	147
2.7 통계주요지표	148
2.7.1 특징	148
2.7.2 서비스활용	148
2.7.2.1 URL생성	148
2.7.3 활용방법	149
2.7.3.1 JSON,XML(지표고유번호별 설명자료조회)	149
2.7.3.2 JSON,XML(지표명별 설명자료조회)	150
2.7.3.3 JSON,XML(목록별 지표조회)	151
2.7.3.4 JSON,XML(지표명별 목록조회)	152
2.7.3.5 JSON,XML(고유번호별 목록조회)	153
2.7.3.6 JSON,XML(고유번호별 지표 상세조회)	154
2.7.3.7 JSON,XML(수록주기별 목록조회)	155
2.7.3.8 JSON,XML(지표명별 상세조회)	156

<참고> SDMX	157
-----------------	-----

1. KOSIS 공유서비스(OpenAPI) 개요

KOSIS 공유서비스는 외부(기관,개인)에서 국가통계포털(KOSIS)의 통계정보를 서비스 또는 콘텐츠 개발에 활용할 수 있도록 인터페이스(API)를 제공하는 서비스입니다. 제공되는 정보는 아래와 같습니다.

서비스 대상	제공 형태
1) 통계목록 : KOSIS에서 서비스되고 있는 통계목록에 대한 정보 및 관련 통계표 정보 - 국내통계 주제별, 국내통계 기관별, 광복이전통계(1908~1943), 대한민국통계연감, 작성중지통계, 지역통계(주제별), 지역통계(기관별), e-지방지표(주제별), 영문 KOSIS	JSON, SDMX
2) 통계자료 : KOSIS에서 공개되어 서비스되고 있는 약 7만여개 통계표에 대한 수치 데이터 및 메타정보(분류, 항목, 단위 등)	JSON, SDMX
3) 대용량 통계자료 : 상기 2) 통계자료와 동일한 정보를 제공하나 대량의 자료를 한번에 제공	XLS, SDMX
4) 통계설명 : 통계조사에 대한 상세 설명자료 - 조사명, 통계종류, 계속여부, 법적근거, 조사목적, 조사주기, 조사체계, 공표범위, 공표주기, 연락처	JSON, XML

* 통계자료와 대용량 통계자료는 국제통계 등 라이선스 제약에 해당되는 자료는 서비스 제외

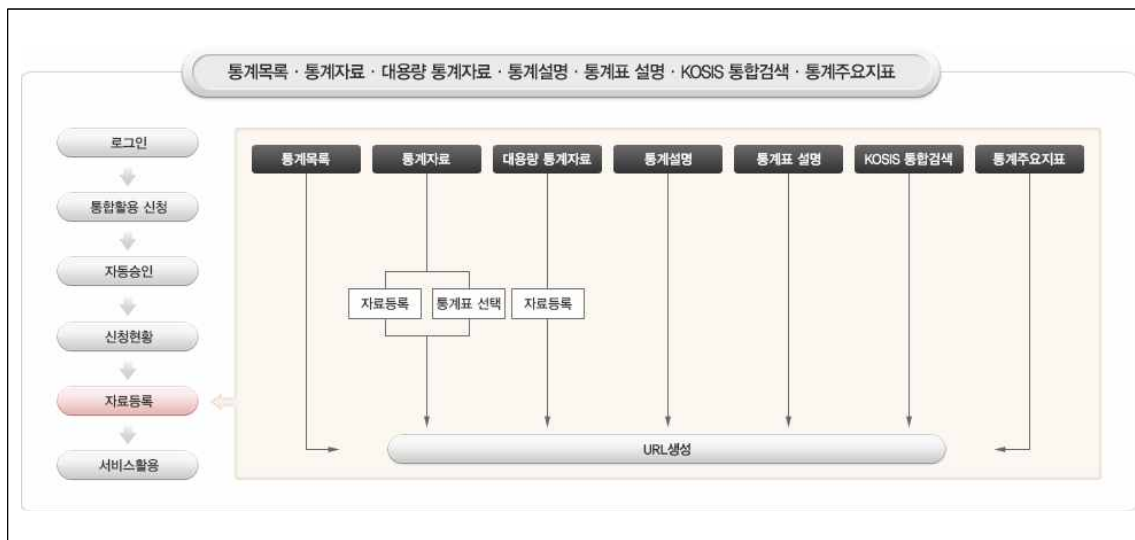
KOSIS 공유서비스를 활용하려면 국가통계포털(KOSIS) 회원으로 가입되어야 합니다. 기존 국가통계포털에 회원으로 가입된 사용자는 그대로 사용하시면 됩니다.

인증키는 회원 당 1개 발급되며, 1개 인증키로 모든 서비스를 이용하실 수 있습니다.

활용신청을 하시면 모든 서비스를 이용하실 수 있습니다.

활용신청에 대한 승인은 신청 후 자동 승인되어 바로 이용하실 수 있습니다.

서비스 대상별로 이용절차는 다음과 같습니다.

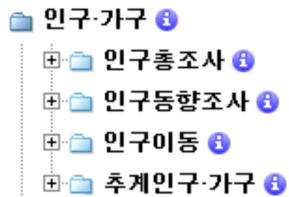


1.1 제공 콘텐츠

1.1.1 통계목록

■ 개요

통계표의 목록구성 정보 제공을 위한 OpenAPI입니다. 통계목록 단위로 호출하고 서비스뷰(주제별, 기관별 등 9가지)별로 상위목록의 정보와 연결된 통계표명을 제공합니다. 활용신청을 하면 자동으로 인증키가 발급되며 URL을 생성하여 자료를 활용할 수 있습니다.



■ 서비스 내용

서비스뷰	목록ID	목록명	기관코드	통계표ID
MT_ZTITLE	A1	인구총조사		
MT_ZTITLE	A2	인구동향조사		
MT_ZTITLE	A3	인구이동		
MT_ZTITLE	A4	추계인구·가구		
MT_ZTITLE	A5	생명표		
MT_ZTITLE	A6	주민등록인구통계		
MT_ZTITLE	A7	인구밀도		
MT_ZTITLE	A8	체류외국인통계		
MT_ZTITLE	110_11025	지방자치단체 외국인주민현황		

1.1.2 통계자료

■ 개요

통계표의 수치자료 및 메타정보(수록정보, 출처, 단위 등) 제공을 위한 OpenAPI입니다.

메타정보는 수치자료와 같이 호출할 수도 있습니다. 활용신청을 하면 자동으로 인증키가 발급되며 통계자료를 등록하여 활용하여야 합니다.

행정구역별(읍면동)	항목	2010
전국	총인구 (명)	48,580,293
	남자 (명)	24,167,098
	여자 (명)	24,413,195
	내국인-계 (명)	47,990,761
	내국인-남자 (명)	23,840,896

■ 서비스 내용

시도별	항목	2010	2011	2012
전국	주민등록인구 (명)	50,515,666	50,734,284	50,948,272

< 시계열 자료 >

시도별	주민등록인구 (명)
서울특별시	10,195,318
부산광역시	3,538,484
대구광역시	2,505,644
인천광역시	2,843,981
광주광역시	1,469,216
대전광역시	1,524,583
울산광역시	1,147,256
세종특별자치시	113,117
경기도	12,093,299
강원도	1,538,630
충청북도	1,565,628
충청남도	2,028,777
전라북도	1,873,341
전라남도	1,909,618
경상북도	2,698,353
경상남도	3,319,314
제주특별자치도	583,713

< 횡단면 자료 >

1.1.3 대용량 통계자료

■ 개요

통계표의 수치자료 및 메타정보 제공을 위한 OpenAPI로, 통계표 전체, 분류 전체(일부), 항목 전체(일부)를 선택적으로 요청합니다. 데이터량이 많은 특성 상 자료제공 형태가 SDMX 외 CSV가 추가되며, 활용신청을 하면 자동으로 인증키가 발급되며 대용량 통계자료를 등록하여 활용하여야 합니다.

행정구역별(읍면동)	항목	2010
전국	총인구 (명)	48,580,293
	남자 (명)	24,167,098
	여자 (명)	24,413,195
	내국인-계 (명)	47,990,761
	내국인-남자 (명)	23,840,896

■ 서비스 내용

행정구역별	연령별	항목	2038	2039	2040
전국	계	추계인구 (명)	51,470,244	51,291,401	51,091,352
전국	계	추계인구(남) (명)	25,474,360	25,375,247	25,264,681
전국	계	추계인구(여) (명)	25,995,884	25,916,154	25,826,671
서울특별시	계	추계인구 (명)	10,003,809	9,965,542	9,924,373
서울특별시	계	추계인구(남) (명)	4,842,375	4,822,073	4,800,386
서울특별시	계	추계인구(여) (명)	5,161,434	5,143,469	5,123,987
부산광역시	계	추계인구 (명)	3,059,995	3,037,918	3,014,946
부산광역시	계	추계인구(남) (명)	1,483,856	1,472,777	1,461,266
부산광역시	계	추계인구(여) (명)	1,576,139	1,565,141	1,553,680
대구광역시	계	추계인구 (명)	2,253,057	2,237,153	2,220,439
대구광역시	계	추계인구(남) (명)	1,102,911	1,094,925	1,086,547
대구광역시	계	추계인구(여) (명)	1,150,146	1,142,228	1,133,892
인천광역시	계	추계인구 (명)	3,052,428	3,045,109	3,036,476
인천광역시	계	추계인구(남) (명)	1,517,059	1,512,525	1,507,274
인천광역시	계	추계인구(여) (명)	1,535,369	1,532,584	1,529,202

1.1.4 통계설명

■ 개요

통계조사에 대한 설명자료 제공을 위한 OpenAPI입니다. 통계표 또는 통계조사 단위로 호출하면 통계조사에 대한 설명자료 정보가 API로 제공됩니다.

활용신청을 하면 자동으로 인증키가 발급되며 URL을 생성하여 자료를 활용할 수 있습니다.

통계표 정보

통계표명	인구, 가구 및 주택 - 읍면동
수록기간	5년 2010~2010
출처	통계청, 인구총조사 문의처 042-481-3738

■ 서비스 내용

항목	설명
조사명	가계동향조사
통계종류	지정통계 / 조사통계
계속여부	계속통계
법적근거	승인번호:10106, 승인일자:1962.12
조사목적	가구의 생활수준실태와 그 변동사항을 파악하기 위해서 가계의 수입과 지출을 조사하여 - 국민소비수준 변화의 측정 및 분석 - 소비자물가지수 편제에 필요한 가중치 모집단 자료 등 각종 경제, 사회정책에 필요한 자료를 제공 - 공공사업 시행으로 인한 세입자의 주거대책비 산정 및 국민주택 공급대상의 기준설정 자료
조사주기	분기
조사체계	표본가구→조사원→지방 통계사무소(품목코드부여,온라인입력)→통계청
공표단위	전국
공표주기	분기
연락처	통계청 사회통계국 복지통계과 (전화 : 042-481-2280)

1.1.5 메타자료

■ 개요

통계자료에 대한 메타자료 제공을 위한 OpenAPI입니다. 통계자료에 대한 통계표 명칭, 기관명칭, 수록정보, 분류/항목, 주석, 단위, 출처, 가중치, 자료갱신일에 대한 정보가 API로 제공됩니다.

■ 서비스 내용

통계표 국문명	통계표 영문명
총조사인구 총괄(읍면동/성/연령별)	Summary of Census Population(By administrative district/sex/age)

< 통계표 명칭 >

기관 국문명	기관 영문명
통계청	STATISTICS KOREA

< 기관 명칭 >

수록주기	시작 수록시점	종료 수록시점
5년	1925	2010

< 수록정보 >

분류 ID	분류 국문명	분류 영문명	자료코드 ID	자료코드 국문명	자료코드 영문명	상위 자료코드
A	행정구역별	By administrative divisions	00	전국	Whole country	

< 분류/항목 >

[illegible]

< 주석 >

단위 국문명	단위 영문명
포	Person

< 단위 >

조사명	통계표 담당부서	통계표 담당부서 전화번호
통계청, 인구총조사	통계청 조사관리국 인구총조사과	042-481-3738

< 출처 >

1.1.6 KOSIS통합검색

■ 개요

통계표의 목록구성 정보 제공을 위한 OpenAPI입니다. 통계목록 단위로 호출하고 서비스뷰 (주제별, 기관별 등 13가지)별로 상위목록의 정보와 연결된 통계표명을 제공합니다. 활용신청을 하면 자동으로 인증키가 발급되며 URL을 생성하여 자료를 활용할 수 있습니다.



■ 서비스 내용

결과변수	값
기관코드	101
기관명	통계청
통계표ID	DT_1J17041
통계표명	연도별 소비자물가 등락률
조사코드	1964001
조사명	소비자물가조사
KOSIS 목록구분	MT_ZTITLE
KOSIS 통계표 위치	물가·가계 > 물가 > 소비자물가조사(2015=100)
통계표 위치	C > C1 > C1_15
통계표 주요내용	지수종류 농생활물가지수 농산물및석유류제외지수 생활물가지수 식료품및에너지제외지수 신선식품지수 전년비 증지수
수록기간 시작일	1966
수록기간 종료일	2018
통계표 주석	- 생활물가지수 전체 소비자물가 460개 품목 중 채감물가를 설명하기 위해 구입빈도가 높고 지출비중이 높아 가격변동을 민감하게 느끼는 141개 품목을 별도로 집계 - 신선식품지수 전체 소비자물가 460개 품목 중 기상조건이나 계절에 따라 가격변동이 큰 생선류, 채소류, 과실류 50개 품목을 별도로 집계한 지수 - 농산물및석유류제외지수 전체 소비자물가 460개 품목 중에서 계절적으로 영향받는 농산물과 외부적 요인에 크게 영향받는 석유류 등 53개 품목을 제거하고 나머지 407개 품목을 별도로 집계한 지수, 물가변동의 장기적인 추세를 파악하기 위한 것으로 근원인플레이션 지수라 할 수 있음 - 식료품및에너지제외지수 전체 소비자물가 460개 품목 중에서 농산물과 석유류 외에도 축산물, 수산물, 가공식품, 전기, 지역난방비 등의 품목을 제외한 317개 품목으로 작성한 지수
추천통계표 여부	N
통계표 이동URL (KOSIS 목록으로 이동)	http://kosis.kr/statisticsList/statisticsListIndex.do?menuId=M_01_01&vwcd=MT_ZTITLE&parmTabId=M_01_01&parentId=C.1;C1.2;C1_15.3;
통계표 이동URL (KOSIS 통계표로 이동)	http://kosis.kr/statHtml/statHtml.do?orgId=101&tblId=DT_1J17042
검색결과 건수	2399
검색어명	물가

1.1.7 통계주요지표

■ 개요

통계표의 목록구성 정보 제공을 위한 OpenAPI입니다. 통계목록 단위로 호출하고 서비스뷰(주제별, 기관별 등 13가지)별로 상위목록의 정보와 연결된 통계표명을 제공합니다.

활용신청을 하면 자동으로 인증키가 발급되며 URL을 생성하여 자료를 활용할 수 있습니다.

■ 서비스 내용

지표ID	지표명	설명자료 제목	개념	선정방법	출처정보
168	노년부양비	노년부양비	생산가능인구(15~64세)에 대한 고령인구(65세 이상)의 비율로 생산가능인구(15~64세 인구) 100명이 부양해야 할 65세 이상 인구의 수를 표시	노년부양비 = [고령인구(65세 이상) / 생산가능인구(15~64세)] x 100	통계청, 장래인구추계

< 지표고유번호별/지표명별 설명자료조회 >

지표ID	지표명	단위	지역구분명	수록주기명	수록시작시점	수록종료시점	수록시점개수	종료시점+주기명
73	강수량	mm	전국	년	2000	2018	19	2018년
1068	시도별 강수량	mm	시도	년	2000	2018	19	2018년

< 지표명별 목록조회 >

지표ID	지표명	단위	지역구분명	수록주기명	수록시작시점	수록종료시점	수록시점개수	종료시점+주기명
100	주민등록인구	명	전국	년	1992	2018	27	2018년

< 고유번호별 목록조회 >

지표ID	지표명	수록주기	시점	항목	통계수치
295	비정규직 근로자 비중(남자)	M	201808	전국	26.3
295	비정규직 근로자 비중(남자)	M	201908	전국	29.4

< 고유번호별 지표 상세조회 >

지표ID	지표명	단위	지역구분명	수록주기명	수록시작시점	수록종료시점	수록시점개수	종료시점+주기명
93	추계인구(중위추계, 총인구)	명	전국	년	1960	2067	108	2067년
94	추계인구(중위추계, 남자)	명	전국	년	1960	2067	108	2067년

< 수록주기별 목록조회 >

지표ID	지표명	수록주기	시점	항목	통계수치
114	국내인구이동자수(시도내이동)	M	201907	전국	377,414
114	국내인구이동자수(시도내이동)	M	201908	전국	376,098

< 지표명별 상세조회 >

1.2 회원가입

- OpenAPI를 활용하기 위해서는 KOSIS회원이어야 합니다. KOSIS 회원가입 방법 및 절차는 아래와 같습니다.



KOSIS 공유서비스

로그인 **회원가입** 마이페이지

국가통계포털 공공데이터포털

소개 및 이용방법 서비스이용 개발가이드 커뮤니티

회원정보 > 로그인 회원가입 아이디/비밀번호 찾기

로그인 Home > 회원정보 > 로그인 인쇄

안녕하세요. KOSIS 공유서비스를 방문하여 주셔서 감사합니다.
국가통계포털 회원과 동일하게 로그인하시면 서비스를 이용하실 수 있습니다.

회원로그인

· 아이디

· 비밀번호

☐ 아이디저장

회원가입 / 아이디/비밀번호찾기

본 약관은 국가통계포털사이트(이하 '당 사이트')가 제공하는 모든 서비스(이하 '서비스')의 이용조건 및 절차, 이용자와 당 사이트의 권리, 의무, 책임사항과 기타 필요한 사항을 규정함을 목적으로 합니다.

제 2 조 (약관의 효력과 변경)

① 당 사이트는 이용자가 본 약관 내용에 동의하는 것을 조건으로 이용자에게 서비스를 제공할 것이며, 이용자가 본 약관의 내용에 동의하는 경우 당 사이트의 서비스 제공 행위 및 이용자의 서비스 사용 행위는 본 약관이 우선적으로 적용될 것입니다.

- 페이지 상단의 회원가입 또는 로그인 페이지의 회원가입 버튼을 클릭하면 KOSIS 회원가입페이지로 이동하며 제공된 양식에 따라 회원가입을 진행 할 수 있습니다.

1.3 인증키 발급 및 서비스 신청

1.3.1 활용신청

- 활용신청 양식을 활용용도와 사용목적 등을 작성한 뒤 ①신청버튼을 클릭합니다.
②의 [약관보기]를 클릭하면 통계정보 활용약관 팝업이 호출됩니다.

활용신청

Home > 활용신청

인쇄

(*)는 필수입력 항목입니다.

신청자명		전화번호	
서비스유형 (?)	REST	심의여부	자동승인
활용용도 (*)	<input type="radio"/> 앱개발 <input type="radio"/> 앱개발 <input type="radio"/> 기타	활용기간	인증키 만료일까지 활용 가능
상업적활용여부 (*)	<input type="radio"/> 예 <input type="radio"/> 아니오		
사용목적 (*)	<div> 사용목적은 반드시 작성해 주십시오. </div> <div> *사용목적은 불성실하게 작성시 활용신청 승인이 취소될 수 있습니다. 상세히 작성해 주시기 바랍니다. (2,000자 이내) </div>		
첨부파일	<div> <div>파일 선택</div> <div>선택된 파일 없음</div> </div> <div> *첨부파일 용량은 10Mb이하로 한 개의 파일만 첨부할 수 있습니다. [xls, doc, hwp, txt, gif, jpg, jpeg, png, bmp 파일만 가능합니다.] </div>		
약관동의	<div> <div>2</div> <div> <input type="checkbox"/> 통계정보 활용약관에 동의합니다. [약관보기] </div> </div>		

1

신청

1.3.2 신청현황

- 활용신청이 완료되면 신청현황 단계로 넘어가며, 기본정보에서는 사용자 인증키가 발급된 것을 확인이 가능하고, 서비스정보, 활용정보를 확인 할 수 있습니다. ①URL 생성 버튼을 누르면 URL 생성 단계로 넘어갑니다.

활용신청

Home > 활용신청

인쇄

기본정보

서비스유형	REST	활용기간	인증키 만료일까지 활용 가능
심의여부	자동승인	상업적활용유무	비활용
사용자 인증키	NGQxMjMxNWM2MmQxMmUj		

활용정보

활용용도	웹개발
활용목적	확인
사이트주소	http://
첨부파일	첨부된 파일이 없습니다.

1

신청정보 수정

1.4 에러메시지

- KOSIS OpenAPI 에러 메시지는 아래와 같습니다.

1.4.1 오류메시지 형식

```
<?xml version="1.0" encoding="UTF-8" ?>
- <error>
  <err>50</err>
  <errMsg>서버오류가 발생하였습니다.</errMsg>
</error>
```

1.4.2 오류메시지 유형

오류코드	오류메시지	조치방법
10	인증키 누락	인증키 확인
11	인증키 기간만료	인증키 기간 연장
20	필수요청변수 누락	필수요청변수 확인
21	잘못된 요청변수	요청변수 확인
30	조회결과 없음	조회조건 확인
31	조회결과 초과	호출건수 조정
50	서버오류	관리자에게 문의

2. OpenAPI 주요기능 및 활용방법

2.1 통계목록

- 통계표의 목록구성 정보 제공을 위한 OpenAPI입니다.

2.1.1 특징

- 통계표 서비스 목록으로서 레벨 형태로 구성
- 자료 제공 형태 : SDMX, JSON



2.1.2 서비스 활용

2.1.2.1 URL생성

■ 개발가이드 > 통계목록 > URL생성

목록구분, 목록조회, 결과유형을 입력한 뒤 ‘URL복사’, ‘결과값보기’ 중 원하는 서비스에 해당하는 버튼을 누르면 결과값을 제공받을 수 있습니다.

통계목록

[Home](#) >
[개발 가이드](#) >
[통계목록](#)

인쇄

통계목록 API를 사용하기 위한 요청변수, 출력정보, 샘플데이터를 제공합니다. 출력포맷으로 JSON과 SDMX를 제공합니다.

JSON

SDMX

URL 생성

개발가이드 다운로드

URL생성 조건 설정

목록구분

국내통계 주제별

목록조회

목록ID

A

목록명

인구

결과유형

SDMX

URL 복사

결과값보기

생성된 URL

https://kosis.kr/openapi/statisticsList.do?method=getList&apiKey=WNvHzDgyNDIVw2[REDACTED]&swCd=MT_ZTITLE&parentListId=A&format=sdmx&jsonV0=y&version=v2_1

데이터 보기

서비스뷰	목록ID	목록명	기관코드	통계표ID	추천통계표여부
MT_ZTITLE	A_4	인구총조사	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_7	주민등록인구현황	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_3	인구동향조사	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_6	장래인구추계	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_1	국내인구이동통계	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_9	출입국자및체류외국인통계	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_8	지방자치단체외국인주민현황	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_5	장래가구추계	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_2	국제인구이동통계	<해당없음>	<해당없음>	<해당없음>

결과값보기

<?xml version="1.0" encoding="UTF-8" ?><message:Structure xmlns:message="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/message" xmlns:structure="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/structure" xmlns:common="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/message .../schemas/SDMXmessage.xsd"><message:Header><message:ID>CATEGORY_MT_ZTITLE</message:ID><message:Test>false</message:Test><message:Prepared>2022-09-21T10:04:40</message:Prepared><message:Sender id="KOSIS" ><common:Name>통계청</common:Name></message:Sender><common:Name>국내 통계 주제</common:Name></message:Header><message:Structures><structure:OrganisationSchemas><structure:AgencyScheme id="AGENCIES" agencyID="SDMX" version="1.0"><common:Name xml:lang="en">SDMX Agency Scheme</common:Name><structure:Agency id="SDMX" ><common:Name xml:lang="en">SDMX</common:Name></structure:Agency><structure:Agency id="KOSIS" ><common:Name xml:lang="en">KOSIS</common:Name></structure:Agency><structure:AgencyScheme</structure:OrganisationSchemas><structure:CategorySchemas><structure:CategoryScheme id="A" agencyID="KOSIS" ><common:Name>인구</common:Name><structure:Category id="A_4" ><common:Name>인구총조사</common:Name></structure:Category><structure:Category id="A_7" ><common:Name>주민등록인구현황</common:Name></structure:Category><structure:Category id="A_3" ><common:Name>인구동향조사</common:Name></structure:Category><structure:Category id="A_6" ><common:Name>장래인구추계</common:Name></structure:Category><structure:Category id="A_1" ><common:Name>국내인구이동통계</common:Name></structure:Category><structure:Category id="A_9" ><common:Name>출입국자및체류외국인통계</common:Name></structure:Category><structure:Category id="A_8" ><common:Name>지방자치단체외국인주민현황</common:Name></structure:Category><structure:Category id="A_5" ><common:Name>장래가구추계</common:Name></structure:Category><structure:Category id="A_2" ><common:Name>국제인구이동통계</common:Name></structure:Category></message:Structures></message:Structure></div>

2.1.3 활용방법

2.1.3.1 JSON

■ 호출 URL : <http://kosis.kr/openapi/statisticsList.do>

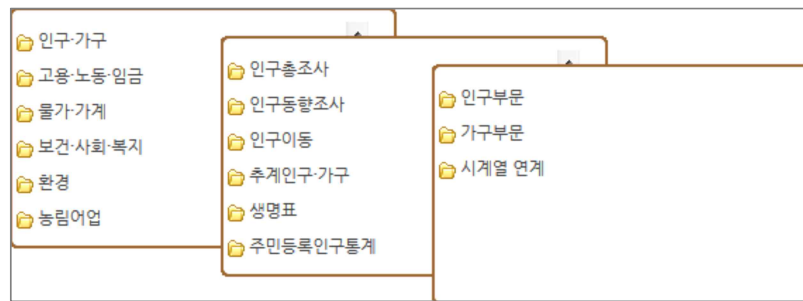
■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
vwCd	String	서비스뷰 코드 · MT_ZTITLE : 주제별 통계 · MT_OTITLE : 기관별 통계 · MT_CHOSUN_TITLE : 광복이전통계(1908~1943) · MT_HANKUK_TITLE : 대한민국통계연감 · MT_STOP_TITLE : 작성중지통계 · MT_ATITLE01 : 지역통계 (주제별) · MT_ATITLE02 : 지역통계 (기관별) · MT_GTITLE01 : e-지방지표(주제별) · MT_ETITLE : 영문KOSIS	필수
parentListId	String	시작목록 ID	필수
format	String	결과 유형(json, sdmx)	필수

■ 출력 변수

항목명(영문)	항목설명
VW_CD	서비스뷰ID
VW_NM	서비스뷰명
LIST_ID	목록ID
LIST_NM	목록명
ORG_ID	기관코드
TBL_ID	통계표ID
TBL_NM	통계표명
STAT_ID	통계조사ID
SEND_DE	최종갱신일
REC_TBL_SE	추천 통계표 여부

■ 예제 소스 결과



■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 통계목록을 생성하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all"
href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />
<link type="text/css" rel="stylesheet" media="all"
href="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide01/css/openTmp.css" />

<script type="text/javascript" language="JavaScript"
src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide01/js/dojo.js" ></script>
<script type="text/javascript" language="JavaScript"
src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide01/js/json.js" ></script>
<script type="text/javascript" language="JavaScript"
src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide01/js/ajax.js"></script>
<script type="text/javascript" language="JavaScript">

var mapData;
```

```
// window onload 되었을때 실행 함수
dojo.addOnLoad ( function() {

// 통계목록 리스트를 조회하기위해 함수를 호출한다.
getSubList("MT_ZTITLE", 0, "");
});

/*****
* 통계목록 리스트 조회 함수
* parameter : vwcd - 서비스뷰 코드 (통계목록구분)
* listLev - 목록 레벨
* parentId - 시작목록 Id
*****/
function getSubList(vwcd, listLev, parentId) {

// ajax 통신을 위한 파라미터를 변수에 담는다.
var paraObj = {
// 임의의 jsp 페이지를 호출함으로써 cross domain 제약을 우회할 수 있다. (devGuidePop.jsp 소스는 소스 하단에
제공)
url : "http://[개발자 홈페이지
주소]/devGuidePop.jsp?method=getList&key=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&vwcd=MT_Z
TITLE&parentId=" + parentId + "&type=json",
sync : true,
load : function(resObj, a, b) { mapData = resObj; },
error : function ( resObj, e ) { alert(dojo.toJson(resObj)); }
}

// ajax 통신 호출 함수
sendPost( paraObj );

// 통계목록 리스트를 화면에 출력하기 위한 함수
makeNode( Number(listLev) + 1 );
}

/*****
* 통계목록 리스트를 화면에 출력하기 위한 함수
* parameter : listLev - 목록 레벨
*****/
function makeNode(listLev) {

var nodeInfo="";

nodeInfo = nodeInfo+"<ul>";
```

```

for(var cnt=0; cnt<mapData.length; cnt++) {
    nodeInfo = nodeInfo + "<li>";
    if ( mapData[cnt].TBL_ID != null ) {
        nodeInfo = nodeInfo + "<img src='http://mgmk.kosis.kr/openapi_dev/devGuide/devGuide01/image/stats.gif'> <a
t      a      r      g      e      t      =      '      _      b      a      l      n      k      '
href=W\"http://kosis.kr/start.jsp?orgId="+mapData[cnt].ORG_ID+"&tblId="+mapData[cnt].TBL_ID+"&vw_cd="+map
Data[cnt].VW_CD+"&up_id="+mapData[cnt].UP_ID+W">"+mapData[cnt].TBL_NM+"</a>\" ;
    } else {
        nodeInfo = nodeInfo + "<img src='http://mgmk.kosis.kr/openapi_dev/devGuide/devGuide01/image/folder.gif'> <a
href=W\"javascript:getSubList(\""+mapData[cnt].VW_CD+\"\",          \"          +          listLev          +          \"\",
\""+mapData[cnt].LIST_ID+\" \");W\">"+mapData[cnt].LIST_NM+"</a>\" ;
    }
    nodeInfo = nodeInfo + "</li>";
}
nodeInfo = nodeInfo+"</ul>";

var r_node = document.getElementById("content"); //
var v_node = document.getElementById("depth"+listLev);
if( (typeof(v_node)!="undefined") && v_node!=null) {
    v_node.innerHTML = nodeInfo;
}
else {
    v_node = document.createElement("div");
    v_node.setAttribute("id", "depth"+listLev);
    v_node.className = "category0"+listLev;
    v_node.innerHTML = nodeInfo;
    r_node.appendChild(v_node);
}

var nodeCount = document.getElementsByTagName("div").length;

for( var cnt=(Number(listLev)+1); cnt< nodeCount; cnt++) {
    if(document.getElementById("depth"+cnt)!=null)
    r_node.removeChild(document.getElementById("depth"+cnt));
}
}

</script>
</head>
<body>
<div id="content" ></div>
</body>
</html>

```

```
// cross domain 제약을 우회하기 위한 jsp (devGuidePop.jsp)
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>

<c:set var="method" ><%out.println(request.getParameter("method") == null ? "" : (request.getParameter("method"))); %></c:set>
<c:set var="key" ><%out.println(request.getParameter("key") == null ? "" : (request.getParameter("key"))); %></c:set>
<c:set var="vwcd" ><%out.println(request.getParameter("vwcd") == null ? "" : (request.getParameter("vwcd"))); %></c:set>
<c:set var="parentld"><%out.println(request.getParameter("parentld") == null ? "" : (request.getParameter("parentld")));%></c:set>
<c:set var="type" ><%out.println(request.getParameter("type") == null ? "" : (request.getParameter("type"))); %></c:set>

< c : i m p o r t
url="http://mgmk.kosis.kr/openapi_dev/Expt/statisticsList.do?method=${method}&apiKey=${key}&vwCd=${vwcd}&parentListId=${parentld}&format=${type}" charEncoding="utf-8"/>
```

■ 예제 소스(R)

```
library(httr) # api, 크롤링 등에 사용
library(rvest) # HTML처리
library(jsonlite) # JSON 읽어올때 사용

rm(list=ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')

baseurl <- 'https://kosis.kr/openapi/statisticsList.do'
v_vwCd = 'MT_ZTITLE' # 서비스뷰코드
v_parentListId = 'A' # 시작목록ID

url_page <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'json',
      vwCd = v_vwCd,
      parentListId = v_parentListId,
      jsonVD = 'Y',
      apiKey = v_apiKey %>% I()
    )
  )
```

```
url_page %>% content(as = 'text', encoding = 'UTF-8') %>% fromJSON() -> json

df_lists <- data.frame(서비스뷰 = c(json$VW_CD))

# 통계표가 연결된 목록과 중간목록을 구분하여 처리
if (is.null(json$LIST_ID) == FALSE) {
  df_lists <- cbind(df_lists, 목록ID = c(json$LIST_ID))
  df_lists <- cbind(df_lists, 목록명 = c(json$LIST_NM))
} else {
  df_lists <- cbind(df_lists, 기관코드 = c(json$ORG_ID))
  df_lists <- cbind(df_lists, 통계표ID = c(json$TBL_ID))
  df_lists <- cbind(df_lists, 통계표명 = c(json$TBL_NM))
}

View(df_lists)
```

■ 예제 소스(Python)

```
import json
from urllib.request import urlopen # python 3.x 버전에서 사용 (2.x 버전이라면 from urllib import urlopen)
from PyQt5 import QtWidgets
from PyQt5.QtWidgets import *
import functools as fc
import sys

#클릭한 목록에 대한 하위 목록 생성
class NewWindow(QtWidgets.QMainWindow):
    def __init__(self, parent=None):
        super(NewWindow, self).__init__(parent)
        centralWidget = QWidget()
        self.setCentralWidget(centralWidget)
        self.setGeometry(300, 300, 500, 500)

        Setting(self, List_Id)

#최상위 목록 생성
class MyWindow(QtWidgets.QMainWindow, QPushButton):
    def __init__(self):
        super(MyWindow, self).__init__()
        centralWidget = QWidget()
        self.setCentralWidget(centralWidget)
        self.setGeometry(200, 200, 500, 500)
        self.setStyleSheet("background-color: white")

        Setting(self, 'A')

#목록 셋팅 함수
def Setting(self, parentId):
    #url을 통해 json 데이터 가져오기
    with urlopen(
        "https://kosis.kr/openapi/statisticsList.do?method=getList&apiKey=ZiZiOTl3MiRiNmU1YzdhZTMwOWRi"
```



```
NjgxN2MzNDgwNmY=&vwCd=MT_ZTITLE&parentListId="+parentId+"&format=json&jsonVD=Y") as url:
```

```
    json_file = url.read()
```

```
py_json = json.loads(json_file.decode('utf-8'))
```

```
#하위 목록이 있다면 LinkButton, 하위 목록이 없다면 label로 생성
```

```
for i, v in enumerate(py_json):
```

```
    if 'LIST_NM' in v:
```

```
        btn = QCommandLinkButton(v['LIST_NM'], self)
```

```
        btn.setStyleSheet("Text-align: left;"
```

```
                           "border: none;"
```

```
                           )
```

```
        btn.setGeometry(100, 50 * i, 500, 40)
```

```
        btn.clicked.connect(fc.partial(Action, self, v['LIST_ID']))
```

```
    else:
```

```
        lbl = QLabel(v['TBL_NM'], self)
```

```
        lbl.setGeometry(100, 50 * i, 500, 40)
```

```
def Action(self, check):
```

```
    global List_Id
```

```
    List_Id = check
```

```
    NewWindow(self).show()
```

```
if __name__ == "__main__":
```

```
    app = QtWidgets.QApplication(sys.argv)
```

```
    window = MyWindow()
```

```
    window.show()
```

```
    sys.exit(app.exec_())
```

2.1.3.2 SDMX(Category)

■ 호출 URL : <http://kosis.kr/openapi/statisticsList.do>

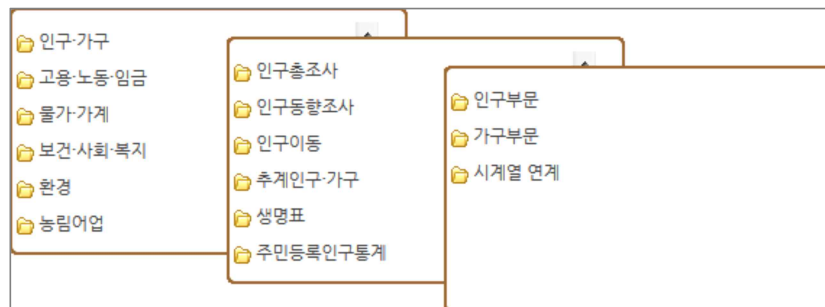
■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
vwCd	String	서비스뷰 코드 · MT_ZTITLE : 주제별 통계 · MT_OTITLE : 기관별 통계 · MT_CHOSUN_TITLE : 광복이전통계(1908~1943) · MT_HANKUK_TITLE : 대한민국통계연감 · MT_STOP_TITLE : 작성중지통계 · MT_ATITLE01 : 지역통계 (주제별) · MT_ATITLE02 : 지역통계 (기관별) · MT_GTITLE01 : e-지방지표(주제별) · MT_ETITLE : 영문KOSIS	필수
parentListId	String	시작목록 ID	필수
format	String	결과 유형(json, sdmx)	필수
version	String	결과값 구분	생략시 구버전으로 데이터 출력

■ 출력 변수

항목명(영문)					항목설명
Header	Id				서비스뷰ID
	Name				서비스뷰명
	Prepared				전송시간
	Sender		Id		전송기관
			Name		전송기관명
Structures	Category Schemes	Category Scheme	Id		상위목록 ID
			Name		상위목록 명
			Category	Id	목록ID
			Category	Name	목록명칭
			Description	Description	기관코드_ 통계표ID
			StatId	StatId	통계조사ID
			SendDe	SendDe	최종갱신일
			RecTblSe	RecTblSe	추천 통계표 여부

■ 예제 소스 결과



■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 통계목록을 생성하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all"
href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />
<link type="text/css" rel="stylesheet" media="all"
href="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide01/css/openTmp.css" />

<script type="text/javascript" language="JavaScript"
src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.6.1.min.js"></script>
<script type="text/javascript" language="JavaScript">

var mapData;

// window onload 되었을때 실행 함수
$(document).ready(function(){
```

```
// 통계목록 리스트를 조회하기위해 함수를 호출한다.
getSubList("MT_ZTITLE", 0, "");
});

/*****
* 통계목록 리스트 조회 함수
* parameter : vwcd - 서비스뷰 코드 (통계목록구분)
* listLev - 목록 레벨
* parentId - 시작목록 Id
*****/
function getSubList(vwcd, listLev, parentId) {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsList.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&vwCd=MT_ZTITLE&parentListId=" + parentId + "&format=sdmx&version=v2_1",
async : true,
dataType: "xml",
success:function(object)
{

// ajax 통신이 성공하였을 때 통계목록 리스트를 화면에 출력하기 위한 함수
var data = object.documentElement.childNodes[1].childNodes[0].childNodes[0].childNodes;
var nodeInfo="";

nodeInfo = nodeInfo+"<ul>";

listLev = Number(listLev) + 1 ;

for(var cnt=1; cnt < data.length; cnt++) {
nodeInfo = nodeInfo + "<li>";

if ( data[cnt].childNodes.length== 2 ) {

nodeInfo = nodeInfo + "<img src='image/stats.gif'> <a target='_blank' href=W'http://kosis.kr/start.jsp?orgId=" +
data[cnt].attributes[0].value.split("_")[0] + "&tblId=" + data[cnt].attributes[0].value.split("_")[1] + "_" +
data[cnt].attributes[0].value.split("_")[2] + "W">" + data[cnt].childNodes[0].childNodes[0].data + "</a>" ;
} else {
nodeInfo = nodeInfo + "<img src='image/folder.gif'> <a href=W'javascript:getSubList('MT_ZTITLE', '" + listLev +
"', '" + data[cnt].attributes[0].value + "');W">" + data[cnt].childNodes[0].childNodes[0].data + "</a>" ;
}
}

nodeInfo = nodeInfo + "</li>";
}
```

```

}
nodeInfo = nodeInfo+"</ul>";

var r_node = document.getElementById("content"); //
var v_node = document.getElementById("depth"+listLev);
if( (typeof(v_node)!="undefined") && v_node!=null) {
v_node.innerHTML = nodeInfo;
}
else {
v_node = document.createElement("div");
v_node.setAttribute("id", "depth"+listLev);
v_node.className = "category0"+listLev;
v_node.innerHTML = nodeInfo;
r_node.appendChild(v_node);
}

var nodeCount = document.getElementsByTagName("div").length;

for( var cnt=(Number(listLev)+1); cnt< nodeCount; cnt++) {
if(document.getElementById("depth"+cnt)!=null)
r_node.removeChild(document.getElementById("depth"+cnt));
}
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}
</script>
</head>
<body>
<div id="content" ></div>
</body>
</html>

```

■ 예제 소스(R)

```

library(httr)
library(rvest)
library(XML)
library(stringr)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )

```

```
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')

baseurl <- 'https://kosis.kr/openapi/statisticsList.do'
v_vwCd = 'MT_ZTITLE'                # 서비스뷰코드
v_parentListId = 'A'                # 시작목록ID

url_page <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'sdmx',
      vwCd = v_vwCd,
      parentListId = v_parentListId,
      apiKey = v_apiKey %>% I()
    )
  )

docParse <- xmlParse(url_page)

docList <-
  xmlToList(docParse)$Structures$CategorySchemes$CategoryScheme

df_lists <- data.frame()

for (i in 1:length(docList)) {
  v_len = length(docList[i]$Category)

  if (length(docList[i]$Category) == 2) {
    df_lists <-
      rbind(df_lists, cbind(
        목록ID = c(docList[i]$Category$.attrs),
        목록명 = c(docList[i]$Category$Name)
      ))
  } else if (v_len == 3) {
    tbl_info <- unlist(docList[2]$Category$.attrs)
    tbl_infos <- unlist(strsplit(tbl_info, 'WW_'))
    v_org_id <- tbl_infos[1]
    v_tbl_id <-
      substr(
        tbl_info,
        str_length(v_org_id) + 2,
        str_length(tbl_info) - str_length(v_org_id) + 3
      )
  }
}
```

```

    )

df_lists <-
  rbind(df_lists, cbind(
    기관코드 = v_org_id,
    통계표ID = v_tbl_id,
    통계표명 = c(docList[i]$Category$Name)
  ))
}
}

View(df_lists)

```

■ 예제 소스(Python)

```

from PyQt5 import QtWidgets
from PyQt5.QtWidgets import *
import functools as fc
import sys
import requests
from bs4 import BeautifulSoup

#클릭한 목록에 대한 하위 목록 생성
class NewWindow(QtWidgets.QMainWindow):
    def __init__(self, parent=None):
        super(NewWindow, self).__init__(parent)
        centralWidget = QWidget()
        self.setCentralWidget(centralWidget)
        self.setGeometry(300, 300, 500, 500)

        Setting(self, List_Id)

#최상위 목록 생성
class MyWindow(QtWidgets.QMainWindow, QPushButton):
    def __init__(self):
        super(MyWindow, self).__init__()
        centralWidget = QWidget()
        self.setCentralWidget(centralWidget)
        self.setGeometry(200, 200, 500, 500)
        self.setStyleSheet("background-color: white")

        Setting(self, 'A')

#목록 셋팅 함수
def Setting(self, parentId):

```

```

open_url
=
'https://kosis.kr/openapi/statisticsList.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&vwCd=MT_ZTITLE&parentListId='+parentId+'&format=sdmx&jsonVD=Y&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

data = soup.find_all('structure:category')

#window title 설정
self.setWindowTitle(soup.find('structure:categoryscheme').find('common:name').text)

#하위 목록이 있다면 LinkButton, 하위 목록이 없다면 label로 생성
for i, item in enumerate(data):
    if item.find('common:description') == None:
        btn = QCommandLinkButton(item.find('common:name').text, self)
        btn.setStyleSheet("Text-align: left;"
                           "border: none;"
                           )
        btn.setGeometry(100, 50 * i, 500, 40)
        btn.clicked.connect(fc.partial(Action, self, item.get('id')))
    else:
        lbl = QLabel(item.find('common:name').text, self)
        lbl.setGeometry(100, 50 * i, 500, 40)

def Action(self, check):
    global List_Id
    List_Id = check

    NewWindow(self).show()

if __name__ == "__main__":
    app = QtWidgets.QApplication(sys.argv)
    window = MyWindow()
    window.show()
    sys.exit(app.exec_())

```


2.2 통계자료

- 통계표의 수치자료 및 메타정보(수록정보, 출처, 단위 등) 제공을 위한 OpenAPI입니다.

2.2.1 특징

- 통계표의 수치자료를 시계열(단일계열, 여러시점) 또는 횡단면(다중계열, 단일시점)으로 제공
- 통계표의 수록정보, 분류/항목, 출처, 단위 등 메타정보 제공
- 자료 제공형태 : SDMX (DSD, Generic, StructureSpecific), JSON, XML
- URL 선택 방식은 별도의 자료등록없이, 바로 필요한 통계표/항목/분류를 선택하고, URL에 테이블 ID/항목/분류 파라메타를 제공해, 이 값을 이용자가 동적으로 변경가능 함 으로서 이용할 자료를 미리 등록하는 과정이 불필요한 방식입니다.(등록한 자료의 통계표 메타정보가 변경될 경우 확인 불가능)
- URL 자료등록 방식은 생성내역의 통계표 및 URL이 저장되어 등록된 자료의 통계표 메타정보가 변경될 경우 이용자가 확인 가능하며 테이블/항목/분류 등 정보값을 이용자가 동적으로 변경할 수 없는 방식입니다.

2.2.2 서비스 활용

2.2.2.1 URL 생성

- 개발가이드 > 통계자료 > URL생성
 - ‘작성기관’, ‘통계조사명’, ‘통계표명’ 등 을 입력하여 사용하고자 하는 자료를 조회한 뒤 조회결과에서 생성할 통계표 ①선택하면 하단에 ③URL생성 조건 설정이 출력됩니다.
 - ②통계표조회 버튼을 클릭하면 해당 자료의 통계표를 볼 수 있습니다.
 - ③URL생성 조건 설정에서 ‘분류/항목선택’, 출력형태 설정, 조회기간 설정을 입력 후 ④URL보기 버튼을 클릭하면 생성된 URL이 조회되고, ⑤URL복사 버튼을 누르면 생성된 URL이 클립보드에 복사됩니다.
 - URL길이를 줄이기 위해 항목/분류 파라메타에 간판 키워드 적용
 - 항목과 분류별 전체선택시 ‘all’ -> 예) itmID = all, objL1 = all - 레벨로 구성된 분류의 경우 하위레벨 전체 선택시 ‘*’ -> 예) 행정구역별에서 서울(11)의 하위레벨 전체를 포함하고자 할 경우 objL1 = 11*
 - 항목, 분류값 추가시 ‘+’ -> 예) 행정구역별에서 서울(11)과 부산(21)을 선택할 경우 objL1 = 11+ 21
 - * 시스템 부하를 줄이기 위해 통계자료 요청은 4만셀 이하로 제한

통계자료
Home > 개발 가이드 > 통계자료
인쇄

통계자료 API를 이용하는 요청번호, 출력번호, 샘플데이터를 제공하며 자료등록방법 (자료등록 후 이용해야하며 URL 변경 불가)과 통계표선택 방법(분류와 항목정보가 파라미터로 URL에 포함되어 파라미터를 변경해가며 이용 가능) 2가지 방법으로 활용 가능합니다. 출력포맷으로 JSON과 SDMX를 제공합니다.

JSON
SDMX
URL 생성
자료등록 URL 생성

작성기관: 통계청
통계표명:
분류:
분류값/항목:

통계조사명: 인구총조사
통계표ID: DT_1B08024
주기: 전체

검색

기관명	통계표명	통계표ID	통계조사명	수집기간	선택	통계표조회
통계청 (101)	인구밀도(인구주택총조사기준)	DT_1B08024	인구총조사	5년 1966~2021	선택	

URL생성 조건 설정

통계표명: 인구밀도(인구주택총조사기준)
통계표ID: DT_1B08024

분류
항목

정정구분
개별
선택

인구밀도

출력형태 설정

데이터 포맷: ☒ JSON ☐ SDMX

사용자 인증키: NwVhZDgyNDVhZTJhZTk2MmE0OGVhYjY0YzgzMTZwZw

조회기간 설정: ☐ 기간설정 ☒ 최근시점기준

최근시점부터: 주기 5년 최근 3 개 자료

수집시점간격:

URL생성 URL

URL보기

URL복사

```

https://kosis.kr/openapi/Param/statisticsParameterData.do?
method=getList&apiKey=NwVhZDgyNDVhZTJhZTk2MmE0OGVhYjY0YzgzMTZwZw&timId=T7+8objL1=00+8objL2=8objL3=8objL4=8objL5=8objL6=8objL7=8objL8=&format=json&sonD=73prdsE=73newEstPrdCnt=33orgId=1018161d=DT_1B08024

```


2.2.2.2.2 등록된 자료

■ 개발가이드 > 통계자료 > 자료등록 URL생성 > 등록된 자료

자료등록의 통계표 등록을 마치고 등록된 자료 탭을 클릭하면, 이용자가 등록한 자료들의 목록이 나타나며, 등록된 자료 중 URL생성을 원하는 자료의 ①URL생성 버튼을 누른다.
②통계표조회 버튼을 클릭하면 해당 자료의 통계표를 볼 수 있습니다.

통계자료
Home > 개발 가이드 > 통계자료
인쇄

통계자료 API를 이용하는 요청변수, 출력변수, 샘플데이터를 제공하며 자료등록방법 (자료등록 후 이용해야하며 URL 변경 불가)과 통계표선택 방법(분류와 항목정보가 파라미터로 URL에 포함되어 파라미터를 변경해가며 이용 가능) 2가지 방법으로 활용 가능합니다. 출력포맷으로 JSON과 SDMX를 제공합니다.

JSON
SDMX
URL 생성
자료등록 URL 생성

·자료 등록 ·등록된 자료

작성기관
통계표명
분류
분류값/항목

통계조사명
통계표ID
주기
전체

검색

기관명	통계표명	통계표ID	통계조사명	수록기간	신청일자	통계표조회	URL생성
통계청 (101)	작성기관별/담당정보별 전체 및 통계...	DT_1011301	통계인력및예산조사	2년 1990~2020	2022-09-2 19:39:46	②	① URL생성
통계청 (101)	인구총조사 인구(시도/시/군/구)	INH_1IN1503_01	인구총조사	년 2015~2021	2022-09-15 16:00:01		URL생성
통계청 (101)	순이동인구(시도/시/군/구)	INH_1B26001_A021	국내인구이동통계	월,분기,년 1970.01~2022.07	2022-08-26 16:20:44		URL생성

■ 개발가이드 > 통계자료 > 자료등록 URL생성 > 등록된 자료 > 삭제

①버튼을 클릭한 후 하단에 조회되는 사용자 생성 URL 목록을 ②삭제할 수 있습니다.

통계자료
Home > 개발 가이드 > 통계자료
인쇄

통계자료 API를 이용하는 요청변수, 출력변수, 샘플데이터를 제공하며 통계표선택(URL 생성) 방법 (분류와 항목정보가 파라미터로 URL에 포함되어 파라미터를 변경해가며 이용 가능) 과 자료등록(자료등록 URL 생성) 방법 (자료등록 후 이용해야하며 URL 변경 불가) 2가지 방법으로 활용 가능합니다. 출력포맷으로 JSON과 SDMX를 제공합니다.

JSON
SDMX
URL 생성
자료등록 URL 생성

·자료 등록 ·등록된 자료

작성기관
통계표명
분류
분류값/항목

통계조사명
통계표ID
주기
전체

검색

기관명	통계표명	통계표ID	통계조사명	수록기간	신청일자	통계표조회	URL생성
통계청 (101)	GDP 대비 정부 재원 교육비 비율(APEC...	DT_2KAAB05_APEC	국제통계연감	년 1998~2020	2022-10-05 15:32:00		URL생성

사용자 생성 URL 목록

활용자료명	URL 상세 정보	삭제	URL보기
GDP 대비 정부 재원 교육비 비율 (APEC)	학교전 교육 * 아시아	② 삭제	URL보기

2.2.2.2.3 URL생성

■ 개발가이드 > 통계자료 > 자료등록 URL생성 > 등록된 자료 > URL 생성

URL생성 단계에서는 URL생성 조건 설정의 '활용 자료명', '분류/항목선택'을 입력 후 ①URL생성 버튼을 누르면 URL생성 상세조건 화면으로 이동 후 URL이 하단에 생성됩니다. URL생성 상세조건 화면에서 상세설정 후 ②URL보기, 결과값보기 버튼을 클릭하여 페이지 하단에서 결과를 확인 할 수 있고, ③URL복사 버튼을 누르면 생성된 URL이 클립보드에 복사됩니다. 또한 ④목록 버튼을 누르면 등록된 자료화면으로 돌아갑니다.

통계자료
Home > 개발 가이드 > 통계자료
인쇄

통계자료 API를 이용하는 요청변수, 출력변수, 샘플데이터를 제공하며 자료등록방법 (자료등록 후 이용해야하며 URL 변경 불가)과 통계표선택 방법(분류와 항목정보가 파라미터로 URL에 포함되어 파라미터를 변경해가며 이용 가능) 2가지 방법으로 활용 가능합니다. 출력포맷으로 JSON과 SDMX를 제공합니다.

JSON
SDMX
URL 생성
자료등록 URL 생성

·자료 등록 ·등록된 자료

URL생성

URL생성 조건 설정

통계표명	인구총조사 인구(시도/시/군/구)	통계표 ID	INH_1IN1503_01
활용 자료명	인구총조사 인구(시도/시/군/구)		
분류/항목선택	<div> 분류 항목 전체선택 해제 </div> <div> 행정구역별(읍면동) <input checked="" type="radio"/> 개별 <input type="radio"/> 레벨별 1 / 2 </div>		<input checked="" type="checkbox"/> 총인구(경)
	<div> URL생성 취소 </div>		

통계자료 API를 이용하는 요청변수, 출력변수, 샘플데이터를 제공하며 자료등록 방법 (자료등록 후 이동해야하며 URL 변경 불가)과 통계표선택 방법(분류와 항목정보가 파라미터로 URL에 포함되어 파라미터를 변경해야하며 이동 가능) 2가지 방법으로 활용 가능합니다. 출력포맷으로 JSON과 SDMX를 제공합니다.

JSON
SDMX
URL 생성
자료등록 URL 생성

·자료 등록 ·등록된 자료

URL생성

URL생성 조건 설정

통계표명	인구총조사 인구(시도/시/군/구)	통계표 ID	INH_1IN1503_01
------	--------------------	--------	----------------

URL생성 상세조건 정보

출력형태 설정

데이터 포맷 ☐ JSON ☒ SDMX

출력 형태 ☐ DSD(자료구조정의) ☐ Generic(일반 수치자료) ☒ StructureSpecific(구조화된 수치자료)

조회기간 설정 ☐ 기간설정 ☒ 최근시점기준

최근시점부터 주기 년 최근 3 개 자료

수색시점간격 최근시점 조정에 사용할 수 있는 항목으로 현재 시점부터 사용할 기준시점까지의 간격을 입력
ex) 2019, 2017, 2015 등 2개 시점 간격으로 추출시 [2] 입력

활용자료명	URL 상세 정보	조회주요
인구총조사 인구(시도/시/군/구)	출인구(명) = 전국	출단면

2 URL보기 / 결과값보기

URL보기

결과값보기

생성된 URL

```
https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=NHVnZDgyNDVhZTJhZTRhZDZmE0OGVhYjY0Yzg1MzowY2Q=&format=sdmx&jsonV0=Y&userStatsId=bang3178/101/INH_1IN1503_01/2/2/20220321100954&type=StructureSpecific&probe=Y&newEstPrdCnt=3&version=v2_1
```

3 URL복사

데이터 보기

인구총조사 인구(시도/시/군/구)(2021)

행정구역별 (출면동)	출인구(명)
전국	51738071

결과값보기

```
<?xml version='1.0' encoding='UTF-8' standalone='yes' ?><message:StructureSpecificTimeSeriesData
xmlns:exr='urn:sdmx:org.sdmx.infomodel.datastructure.DataStructure=101_INH_1IN1503_01:ObsLevelDim:TIME_PERIOD'
xmlns:message='http://www.sdmx.org/resources/sdmx/schemas/v2_1/message'
xmlns:data='http://www.sdmx.org/resources/sdmx/schemas/v2_1/data/structurespecific'
xmlns:common='http://www.sdmx.org/resources/sdmx/schemas/v2_1/common' xmlns:xs='http://www.w3.org/2001/XMLSchema-instance'
xmlns:schemaLocation='http://www.sdmx.org/resources/sdmx/schemas/v2_1/message ../../schemas/SDMXMessage.xsd
urn:sdmx:org.sdmx.infomodel.datastructure.DataStructure=101_INH_1IN1503_01:ObsLevelDim:TIME_PERIOD exr_ng_ts.xsd'><message:Header>
<message:ID>101_INH_1IN1503_01</message:ID><message:Name>인구총조사 인구(시도/시/군/구) (년 2015-2021)</message:Name><message:Test>false</message:Test>
<message:Prepared>2022-03-21 10:10:07</message:Prepared><message:Sender id='KOSIS'><message:Contact><message:Department>통계청</message:Department>
<message:Telephone>042-481-9758</message:Telephone></message:Contact></message:Sender><message:Structure structureID='101_INH_1IN1503_01'
dimensionAtObservation='TIME_PERIOD' /><message:Source>KOSIS 통계청, 「인구총조사」</message:Source></message:Header><message:DataSet
data:structureRef='101_INH_1IN1503_01' data:dataScope='DataStructure'><Series FREQ='A' ITEM='T00' GLA='00'><Obs OBS_VALUE='51779203' TIME_PERIOD='2019' />
</Series></message:DataSet></message:StructureSpecificTimeSeriesData>
```

4 목록

2.2.3 활용방법

2.2.3.1 JSON

- 호출 URL : <http://kosis.kr/openapi/statisticsData.do>
- 입력 변수

● 자료등록 방법

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
userStatsId		String	사용자 등록 통계표	필수
prdSe		String	수록주기	필수
시점기준	startPrdDe	String	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료 기준	newEstPrd Cnt	String	최근수록시점 개수	
	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수

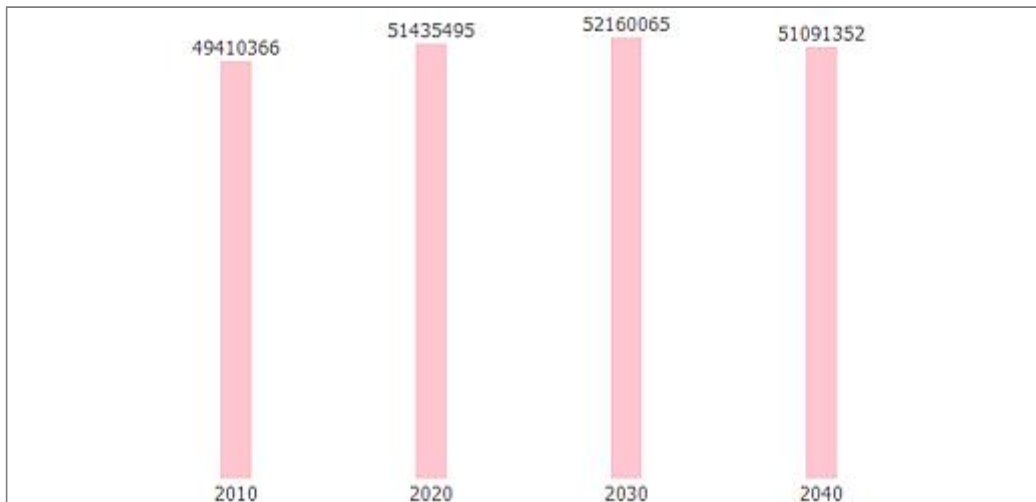
● 통계표선택 방법

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
orgId		String	기관 ID	필수
tblId		String	통계표 ID	필수
objL1		String	분류1(첫번째 분류코드)	필수
objL2 ~ objL8		String	분류2(두번째 분류코드) ~ 분류8(여덟째 분류코드)	선택
itmId		String	항목	필수
prdSe		String	수록주기	필수
시점기준	startPrdDe	String	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료 기준	newEstPrd Cnt	String	최근수록시점 개수	
	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수

■ 출력 변수

항목명(영문)	항목설명	비고
ORG_ID	기관코드	
TBL_ID	통계표ID	
TBL_NM	통계표명	
C1 ~ C8	분류값 ID1 ~ 분류값 ID8	2~8 분류값은 없을 경우 생략
C1_OBJ_NM ~ C8_OBJ_NM	분류명1 ~ 분류명8	
C1_OBJ_NM_ENG ~ C8_OBJ_NM_ENG	분류 영문명1 ~ 분류 영문명8	
C1_NM ~ C8_NM	분류값 명1 ~ 분류값 명8	
C1_NM_ENG ~ C8_NM_ENG	분류값 영문명1 ~ 분류값 영문명8	
ITM_ID	항목 ID	
ITM_NM	항목명	
ITM_NM_ENG	항목영문명	
UNIT_ID	단위ID	
UNIT_NM	단위명	
UNIT_NM_ENG	단위영문명	
PRD_SE	수록주기	
PRD_DE	수록시점	
DT	수치값	

■ 예제 소스 결과(막대차트)



■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 막대차트를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
```


호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.

```
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<xml:namespace prefix="v"/>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide02/js/dojo.js" ></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide02/js/json.js" ></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide02/js/ajax.js"></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.6.1.min.js"></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide02/js/jquery.SimpleChart.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
dojo.addOnLoad ( function() {

getList();
});

/*****
* 통계자료 조회 함수
*****/
function getList() {

// ajax 통신을 위한 파라미터를 변수에 담는다.
var paraObj = {
// 임의의 jsp 페이지를 호출함으로써 cross domain 제약을 우회할 수 있다. (devGuidePop.jsp 소스는 소스 하단에 제공)
url : "http://[개발자 홈페이지 주소]/devGuidePop.jsp?method=getList&key=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=json&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3",
sync : true,
load : function(mapData, a, b) {

// 차트에 보여주기위해 값을 담고있을 배열변수를 선언
var aData = new Array();

// 데이터를 배열변수에 적재
for(var i=0;i<mapData.length;i++) {
```

```

aData[i] = [mapData[i].DT.replace(/,/gi,''), mapData[i].PRD_DE, 'pink'];
}

// 차트를 그리기 위한 옵션을 정의
var options = { 'BarSize': '20px', 'BarSpace': '2px', 'type' : 'horizontal', 'Font': '2px' }

// 차트를 화면에 출력
$("#chart2").SimpleChart(aData,options);
},
error : function ( resObj, e ) {

alert(dojotoJson(resObj));
}
}

// ajax 통신 호출 함수
sendPost( paraObj );
}
</script>
</head>
<body>
<div id="chart2" style="padding-top:30px;"></div>
</body>
</html>

// cross domain 제약을 우회하기 위한 jsp (devGuidePop.jsp)
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>

<c:set var="method" ><%out.println(request.getParameter("method") == null ? "" : (request.getParameter("method"))); %></c:set>
<c:set var="key" ><%out.println(request.getParameter("key") == null ? "" : (request.getParameter("key"))); %></c:set>
<c:set var="userStatsId"><%out.println(request.getParameter("userStatsId") == null ? "" : (request.getParameter("userStatsId"))); %></c:set>
<c:set var="newEstPrdCnt"><%out.println(request.getParameter("newEstPrdCnt")== null ? "" : (request.getParameter("newEstPrdCnt"))); %></c:set>
<c:set var="prdSe" ><%out.println(request.getParameter("prdSe") == null ? "" : (request.getParameter("prdSe"))); %></c:set>
<c:set var="format" ><%out.println(request.getParameter("format") == null ? "" : (request.getParameter("format"))); %></c:set>

<c:import url="http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=${method}&apiKey=${key}&format=${format}&userStatsId=${userStatsId}&prdSe=${prdSe}&newEstPrdCnt=${newEstPrdCnt}" charEncoding="utf-8" />

```

■ 예제 소스(R)

```

library(httr)
library(rvest)
library(jsonlite)

```

```
library(ggplot2)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

url_page <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'json',
      jsonVD = 'Y',
      apiKey = v_apiKey %>% I(),
      userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1', # 시계열 조회 키값
      prdSe = 'Y',
      newEstPrdCnt = 3, # 최근수록시점 개수
      prdInterval = 1 # 수록시점 간격
    )
  )

url_page %>% content(as = 'text', encoding = 'UTF-8') %>% fromJSON() -> json

df_lists <-
  data.frame(
    C1_NM = c(json$C1_NM),
    ITM_NM = c(json$ITM_NM),
    prd_de = c(json$PRD_DE),
    dt = c(json$DT)
  )

# 바차트
ggplot(df_lists, aes(x = prd_de, y = dt, fill = prd_de)) + xlab("시점") + ylab("") + ggtitle(json$TBL_NM[1]) + geom_bar(stat = "identity")
```

■ 예제 소스(Python)

```
import json
from urllib.request import urlopen # python 3.x 버전에서 사용 (2.x 버전이라면 from urllib import urlopen)
import matplotlib.pyplot as plt

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 json 데이터 가져오기
with urlopen("https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWR")
```

```
jNjgxN2MzNDgwNmY=&format=json&jsonVD=Y&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026
_1&prdSe=Y&newEstPrdCnt=3") as url:
    json_file = url.read()

py_json = json.loads(json_file.decode('utf-8'))

#변수 지정 및 데이터 저장
xAxis = []
yAxis = []
title = ''
for i, v in enumerate(py_json):
    xAxis.append(v['PRD_DE']) #x축에 들어갈 데이터
    yAxis.append(int(v['DT'])) #y축에 들어갈 데이터
    if i == 0 :
        title = v['TBL_NM'] #차트제목

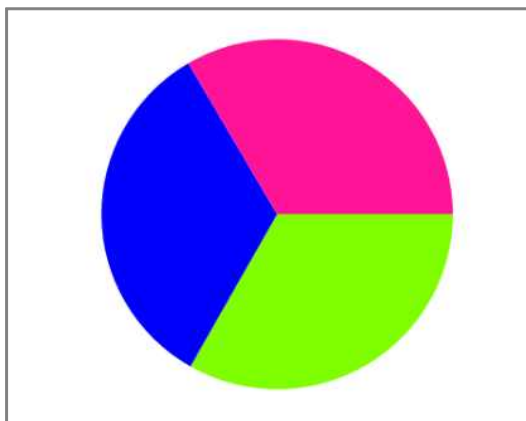
#Bar차트 그리기
plt.bar(xAxis, yAxis)
plt.title(title)

#y축 수치를 안보이게 하는 코드. 필요에 따라 선택하여 사용
plt.gca().axes.yaxis.set_visible(False)

#Bar의 가운데에 text로 수치 표시
for i, v in enumerate(xAxis):
    plt.text(v, yAxis[i], yAxis[i],
             fontsize = 9,
             color='blue',
             horizontalalignment='center',
             verticalalignment='bottom')

plt.show()
```

■ 예제 소스 결과(파이차트)



■ 예제 소스(JSP)

```
<!--
```

KOSIS OpenAPI를 이용하여 파이차트를 출력하는 예제입니다.

이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

- * 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.

```
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide02/js/dojo.js" ></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide02/js/json.js" ></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide02/js/ajax.js"></script>
<script type="text/javascript" language="JavaScript">

var mapData;

// window onload 되었을때 실행 함수
dojo.addOnLoad ( function() {

getList();
});

/*****
* 통계자료 조회 함수
*****/
function getList() {

// ajax 통신을 위한 파라미터를 변수에 담는다.
var paraObj = {
// 임의의 jsp 페이지를 호출함으로써 cross domain 제약을 우회할 수 있다. (devGuidePop.jsp 소스는 소스 하단에 제공)
url : "http://[개발자 홈페이지 주소]/devGuidePop.jsp?method=getList&key=ZjZjOTI3MjRjNmU1YzdhdjZTMwOWRjNjgxN2MzNDgwNmY=&format=json&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3",
sync : true,
load : function(resObj, a, b) {

// 차트에 보여주기 위해 값을 담고있을 배열변수를 선언
```

```

mapData = resObj;
// 차트를 생성하기위한 함수 호출
chartShow();
},
error : function ( resObj, e ) {

alert(dojotoJson(resObj));
}
}

// ajax 통신 호출 함수
sendPost( paraObj );
}

function chartShow(){

// 차트에 보여주기위해 값을 담고있을 배열변수를 선언
var data = new Array(mapData.length);

// 데이터를 배열변수에 적재
for(var i=0;i<mapData.length;i++) {
data[i] = mapData[i].DT;
}

// 파이차트를 출력하기위한 JavaScript Start
var canvas = document.getElementById("chartcanvas");

var context = canvas.getContext("2d");

var sw = canvas.width;
var sh = canvas.height;

var PADDING=50;

var colors = ["#7cfc00", "#0000ff", "#ff1493", "#66CDAA", "#ff00ff", "#FFD700"];

var center_X=sw/2; //원의 중심 x 좌표
var center_Y=sh/2; //원의 중심 y 좌표

var radius = Math.min(sw-(PADDING*2), sh-(PADDING*2)) / 2;
var angle = 0;
var total = 0;

for (var i in data) { total += Number(data[i]); } //데이터(data)의 총합 계산

for (var i = 0; i < data.length; i++) {

context.fillStyle = colors[i]; //생성되는 부분의 채울 색 설정
context.beginPath();
context.moveTo(center_X, center_Y); //원의 중심으로 이동
context.arc(center_X, center_Y, radius, angle, angle +(Math.PI*2*(data[i]/total)));
context.lineTo(center_X,center_Y);
context.fill();
}

```

```

angle += Math.PI*2*(data[i]/total);
}
// 파이차트를 출력하기 위한 JavaScript End
}
</script>
</head>
<body>
<canvas id="chartcanvas" width="500" height="400"></canvas>
</body>
</html>

// cross domain 제약을 우회하기 위한 jsp (devGuidePop.jsp)
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>

<c:set var="method" ><%out.println(request.getParameter("method") == null ? "" : (request.getParameter("method"))); %></c:set>
<c:set var="key" ><%out.println(request.getParameter("key") == null ? "" : (request.getParameter("key"))); %></c:set>
<c:set var="userStatsId"><%out.println(request.getParameter("userStatsId") == null ? "" : (request.getParameter("userStatsId"))); %></c:set>
<c:set var="newEstPrdCnt"><%out.println(request.getParameter("newEstPrdCnt")== null ? "" : (request.getParameter("newEstPrdCnt"))); %></c:set>
<c:set var="prdSe" ><%out.println(request.getParameter("prdSe") == null ? "" : (request.getParameter("prdSe"))); %></c:set>
<c:set var="format" ><%out.println(request.getParameter("format") == null ? "" : (request.getParameter("format"))); %></c:set>

<c:import url="http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=${method}&apiKey=${key}&format=${format}&userStatsId=${userStatsId}&prdSe=${prdSe}&newEstPrdCnt=${newEstPrdCnt}" charEncoding="utf-8" />

```

■ 예제 소스(R)

```

library(httr)
library(rvest)
library(jsonlite)
library(ggplot2)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

url_page <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'json'.

```

```

jsonVD = 'Y',
apiKey = v_apiKey %>% I(),
userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1', # 시계열 조회 키값
prdSe = 'Y',
newEstPrdCnt = 3, # 최근수룩시점 개수
prdInterval = 1 # 수룩시점 간격
)
)

url_page %>% content(as = 'text', encoding = 'UTF-8') %>% fromJSON() -> json

df_lists <-
  data.frame(
    C1_NM = c(json$C1_NM),
    ITM_NM = c(json$ITM_NM),
    prd_de = c(json$PRD_DE),
    dt = c(json$DT)
  )

# 파이차트 비율 라벨 값
pct <-
  round(as.numeric(df_lists$dt) / sum(as.numeric(df_lists$dt)) * 100, 1)
df_lists <- data.frame(df_lists, pct = pct)

# 파이차트
ggplot(df_lists, aes(
  x = factor(1),
  y = "",
  fill = factor(prd_de)
)) +
  geom_bar(stat = 'identity') +
  theme_void() +
  ggtitle(json$TBL_NM[1]) +
  coord_polar('y', start = 0) +
  geom_text(aes(label = paste0(round(pct, 1), '%')),
            position = position_stack(vjust = 0.5))

```

■ 예제 소스(Python)

```

import json
from urllib.request import urlopen
import matplotlib.pyplot as plt

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 json 데이터 가져오기
with urlopen("https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRiNmU1YzdhZTMwOWR")

```



```
jNjgxN2MzNDgwNmY=&format=json&jsonVD=Y&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026
_1&prdSe=Y&newEstPrdCnt=3") as url:
    json_file = url.read()

py_json = json.loads(json_file.decode('utf-8'))

#변수 지정 및 데이터 저장
labels = []
ratio = []
title = ''

for i, v in enumerate(py_json):
    labels.append(v['PRD_DE']) #년도 데이터
    ratio.append(v['DT']) #값/비율 데이터
    if i == 0 :
        title = v['TBL_NM'] #차트제목

#Pie차트 그리기
plt.pie(ratio, labels=labels, autopct='%.1f%%')
plt.title(title)
plt.show()
```

■ 예제 소스 결과(표차트)

연구 가구 및 주택 - 읍면동(2015) 시군구(2016~) 단위 : 명

시점	총인구(전국)
2017	51422507
2018	51629512
2019	51779203

출처 : 「인구총조사」

■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 통계표를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
```

```
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.
css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide0
1/js/dojo.js" ></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide0
1/js/json.js" ></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide0
1/js/ajax.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
dojo.addOnLoad ( function() {

getList();
});

/*****
* 통계자료 조회 함수
*****/
function getList(vwcd, listLev, parentId) {

// ajax 통신을 위한 파라미터를 변수에 담는다. (통계자료)
var paraObj = {
// 임의의 jsp 페이지를 호출함으로써 cross domain 제약을 우회할 수 있다. (devGuidePop.jsp 소스는 소스 하단에
제공)
url : "http:[개발자 홈페이지 주소]/devGuidePop.jsp?method=getList&key=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN
2MzNDgwNmY=&format=json&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&ne
wEstPrdCnt=3",
sync : true,
load : function(mapData, a, b) {

var strTable = "";
// 조회된 결과를 이용하여 통계표 작성
strTable += "<h5>" + mapData[0].TBL_NM + "</h5>";
strTable += "<p style='position:absolute; top:32px; left:450px;'>단위 : " + mapData[0].UNIT_NM + "</p>";
strTable += "<table cellpadding='0' cellspacing='0' class='TB02' summary='' style='margin-top:15px;width:500p
x;'>";
strTable += "<thead>";
strTable += "<tr>";
strTable += "<th class='fst a_LC' scope='col'>시점</th>";
strTable += "<th scope='col'>" + mapData[0].ITM_NM + "(" + mapData[0].C1_NM + ")" + "</th>";
strTable += "</tr>";
strTable += "</thead>";
strTable += "<tbody>";
```

```

for(var i=0;i<mapData.length;i++) {
strTable += "<tr>";
strTable += "<td class='a_LC'>" + mapData[i].PRD_DE + "</td>";
strTable += "<td class='a_LC'>" + mapData[i].DT.replace(/,/gi,'') + "</td>";
strTable += "</tr>";
}
strTable += "</tbody>";
strTable += "</table>";

document.getElementById("content").innerHTML = strTable;
},
error : function ( resObj, e ) {

alert(dojo.toJson(resObj));
}
}
// ajax 통신 호출 함수
sendPost( paraObj );

// ajax 통신을 위한 파라미터를 변수에 담는다. (출처정보)
var paraObj = {
// 임의의 jsp 페이지를 호출함으로써 cross domain 제약을 우회할 수 있다.
url : "http://mgmk.kosis.kr/openapi_dev/devGuidePop.jsp?method=getMeta&key=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&type=SOURCE&format=json&orgId=101&tblId=DT_1B01003",
sync : true,
load : function(mapData, a, b) {

var strTable = "";

// 조회된 결과를 이용하여 출처 작성
strTable += "<p style='position:absolute; top:250px; '>출처 : " + mapData[0].JOSA_NM + "</p>";

document.getElementById("content").innerHTML += strTable;
},
error : function ( resObj, e ) {

alert(dojo.toJson(resObj));
}
}
// ajax 통신 호출 함수
sendPost( paraObj );
}
</script>
</head>
<body>
<div id="content" style="padding-top:30px;"></div>
</body>
</html>

<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>

```

```
<c:set var="method" ><%out.println(request.getParameter("method") == null ? "" : (request.getParameter("method"))); %></c:set>
<c:set var="key" ><%out.println(request.getParameter("key") == null ? "" : (request.getParameter("key"))); %></c:set>
<c:set var="userStatsId" ><%out.println(request.getParameter("userStatsId") == null ? "" : (request.getParameter("userStatsId"))); %></c:set>
<c:set var="newEstPrdCnt" ><%out.println(request.getParameter("newEstPrdCnt")== null ? "" : (request.getParameter("newEstPrdCnt"))); %></c:set>
<c:set var="prdSe" ><%out.println(request.getParameter("prdSe") == null ? "" : (request.getParameter("prdSe"))); %></c:set>
<c:set var="format" ><%out.println(request.getParameter("format") == null ? "" : (request.getParameter("format"))); %></c:set>
<c:set var="type" ><%out.println(request.getParameter("type") == null ? "" : (request.getParameter("type"))); %></c:set>
<c:set var="orgId" ><%out.println(request.getParameter("orgId") == null ? "" : (request.getParameter("orgId"))); %></c:set>
<c:set var="tblId" ><%out.println(request.getParameter("tblId") == null ? "" : (request.getParameter("tblId"))); %></c:set>

<c:import url="http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=${method}&apiKey=${key}&format=${format}&userStatsId=${userStatsId}&prdSe=${prdSe}&newEstPrdCnt=${newEstPrdCnt}&tblId=${tblId}&orgId=${orgId}&type=${type}" charEncoding="utf-8"/>
```

■ 예제 소스(R)

```
library(httr)
library(rvest)
library(jsonlite)
library(dplyr)

rm(list = ls()) # 모든변수 초기화
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

url_page <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'json',
      jsonVD = 'Y',
      apiKey = v_apiKey %>% I(),
      userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1',
      # 시계열 조회 키값
      prdSe = 'Y',
      newEstPrdCnt = 3 # 최근수록시점 개수
    )
  )

url_page %>% content(as = 'text', encoding = 'UTF-8') %>% fromJSON() -> json

df_lists <- data.frame()
```

```
df_lists <- cbind(json$PRD_DE)
df_lists <- cbind(df_lists, json$DT)

# 사용자가 선택한 통계 자료의 분류 갯수에 따라 변경
# 샘플소스의 key는 1개분류, 1개 항목 지정됨

colnames(df_lists) <-
  c("시점", paste(json$ITM_NM[1], "(", json$C1_NM[1], ")", sep = ""))
View(df_lists)
```

■ 예제 소스(Python)

```
import json
from urllib.request import urlopen # python 3.x 버전에서 사용 (2.x 버전이라면 from urllib import urlopen)
import matplotlib.pyplot as plt

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 json 데이터 가져오기
with urlopen("https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=json&jsonVD=Y&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3") as url:
    json_file = url.read()

py_json = json.loads(json_file.decode('utf-8'))

#변수 지정 및 데이터 저장
data = []

for i, v in enumerate(py_json):
    value = []
    value.append(v['PRD_DE'])
    value.append(v['DT'])

    data.append(value)

#Table 만들기
fig, ax = plt.subplots(1,1)
column_labels=["시점", "총인구(전국)"]
ax.axis('tight')
ax.axis('off')
ax.table(cellText=data,colLabels=column_labels,colColours=["yellow"] * 2, loc="center", cellLoc='center')

plt.show()
```

2.2.3.2 SDMX(DSD)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do>

■ 입력 변수

● 자료등록 방법

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
userStatsId	String	사용자 등록 통계표	필수
type	String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
format	String	결과 유형(json, sdmx)	필수
version	String	결과값 구분	생략시 구버전으로 데이터 출력

● 통계표선택 방법

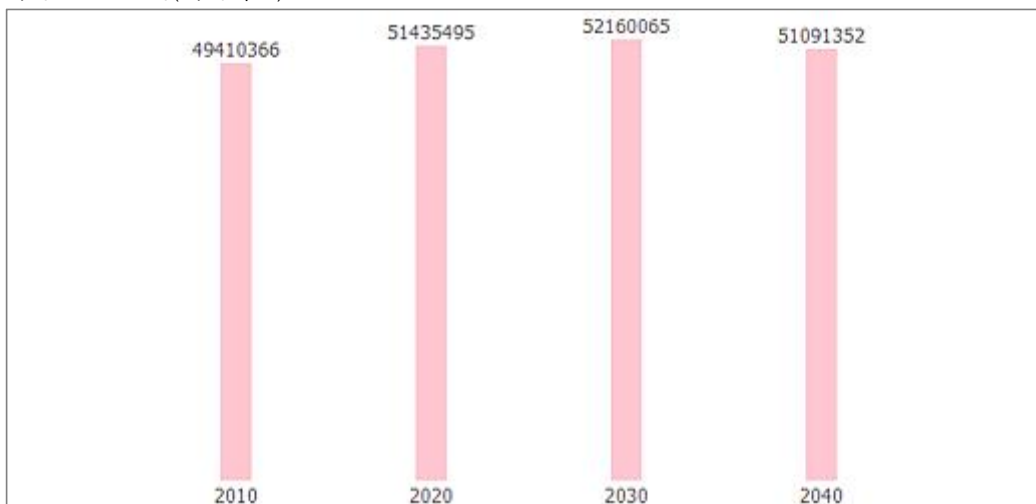
항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표 ID	필수
objL2 ~ objL8	String	분류2(두번째 분류코드) ~ 분류8(여덟째 분류코드)	선택
itmId	String	항목	필수
type	String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
format	String	결과 유형(json, sdmx)	필수
version	String	결과값 구분	생략시 구버전으로 데이터 출력

■ 출력 변수

항목명(영문)					항목설명	
Header	Id		기관코드_통계표ID			
	Name		통계표명			
	Prepared		전송시간			
	Sender	Id		전송기관		
		Name		전송기관명		
		Contact	Department	담당부서		
			Telephone	담당부서 연락처		
Source		출처				
Codelist	Codelist	Id		코드리스트ID		
		Name		코드리스트명		
		Description		코드리스트영문명		
		Code	Id	코드ID		

			Name	코드명
Concepts	ConceptsScheme	Id		컨셉스키마ID
		Name		컨셉스키마명
		Description		컨셉스키마영문명
		Concept	Id	컨셉ID
			Name	컨셉명
			Description	컨셉영문명
DataStructures	DataStructure	Id		통계표ID
		Name		통계표명
		DataStructureComponents	Dimension	Id
				diMenJenId
				conceptIdentity
				Id
				컨셉객체Id

■ 예제 소스 결과(막대차트)



■ 예제 소스(JSP)

```

<!--
KOSIS OpenAPI를 이용하여 막대차트를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">

```

```

<head>
<xml:namespace prefix="v"/>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.
css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.
6.1.min.js"></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide0
2/js/jquery.SimpleChart.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
$(document).ready(function(){
getSubList();
});

/*****
* 통계자료 조회 함수
*****/
function getSubList() {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTM
wOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2
/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1",
data: "",
async : true,
dataType: "xml",
success:function(object)
{

var data = object.documentElement.childNodes[1].childNodes[0].childNodes;

// 차트에 보여주기위해 값을 담고있을 배열변수를 선언
var aData = new Array();

// 데이터를 배열변수에 적재
for(var i=0; i < data.length; i++) {

aData[i] = [data[i].attributes[0].value.replace(/,/gi,""), data[i].attributes[1].value, 'pink'];
}

// 차트를 그리기위한 옵션을 정의
var options = {'BarSize': '20px', 'BarSpace': '2px', 'type' : 'horizontal', 'Font': '2px'}

// 차트를 화면에 출력
$("#chart2").SimpleChart(aData,options);
},

```



```

error: function(xhr,status,error){
alert("error = " + error);
}
});
}

</script>
</head>
<body>
<div id="chart2" style="padding-top:30px;"></div>
</body>
</html>

```

■ 예제 소스(R)

```

library(httr)
library(rvest)
library(XML)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZiOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseUrl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

res <-
  GET(
    url = baseUrl,
    query = list(
      method = 'getList',
      format = 'sdmx',
      apiKey = v_apiKey %>% I(),
      userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1',
      # 시계열 조회 키값
      jsonVD = 'Y',
      type = 'StructureSpecific',
      prdSe = 'Y',
      newEstPrdCnt = 3,
      # 최근수록시점 개수
      prdInterval = 1,
      # 수록시점 간격
      version = 'v2_1'
    )
  )

docParse <- xmlParse(res)

tbl_title <- xmlToList(docParse)$Header$Name # 통계표명
docList <- xmlToList(docParse)$DataSet$Series

```

```
df_lists <- data.frame()

for (i in 1:length(docList)) {
  if (length(docList[i]$Obs) == 2) {
    df_lists <-
      rbind(df_lists, cbind(
        prd_de = paste(docList[i]$Obs[2]),
        dt = paste(docList[i]$Obs[1])
      ))
  }
}

# 바차트
ggplot(df_lists, aes(x = prd_de, y = dt, fill = prd_de)) + xlab("시점") + ylab("") + ggtitle(tbl_title) + geom_bar(stat = "identity")
```

■ 예제 소스(Python)

```
import matplotlib.pyplot as plt
import requests
from bs4 import BeautifulSoup

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 sdmx 데이터 가져오기
open_url = 'https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

#변수 지정 및 데이터 저장
xAxis = []
yAxis = []
title = soup.find('common:name').text #차트제목

dataList = soup.find_all('obs')

for item in dataList:
    xAxis.append(item.get('time_period')) #년도 데이터
    yAxis.append(int(item.get('obs_value'))) #값/비율 데이터

#Bar차트 그리기
plt.bar(xAxis, yAxis)
```

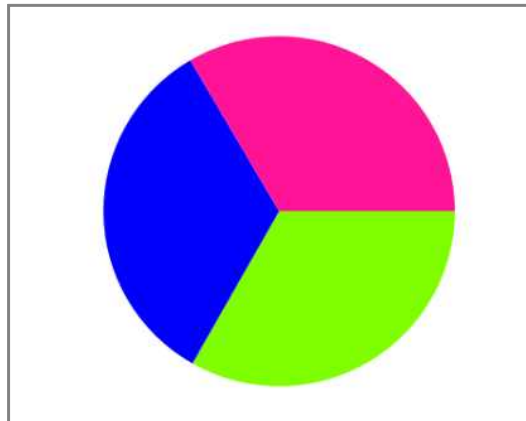
```
plt.title(title)

#y축 수치를 안보이게 하는 코드. 필요에 따라 선택하여 사용
plt.gca().axes.yaxis.set_visible(False)

#Bar의 가운데에 text로 수치 표시
for i, v in enumerate(xAxis):
    plt.text(v, yAxis[i], yAxis[i],
             fontsize = 9,
             color='blue',
             horizontalalignment='center',
             verticalalignment='bottom')

plt.show()
```

■ 예제 소스 결과(파이차트)



■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 파이차트를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<xml:namespace prefix="v"/>
```

```

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.
css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.
6.1.min.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
$(document).ready(function(){
getSubList();
});

/*****
* 통계자료 조회 함수
*****/
function getSubList() {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTM
wOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2
/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1",
data: "",
async : true,
dataType: "xml",
success:function(object)
{

var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

// 차트에 보여주기위해 값을 담고있을 배열변수를 선언
var data = new Array(mapdata.length);

// 데이터를 배열변수에 적재
for(var i=0;i<mapdata.length;i++) {
data[i] = mapdata[i].attributes[0].value;
}

// 파이차트를 출력하기위한 JavaScript Start
var canvas = document.getElementById("chartcanvas");
var context = canvas.getContext("2d");
var sw = canvas.width;
var sh = canvas.height;
var PADDING=100;

//Browser별 색상 lawngreen, blue, deeppink, aquamarine3, magenta, gold
var colors = ["#7cfc00", "#0000ff", "#ff1493", "#66CDAA", "#ff00ff", "#FFD700"];

var center_X=sw/2; //원의 중심 x 좌표

```

```

var center_Y=sh/2; //원의 중심 y 좌표
// 두 계산값 중 작은 값을 원의 반지름으로 설정
var radius = Math.min(sw-(PADDING*2), sh-(PADDING*2)) / 2;
var angle = 0;
var total = 0;

for (var i in data) { total += Number(data[i]); } //데이터(data)의 총합 계산

for (var i = 0; i < data.length; i++) {

context.fillStyle = colors[i]; //생성되는 부분의 채울 색 설정
context.beginPath();
context.moveTo(center_X, center_Y); //원의 중심으로 이동
context.arc(center_X, center_Y, radius, angle, angle +(Math.PI*2*(data[i]/total)));
context.lineTo(center_X,center_Y);
context.fill();
angle += Math.PI*2*(data[i]/total);
}
// 파이차트를 출력하기 위한 JavaScript End
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}
</script>
</head>
<body>
<canvas id="chartcanvas" width="500" height="400"></canvas>
</html>

```

■ 예제 소스(R)

```

library(httr)
library(rvest)
library(XML)
library(ggplot2)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

res <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'sdmx',
      apiKey = v_apiKey %>% I(),
      userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1',

```

```

# 시계열 조회 키값
jsonVD = 'Y',
type = 'StructureSpecific',
prdSe = 'Y',
newEstPrdCnt = 3,
# 최근수룩시점 개수
prdInterval = 1,
# 수룩시점 간격
version = 'v2_1'
)
)

docParse <- xmlParse(res)

v_tbl_nm <- xmlToList(docParse)$Header$Name # 통계표명
docList <- xmlToList(docParse)$DataSet$Series

df_lists <- data.frame()

for (i in 1:length(docList)) {
  if (length(docList[i]$Obs) == 2) {
    df_lists <-
      rbind(df_lists, cbind(
        prd_de = paste(docList[i]$Obs[2]),
        dt = paste(docList[i]$Obs[1])
      ))
  }
}

# 파이차트 비율 라벨 값
pct <-
  round(as.numeric(df_lists$dt) / sum(as.numeric(df_lists$dt)) * 100, 1)
df_lists <- data.frame(df_lists, pct = pct)

# 파이차트
ggplot(df_lists, aes(
  x = factor(1),
  y = "",
  fill = factor(prd_de)
)) +
  geom_bar(stat = 'identity') +
  theme_void() +
  ggtitle(v_tbl_nm) +
  coord_polar('y', start = 0) +
  geom_text(aes(label = paste0(round(pct, 1), '%')),
            position = position_stack(vjust = 0.5))

```

■ 예제 소스(Python)

```

import matplotlib.pyplot as plt
import requests
from bs4 import BeautifulSoup

```

```
# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 sdmx 데이터 가져오기
open_url = 'https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

#변수 지정 및 데이터 저장
labels = []
ratio = []
title = soup.find('common:name').text #차트제목

dataList = soup.find_all('obs')

for item in dataList:
    labels.append(item.get('time_period')) #년도 데이터
    ratio.append(item.get('obs_value')) #값/비율 데이터

#Pie차트 그리기
plt.pie(ratio, labels=labels, autopct='%1f%%')
plt.title(title)
plt.show()
```

■ 예제 소스 결과(표차트)

연구 가구 및 주택 - 읍면동(2015) 시군구(2016~) 단위 : 명

시점	총인구(전국)
2017	51422507
2018	51629512
2019	51779203

출처 : 「인구총조사」

■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 통계표를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
```

통신문제가 발생 할 수 있습니다.

JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여

호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.

-->

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<%@ page contentType="text/html; charset=utf-8" %>
```

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
```

```
<head>
```

```
<xml:namespace prefix="v"/>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

```
<title>통계청 - 공유서비스</title>
```

```
<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />
```

```
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.6.1.min.js"></script>
```

```
<script type="text/javascript" language="JavaScript">
```

```
// window onload 되었을때 실행 함수
```

```
$(document).ready(function(){
```

```
getSubList();
```

```
});
```

```
var strUnitId, strItdId, strC1;
```

```
/*
*****
* 통계자료 조회 함수
*****
*/
```

```
* 통계자료 조회 함수
```

```
*****
*/
```

```
function getSubList() {
```

```
// ajax 통신을 위한 호출 함수
```

```
$.ajax({
```

```
type: "GET",
```

```
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1",
```

```
data: "",
```

```
sync : true,
```

```
dataType: "xml",
```

```
success:function(object)
```

```
{
```

```
var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;
```

```
// 항목, 분류, 단위의 명칭을 조회하기위해 각각의 코드를 변수에 저장
```

```
for(var i=0;i<object.documentElement.childNodes[1].childNodes[0].attributes.length;i++) {
```

```
if (object.documentElement.childNodes[1].childNodes[0].attributes[i].nodeName == "ITEM") {
```

```
strItdId = object.documentElement.childNodes[1].childNodes[0].attributes[i].value;
```



```

} else if (object.documentElement.childNodes[1].childNodes[0].attributes[i].nodeName == "C_HJG") {

strC1 = object.documentElement.childNodes[1].childNodes[0].attributes[i].value;
} else if (object.documentElement.childNodes[1].childNodes[0].attributes[i].nodeName == "UNIT") {

strUnitId = object.documentElement.childNodes[1].childNodes[0].attributes[i].value;
}
}
var strTable = "";

// 조회된 결과를 이용하여 통계표 작성
strTable += "<p style='position:absolute; top:32px; left:450px;' id='unitId'></p>";
strTable += "<table cellpadding='0' cellspacing='0' class='TB02' summary='' style='margin-top:15px;width:500px;'>";
strTable += "<thead>";
strTable += "<tr id = 'theadId'>";

strTable += "</tr>";
strTable += "</thead>";
strTable += "<tbody>";

for(var i=0;i<mapdata.length;i++) {
strTable += "<tr>";
strTable += "<td class='al_C'>" + mapdata[i].attributes[1].value + "</td>";
strTable += "<td class='al_C'>" + mapdata[i].attributes[0].value + "</td>";
strTable += "</tr>";
}
strTable += "</tbody>";
strTable += "</table>";

document.getElementById("content").innerHTML = strTable;

fnGetTitle ();
fnGetUnit ();
fnGetItm ();
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

/*****
* 통계표 명칭 조회
*****/
function fnGetTitle () {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdZTMwOWRjNjgxN2MzNDgwNmY=&format=xml&type=TBL&orgId=101&tblId=DT_1B01003",
data: "",

```

```

sync : true,
dataType: "xml",
success:function(object)
{

var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

// 조회된 결과를 이용하여 통계표 명칭 작성
var strTable = document.getElementById("content").innerHTML;
document.getElementById("content").innerHTML = "<h5>" + object.documentElement.childNodes[1].childNodes
[0].childNodes[0].data + "</h5>" + strTable;

},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

/*****
* 항목 명칭 조회
*****/
function fnGetItm () {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=ITM&orgId=101&tblId=DT_1B01003&objId=ITEM&itmId=" + strItmId,
data: "",
sync : true,
dataType: "xml",
success:function(object)
{
// 분류 명칭 조회
fnGetC1(object.documentElement.childNodes[1].childNodes[0].childNodes[4].childNodes[0].data);
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

/*****
* 분류 명칭 조회
*****/
function fnGetC1(itmNm) {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=ITM&orgId=101&tblId=DT_1B01003&objId=HJG&itmId=" + strC1,

```

```

data: "",
sync : true,
dataType: "xml",
success:function(object)
{
var strTable = "";

// 조회된 결과를 이용하여 항목, 분류 작성
strTable += "<th class='fst a_LC' scope='col'>시점</th>";
strTable += "<th scope='col'>" + itmNm + "(" + object.documentElement.childNodes[1].childNodes[0].childNodes[4].childNodes[0].data + ")" + "</th>";

document.getElementById("theadId").innerHTML = strTable;

},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

/*****
* 단위 명칭 조회
*****/
function fnGetUnit () {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=xml&type=UNIT&unitId=" + strUnitId,
data: "",
sync : true,
dataType: "xml",
success:function(object)
{

// 조회된 결과를 이용하여 단위 명칭 작성
var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

document.getElementById("unitId").innerHTML = "단위 : " + object.documentElement.childNodes[1].childNodes[0].childNodes[0].data;

},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

</script>
</head>
<body>

```

```
<div id="content" style="padding-top:30px;"></div>
</html>
```

■ 예제 소스(R)

```
library(httr)
library(rvest)
library(XML)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

res <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'sdmx',
      apiKey = v_apiKey %>% I(),
      userStatsId = 'openapisample/101/DT_11N1502/2/1/20191106094026_1', # 시계열 조회 키값
      jsonVD = 'Y',
      type = 'StructureSpecific',
      prdSe = 'Y',
      newEstPrdCnt = 3, # 최근수록시점 개수
      prdInterval = 1, # 수록시점 간격
      version = 'v2_1'
    )
  )

docParse <- xmlParse(res)

docList <- xmlToList(docParse)$DataSet$Series

df_lists <- data.frame()

for (i in 1:length(docList)) {
  if (length(docList[i]$Obs) == 2) {
    df_lists <-
      rbind(df_lists, cbind(paste(docList[i]$Obs[2]), paste(docList[i]$Obs[1])))
  }
}

colnames(df_lists) <- c("시점", "수치")

View(df_lists)
```

■ 예제 소스(Python)

```
import matplotlib.pyplot as plt
import requests
```

```

from bs4 import BeautifulSoup

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 sdmx 데이터 가져오기
open_url = 'https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

dataList = soup.find_all('obs')

#변수 지정 및 데이터 저장
data = []

for item in dataList:
    value = []
    value.append(item.get('time_period')) #년도 데이터
    value.append(item.get('obs_value')) #값 데이터

    data.append(value)

#Table 만들기
fig, ax = plt.subplots(1,1)
column_labels=["시점", "총인구(전국)"]
ax.axis('tight')
ax.axis('off')
ax.table(cellText=data,colLabels=column_labels,colColours=["yellow"] * 2, loc="center", cellLoc='center')

plt.show()

```

2.2.3.3 SDMX(Generic)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do>

■ 입력 변수

● 자료등록 방법

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
userStatsId		String	사용자 등록 통계표	필수
type		String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
prdSe		String	수록주기	필수
시점기준	startPrdDe	String	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료	newEstPrdCnt	String	최근수록시점 개수	
기준	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수
version		String	결과값 구분	생략시 구버전으로 데이터 출력

● 통계표선택 방법

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
orgId		String	기관 ID	필수
tblId		String	통계표 ID	필수
objL2 ~ objL8		String	분류2(두번째 분류코드) ~ 분류8(여덟째 분류코드)	선택
itmId		String	항목	필수
type		String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
prdSe		String	수록주기	필수
시점기준	startPrdDe	String	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료	newEstPrdCnt	String	최근수록시점 개수	
기준	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수
version		String	결과값 구분	생략시 구버전으로 데이터 출력

■ 출력 변수

항목명(영문)					항목설명
Header	ID				기관코드_통계표ID
	Name				통계표명
	Prepared				전송시간
	Sender	Id		전송기관	
		Name		전송기관명	
		Contact	Department		담당부서
			Telephone		담당부서 연락처
	Source				출처
Series	SeriesKey	Value	Id	시리즈키ID	
			value	시리즈키값	
			UNIT	단위	
	Obs	ObsDimension	Value	시점	
		ObsValue	Value	수치자료값	

■ 예제 소스 결과(막대차트)



■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 막대차트를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
```

```
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<xml:namespace prefix="v"/>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.6.1.min.js"></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide02/js/jquery.SimpleChart.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
$(document).ready(function(){
getSubList();
});

/*****
* 통계자료 조회 함수
*****/
function getSubList() {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1",
data: "",
async : true,
dataType: "xml",
success:function(object)
{

var data = object.documentElement.childNodes[1].childNodes[0].childNodes;

// 차트에 보여주기위해 값을 담고있을 배열변수를 선언
var aData = new Array();

// 데이터를 배열변수에 적재
for(var i=0; i < data.length; i++) {

aData[i] = [data[i].attributes[0].value.replace(/,/gi,""), data[i].attributes[1].value, 'pink'];
}

}
```



```
// 차트를 그리기위한 옵션을 정의
var options = {'BarSize': '20px', 'BarSpace': '2px', 'type' : 'horizontal', 'Font': '2px'}

// 차트를 화면에 출력
$("#chart2").SimpleChart(aData,options);
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

</script>
</head>
<body>
<div id="chart2" style="padding-top:30px;"></div>
</body>
</html>
```

■ 예제 소스(R)

```
library(httr)
library(rvest)
library(XML)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

res <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'sdmx',
      apiKey = v_apiKey %>% I(),
      userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1',
      # 시계열 조회 키값
      jsonVD = 'Y',
      type = 'StructureSpecific',
      prdSe = 'Y',
      newEstPrdCnt = 3,
      # 최근수록시점 개수
      prdInterval = 1,
      # 수록시점 간격
      version = 'v2_1'
    )
  )
```

```
docParse <- xmlParse(res)

tbl_title <- xmlToList(docParse)$Header$Name # 통계표명
docList <- xmlToList(docParse)$DataSet$Series

df_lists <- data.frame()

for (i in 1:length(docList)) {
  if (length(docList[i]$Obs) == 2) {
    df_lists <-
      rbind(df_lists, cbind(
        prd_de = paste(docList[i]$Obs[2]),
        dt = paste(docList[i]$Obs[1])
      ))
  }
}

# 바차트
ggplot(df_lists, aes(x = prd_de, y = dt, fill = prd_de)) + xlab("시점") + ylab("") + ggtitle(tbl_title) + geom_bar(stat = "identity")
```

■ 예제 소스(Python)

```
import matplotlib.pyplot as plt
import requests
from bs4 import BeautifulSoup

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 sdmx 데이터 가져오기
open_url = 'https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

#변수 지정 및 데이터 저장
xAxis = []
yAxis = []
title = soup.find('common:name').text #차트제목

dataList = soup.find_all('obs')
```

```
for item in dataList:
    xAxis.append(item.get('time_period')) #년도 데이터
    yAxis.append(int(item.get('obs_value'))) #값/비율 데이터

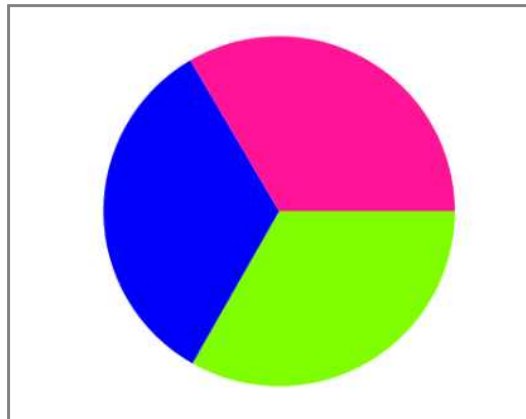
#Bar차트 그리기
plt.bar(xAxis, yAxis)
plt.title(title)

#y축 수치를 안보이게 하는 코드. 필요에 따라 선택하여 사용
plt.gca().axes.yaxis.set_visible(False)

#Bar의 가운데에 text로 수치 표시
for i, v in enumerate(xAxis):
    plt.text(v, yAxis[i], yAxis[i],
             fontsize = 9,
             color='blue',
             horizontalalignment='center',
             verticalalignment='bottom')

plt.show()
```

■ 예제 소스 결과(파이차트)



■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 파이차트를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1
```

```

1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<xml:namespace prefix="v"/>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.
css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.
6.1.min.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
$(document).ready(function(){
getSubList();
});

/*****
* 통계자료 조회 함수
*****/
function getSubList() {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTM
wOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2
/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1",
data: "",
async : true,
dataType: "xml",
success:function(object)
{

var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

// 차트에 보여주기위해 값을 담고있을 배열변수를 선언
var data = new Array(mapdata.length);

// 데이터를 배열변수에 적재
for(var i=0;i<mapdata.length;i++) {
data[i] = mapdata[i].attributes[0].value;
}

// 파이차트를 출력하기위한 JavaScript Start
var canvas = document.getElementById("chartcanvas");
var context = canvas.getContext("2d");
var sw = canvas.width;
var sh = canvas.height;

```

```

var PADDING=100;

//Browser별 색상 lawngreen, blue, deeppink, aquamarine3, magenta, gold
var colors = ["#7cfc00", "#0000ff", "#ff1493", "#66CDAA", "#ff00ff", "#FFD700"];

var center_X=sw/2; //원의 중심 x 좌표
var center_Y=sh/2; //원의 중심 y 좌표
// 두 계산값 중 작은 값을 원의 반지름으로 설정
var radius = Math.min(sw-(PADDING*2), sh-(PADDING*2)) / 2;
var angle = 0;
var total = 0;

for (var i in data) { total += Number(data[i]); } //데이터(data)의 총합 계산

for (var i = 0; i < data.length; i++) {

context.fillStyle = colors[i]; //생성되는 부분의 채울 색 설정
context.beginPath();
context.moveTo(center_X, center_Y); //원의 중심으로 이동
context.arc(center_X, center_Y, radius, angle, angle +(Math.PI*2*(data[i]/total)));
context.lineTo(center_X,center_Y);
context.fill();
angle += Math.PI*2*(data[i]/total);
}
// 파이차트를 출력하기 위한 JavaScript End
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}
</script>
</head>
<body>
<canvas id="chartcanvas" width="500" height="400"></canvas>
</html>

```

■ 예제 소스(R)

```

library(httr)
library(rvest)
library(XML)
library(ggplot2)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

res <-
GET(

```

```

url = baseUrl,
query = list(
  method = 'getList',
  format = 'sdmx',
  apiKey = v_apiKey %>% I(),
  userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1',
  # 시계열 조회 기간
  jsonVD = 'Y',
  type = 'StructureSpecific',
  prdSe = 'Y',
  newEstPrdCnt = 3,
  # 최근수룩시점 개수
  prdInterval = 1,
  # 수룩시점 간격
  version = 'v2_1'
)
)

docParse <- xmlParse(res)

v_tbl_nm <- xmlToList(docParse)$Header$Name # 통계표명
docList <- xmlToList(docParse)$DataSet$Series

df_lists <- data.frame()

for (i in 1:length(docList)) {
  if (length(docList[i]$Obs) == 2) {
    df_lists <-
      rbind(df_lists, cbind(
        prd_de = paste(docList[i]$Obs[2]),
        dt = paste(docList[i]$Obs[1])
      ))
  }
}

# 파이차트 비율 라벨 값
pct <-
  round(as.numeric(df_lists$dt) / sum(as.numeric(df_lists$dt)) * 100, 1)
df_lists <- data.frame(df_lists, pct = pct)

# 파이차트
ggplot(df_lists, aes(
  x = factor(1),
  y = "",
  fill = factor(prd_de)
)) +
  geom_bar(stat = 'identity') +
  theme_void() +
  ggtitle(v_tbl_nm) +
  coord_polar('y', start = 0) +
  geom_text(aes(label = paste0(round(pct, 1), '%')),
            position = position_stack(vjust = 0.5))

```

■ 예제 소스(Python)

```
import matplotlib.pyplot as plt
import requests
from bs4 import BeautifulSoup

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 sdmx 데이터 가져오기
open_url = 'https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

#변수 지정 및 데이터 저장
labels = []
ratio = []
title = soup.find('common:name').text #차트제목

dataList = soup.find_all('obs')

for item in dataList:
    labels.append(item.get('time_period')) #년도 데이터
    ratio.append(item.get('obs_value')) #값/비율 데이터

#Pie차트 그리기
plt.pie(ratio, labels=labels, autopct='%1f%%')
plt.title(title)
plt.show()
```

■ 예제 소스 결과(표차트)

인구 가구 및 주택 - 읍면동(2015) 시군구(2016~) 단위 : 명

시점	총인구(전국)
2017	51422507
2018	51629512
2019	51779203

출처 : 「인구총조사」

■ 예제 소스(JSP)

```
<!--
```

KOSIS OpenAPI를 이용하여 통계표를 출력하는 예제입니다.

이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

- * 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.

```
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<xml:namespace prefix="v"/>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.6.1.min.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
$(document).ready(function(){
getSubList();
});

var strUnitId, strImlId, strC1;

/*****
* 통계자료 조회 함수
*****/
function getSubList() {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1",
data: "",
sync : true,
dataType: "xml",
success:function(object)
{

var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

// 항목, 분류, 단위의 명칭을 조회하기위해 각각의 코드를 변수에 저장
```



```

for(var i=0;i<object.documentElement.childNodes[1].childNodes[0].attributes.length;i++) {

if (object.documentElement.childNodes[1].childNodes[0].attributes[i].nodeName == "ITEM") {

    strImlId = object.documentElement.childNodes[1].childNodes[0].attributes[i].value;
} else if (object.documentElement.childNodes[1].childNodes[0].attributes[i].nodeName == "C_HJG") {

    strC1 = object.documentElement.childNodes[1].childNodes[0].attributes[i].value;
} else if (object.documentElement.childNodes[1].childNodes[0].attributes[i].nodeName == "UNIT") {

    strUnitId = object.documentElement.childNodes[1].childNodes[0].attributes[i].value;
}
}

var strTable = "";

// 조회된 결과를 이용하여 통계표 작성
strTable += "<p style='position:absolute; top:32px; left:450px;' id='unitId'></p>";
strTable += "<table cellpadding='0' cellspacing='0' class='TB02' summary='' style='margin-top:15px;width:500px;'>";
strTable += "<thead>";
strTable += "<tr id = 'theadId'>";

strTable += "</tr>";
strTable += "</thead>";
strTable += "<tbody>";

for(var i=0;i<mapdata.length;i++) {
    strTable += "<tr>";
    strTable += "<td class='al_C'>" + mapdata[i].attributes[1].value + "</td>";
    strTable += "<td class='al_C'>" + mapdata[i].attributes[0].value + "</td>";
    strTable += "</tr>";
}
strTable += "</tbody>";
strTable += "</table>";

document.getElementById("content").innerHTML = strTable;

fnGetTitle ();
fnGetUnit ();
fnGetIml ();
},
error: function(xhr,status,error){
    alert("error = " + error);
}
});
}

/*****
* 통계표 명칭 조회
*****/
function fnGetTitle () {

// ajax 통신을 위한 호출 함수

```

```
$.ajax({
  type: "GET",
  url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=TBL&orgId=101&tblId=DT_1B01003",
  data: "",
  sync : true,
  dataType: "xml",
  success:function(object)
  {

    var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

    // 조회된 결과를 이용하여 통계표 명칭 작성
    var strTable = document.getElementById("content").innerHTML;
    document.getElementById("content").innerHTML = "<h5>" + object.documentElement.childNodes[1].childNodes
    [0].childNodes[0].data + "</h5>" + strTable;

  },
  error: function(xhr,status,error){
    alert("error = " + error);
  }
});
}

/*****
* 항목 명칭 조회
*****/
function fnGetltn () {

  // ajax 통신을 위한 호출 함수
  $.ajax({
    type: "GET",
    url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=ITM&orgId=101&tblId=DT_1B01003&objId=ITEM&itmId=" + strltnId,
    data: "",
    sync : true,
    dataType: "xml",
    success:function(object)
    {
      // 분류 명칭 조회
      fnGetC1(object.documentElement.childNodes[1].childNodes[0].childNodes[4].childNodes[0].data);
    },
    error: function(xhr,status,error){
      alert("error = " + error);
    }
  });
}

/*****
* 분류 명칭 조회
*****/
function fnGetC1(itmNm) {
```

```
// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=ITM&orgId=101&tblId=DT_1B01003&objId=HJG&itmId=" + strC1,
data: "",
sync : true,
dataType: "xml",
success:function(object)
{
var strTable = "";

// 조회된 결과를 이용하여 항목, 분류 작성
strTable += "<th class='fst al_C' scope='col'>시점</th>";
strTable += "<th scope='col'>" + itmNm + "(" + object.documentElement.childNodes[1].childNodes[0].childNodes[4].childNodes[0].data + ")" + "</th>";

document.getElementById("theadId").innerHTML = strTable;

},
error: function(xhr,status,error){
alert("error = " + error);
}
});

}

/*****
* 단위 명칭 조회
*****/
function fnGetUnit () {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=UNIT&unitId=" + strUnitId,
data: "",
sync : true,
dataType: "xml",
success:function(object)
{

// 조회된 결과를 이용하여 단위 명칭 작성
var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

document.getElementById("unitId").innerHTML = "단위 : " + object.documentElement.childNodes[1].childNodes[0].childNodes[0].data;

},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}
```

```

}

</script>
</head>
<body>
<div id="content" style="padding-top:30px;"></div>
</html>

```

■ 예제 소스(R)

```

library(httr)
library(rvest)
library(XML)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

res <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'sdmx',
      apiKey = v_apiKey %>% I(),
      userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1', # 시계열 조회 키값
      jsonVD = 'Y',
      type = 'StructureSpecific',
      prdSe = 'Y',
      newEstPrdCnt = 3, # 최근수록시점 개수
      prdInterval = 1, # 수록시점 간격
      version = 'v2_1'
    )
  )

docParse <- xmlIParse(res)

docList <- xmlToList(docParse)$DataSet$Series

df_lists <- data.frame()

for (i in 1:length(docList)) {
  if (length(docList[i]$Obs) == 2) {
    df_lists <-
      rbind(df_lists, cbind(paste(docList[i]$Obs[2]), paste(docList[i]$Obs[1])))
  }
}

colnames(df_lists) <- c("시점", "수치")

```

View(df_lists)

■ 예제 소스(Python)

```
import matplotlib.pyplot as plt
import requests
from bs4 import BeautifulSoup

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 sdmx 데이터 가져오기
open_url = 'https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

dataList = soup.find_all('obs')

#변수 지정 및 데이터 저장
data = []

for item in dataList:
    value = []
    value.append(item.get('time_period')) #년도 데이터
    value.append(item.get('obs_value')) #값 데이터

    data.append(value)

#Table 만들기
fig, ax = plt.subplots(1,1)
column_labels=["시점", "총인구(전국)"]
ax.axis('tight')
ax.axis('off')
ax.table(cellText=data,colLabels=column_labels,colColours=["yellow"] * 2, loc="center", cellLoc='center')

plt.show()
```

2.2.3.4 SDMX(StructureSpecific)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do>

■ 입력 변수

● 자료등록 방법

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
userStatsId		String	사용자 등록 통계표	필수
type		String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
prdSe		String	수록주기	필수
시점기준	startPrdDe	String	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료	newEstPrdCnt	String	최근수록시점 개수	
기준	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수
version		String	결과값 구분	생략시 구버전으로 데이터 출력

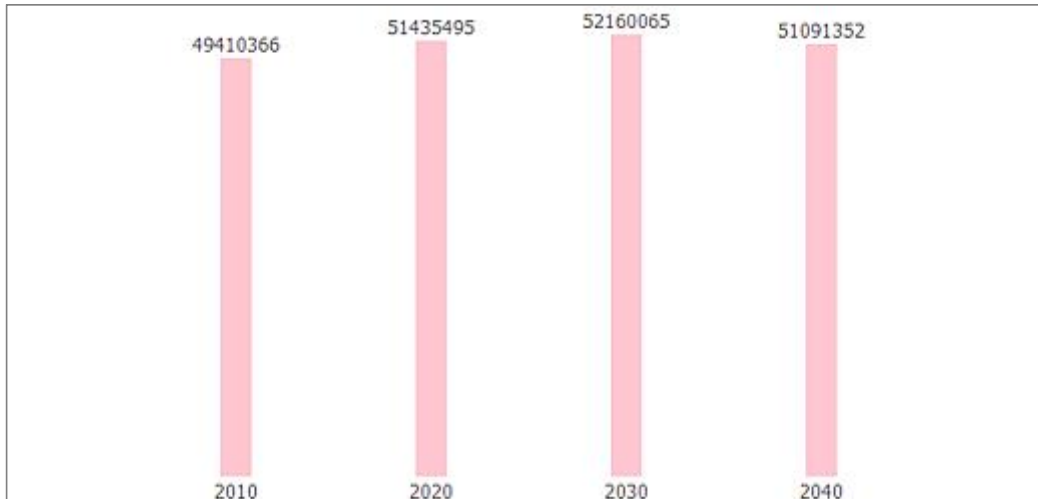
● 통계표선택 방법

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
orgId		String	기관 ID	필수
tblId		String	통계표 ID	필수
objL2 ~ objL8		String	분류2(두번째 분류코드) ~ 분류8(여덟째 분류코드)	선택
itmId		String	항목	필수
type		String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
prdSe		String	수록주기	필수
시점기준	startPrdDe	String	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료	newEstPrdCnt	String	최근수록시점 개수	
기준	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수
version		String	결과값 구분	생략시 구버전으로 데이터 출력

■ 출력 변수

항목명(영문)				항목설명
Header	ID			기관코드_통계표ID
	Name			통계표명
	Prepared			전송시간
	Sender	Id		전송기관
		Name		전송기관명
		Contact	Department	담당부서
			Telephone	담당부서 연락처
	Source			출처
Series	UNIT			단위
	ITEM			항목
	FIEQ			주기
	C_분류			C_분류 (8개 분류까지 가능)
	OBS	TIME_PERIOD		시점
		OBS_VALUE		수치자료

■ 예제 소스 결과(막대차트)



■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 막대차트를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의.jsp를 생성하여
```

호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.

```
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<xml:namespace prefix="v"/>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.6.1.min.js"></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide02/js/jquery.SimpleChart.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
$(document).ready(function(){
getSubList();
});

/*****
* 통계자료 조회 함수
*****/
function getSubList() {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1",
data: "",
async : true,
dataType: "xml",
success:function(object)
{

var data = object.documentElement.childNodes[1].childNodes[0].childNodes;

// 차트에 보여주기위해 값을 담고있을 배열변수를 선언
var aData = new Array();

// 데이터를 배열변수에 적재
for(var i=0; i < data.length; i++) {

aData[i] = [data[i].attributes[0].value.replace(/,/gi,""), data[i].attributes[1].value, 'pink'];
}
}
```



```
// 차트를 그리기위한 옵션을 정의
var options = {'BarSize': '20px', 'BarSpace': '2px', 'type' : 'horizontal', 'Font': '2px'}

// 차트를 화면에 출력
$("#chart2").SimpleChart(aData,options);
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

</script>
</head>
<body>
<div id="chart2" style="padding-top:30px;"></div>
</body>
</html>
```

■ 예제 소스(R)

```
library(httr)
library(rvest)
library(XML)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZOTi3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

res <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'sdmx',
      apiKey = v_apiKey %>% I(),
      userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1',
      # 시계열 조회 기간
      jsonVD = 'Y',
      type = 'StructureSpecific',
      prdSe = 'Y',
      newEstPrdCnt = 3,
      # 최근등록시점 개수
      prdInterval = 1,
      # 등록시점 간격
      version = 'v2_1'
    )
  )
```

```
docParse <- xmlParse(res)

tbl_title <- xmlToList(docParse)$Header$Name # 통계표명
docList <- xmlToList(docParse)$DataSet$Series

df_lists <- data.frame()

for (i in 1:length(docList)) {
  if (length(docList[i]$Obs) == 2) {
    df_lists <-
      rbind(df_lists, cbind(
        prd_de = paste(docList[i]$Obs[2]),
        dt = paste(docList[i]$Obs[1])
      ))
  }
}

# 바차트
ggplot(df_lists, aes(x = prd_de, y = dt, fill = prd_de)) + xlab("시점") + ylab("") + ggtitle(tbl_title) + geom_bar(stat = "identity")
```

■ 예제 소스(Python)

```
import matplotlib.pyplot as plt
import requests
from bs4 import BeautifulSoup

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 sdmx 데이터 가져오기
open_url = 'https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

#변수 지정 및 데이터 저장
xAxis = []
yAxis = []
title = soup.find('common:name').text #차트제목

dataList = soup.find_all('obs')
```

```

for item in dataList:
    xAxis.append(item.get('time_period')) #년도 데이터
    yAxis.append(int(item.get('obs_value'))) #값/비율 데이터

#Bar차트 그리기
plt.bar(xAxis, yAxis)
plt.title(title)

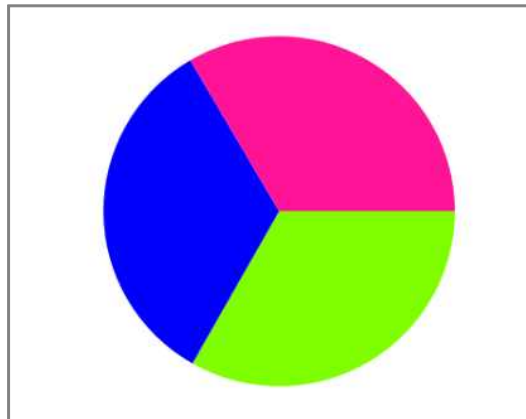
#y축 수치를 안보이게 하는 코드. 필요에 따라 선택하여 사용
plt.gca().axes.yaxis.set_visible(False)

#Bar의 가운데에 text로 수치 표시
for i, v in enumerate(xAxis):
    plt.text(v, yAxis[i], yAxis[i],
             fontsize = 9,
             color='blue',
             horizontalalignment='center',
             verticalalignment='bottom')

plt.show()

```

■ 예제 소스 결과(파이차트)



■ 예제 소스(JSP)

```

<!--
KOSIS OpenAPI를 이용하여 파이차트를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->

```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<xml:namespace prefix="v"/>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.6.1.min.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
$(document).ready(function(){
getSubList();
});

/*****
* 통계자료 조회 함수
*****/
function getSubList() {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1",
data: "",
async : true,
dataType: "xml",
success:function(object)
{

var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

// 차트에 보여주기위해 값을 담고있을 배열변수를 선언
var data = new Array(mapdata.length);

// 데이터를 배열변수에 적재
for(var i=0;i<mapdata.length;i++) {
data[i] = mapdata[i].attributes[0].value;
}

// 파이차트를 출력하기위한 JavaScript Start
var canvas = document.getElementById("chartcanvas");
var context = canvas.getContext("2d");
var sw = canvas.width;
```

```

var sh = canvas.height;
var PADDING=100;

//Browser별 색상 lawngreen, blue, deeppink, aquamarine3, magenta, gold
var colors = ["#7cfc00", "#0000ff", "#ff1493", "#66CDAA", "#ff00ff", "#FFD700"];

var center_X=sw/2; //원의 중심 x 좌표
var center_Y=sh/2; //원의 중심 y 좌표
// 두 계산값 중 작은 값을 원의 반지름으로 설정
var radius = Math.min(sw-(PADDING*2), sh-(PADDING*2)) / 2;
var angle = 0;
var total = 0;

for (var i in data) { total += Number(data[i]); } //데이터(data)의 총합 계산

for (var i = 0; i < data.length; i++) {

context.fillStyle = colors[i]; //생성되는 부분의 채울 색 설정
context.beginPath();
context.moveTo(center_X, center_Y); //원의 중심으로 이동
context.arc(center_X, center_Y, radius, angle, angle +(Math.PI*2*(data[i]/total)));
context.lineTo(center_X,center_Y);
context.fill();
angle += Math.PI*2*(data[i]/total);
}
// 파이차트를 출력하기 위한 JavaScript End
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}
</script>
</head>
<body>
<canvas id="chartcanvas" width="500" height="400"></canvas>
</html>

```

■ 예제 소스(R)

```

library(httr)
library(rvest)
library(XML)
library(ggplot2)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

res <-

```

```

GET(
  url = baseurl,
  query = list(
    method = 'getList',
    format = 'sdmx',
    apiKey = v_apiKey %>% I(),
    userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1',
    # 시계열 조회 기간
    jsonVD = 'Y',
    type = 'StructureSpecific',
    prdSe = 'Y',
    newEstPrdCnt = 3,
    # 최근수룩시점 개수
    prdInterval = 1,
    # 수룩시점 간격
    version = 'v2_1'
  )
)

docParse <- xmlParse(res)

v_tbl_nm <- xmlToList(docParse)$Header$Name # 통계표명
docList <- xmlToList(docParse)$DataSet$Series

df_lists <- data.frame()

for (i in 1:length(docList)) {
  if (length(docList[i]$Obs) == 2) {
    df_lists <-
      rbind(df_lists, cbind(
        prd_de = paste(docList[i]$Obs[2]),
        dt = paste(docList[i]$Obs[1])
      ))
  }
}

# 파이차트 비율 라벨 값
pct <-
  round(as.numeric(df_lists$dt) / sum(as.numeric(df_lists$dt)) * 100, 1)
df_lists <- data.frame(df_lists, pct = pct)

# 파이차트
ggplot(df_lists, aes(
  x = factor(1),
  y = "",
  fill = factor(prd_de)
)) +
  geom_bar(stat = 'identity') +
  theme_void() +
  ggtitle(v_tbl_nm) +
  coord_polar('y', start = 0) +
  geom_text(aes(label = paste0(round(pct, 1), '%')),

```

```
position = position_stack(vjust = 0.5))
```

■ 예제 소스(Python)

```
import matplotlib.pyplot as plt
import requests
from bs4 import BeautifulSoup

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 sdmx 데이터 가져오기
open_url = 'https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

#변수 지정 및 데이터 저장
labels = []
ratio = []
title = soup.find('common:name').text #차트제목

dataList = soup.find_all('obs')

for item in dataList:
    labels.append(item.get('time_period')) #년도 데이터
    ratio.append(item.get('obs_value')) #값/비율 데이터

#Pie차트 그리기
plt.pie(ratio, labels=labels, autopct='%1f%%')
plt.title(title)
plt.show()
```

■ 예제 소스 결과(표차트)

인구 가구 및 주택 - 읍면동(2015) 시군구(2016~)		단위 : 명
시점	총인구(전국)	
2017	51422507	
2018	51629512	
2019	51779203	

출처 : 「인구총조사」

■ 예제 소스(JSP)

```

<!--
KOSIS OpenAPI를 이용하여 통계표를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<xml:namespace prefix="v"/>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.6.1.min.js"></script>
<script type="text/javascript" language="JavaScript">

// window onload 되었을때 실행 함수
$(document).ready(function(){
getSubList();
});

var strUnitId, strItrmId, strC1;

/*****
* 통계자료 조회 함수
*****/
function getSubList() {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1",
data: "",
sync : true,
dataType: "xml",
success:function(object)
{

var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

```



```
// 항목, 분류, 단위의 명칭을 조회하기위해 각각의 코드를 변수에 저장
for(var i=0;i<object.documentElement.childNodes[1].childNodes[0].attributes.length;i++) {

if (object.documentElement.childNodes[1].childNodes[0].attributes[i].nodeName == "ITEM") {

strItemld = object.documentElement.childNodes[1].childNodes[0].attributes[i].value;
} else if (object.documentElement.childNodes[1].childNodes[0].attributes[i].nodeName == "C_HJG") {

strC1 = object.documentElement.childNodes[1].childNodes[0].attributes[i].value;
} else if (object.documentElement.childNodes[1].childNodes[0].attributes[i].nodeName == "UNIT") {

strUnitld = object.documentElement.childNodes[1].childNodes[0].attributes[i].value;
}
}

var strTable = "";

// 조회된 결과를 이용하여 통계표 작성
strTable += "<p style='position:absolute; top:32px; left:450px;' id='unitld'></p>";
strTable += "<table cellpadding='0' cellspacing='0' class='TB02' summary='' style='margin-top:15px;width:500px;'>";
strTable += "<thead>";
strTable += "<tr id = 'theadld'>";

strTable += "</tr>";
strTable += "</thead>";
strTable += "<tbody>";

for(var i=0;i<mapdata.length;i++) {
strTable += "<tr>";
strTable += "<td class='al_C'>" + mapdata[i].attributes[1].value + "</td>";
strTable += "<td class='al_C'>" + mapdata[i].attributes[0].value + "</td>";
strTable += "</tr>";
}
strTable += "</tbody>";
strTable += "</table>";

document.getElementById("content").innerHTML = strTable;

fnGetTitle ();
fnGetUnit ();
fnGetItm ();
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

/*****
* 통계표 명칭 조회
*****/
function fnGetTitle () {
```

```
// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=TBL&orgId=101&tblId=DT_1B01003",
data: "",
sync : true,
dataType: "xml",
success:function(object)
{

var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

// 조회된 결과를 이용하여 통계표 명칭 작성
var strTable = document.getElementById("content").innerHTML;
document.getElementById("content").innerHTML = "<h5>" + object.documentElement.childNodes[1].childNodes
[0].childNodes[0].data + "</h5>" + strTable;

},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

/*****
* 항목 명칭 조회
*****/
function fnGetItm () {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=ITM&orgId=101&tblId=DT_1B01003&objId=ITEM&itmId=" + strItmId,
data: "",
sync : true,
dataType: "xml",
success:function(object)
{
// 분류 명칭 조회
fnGetC1(object.documentElement.childNodes[1].childNodes[0].childNodes[4].childNodes[0].data);
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

/*****
* 분류 명칭 조회
*****/
function fnGetC1(itmNm) {
```

```
// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=ITM&orgId=101&tblId=DT_1B01003&objId=HJG&itmId=" + strC1,
data: "",
sync : true,
dataType: "xml",
success:function(object)
{
var strTable = "";

// 조회된 결과를 이용하여 항목, 분류 작성
strTable += "<th class='fst aLC' scope='col'>시점</th>";
strTable += "<th scope='col'>" + itmNm + "(" + object.documentElement.childNodes[1].childNodes[0].childNodes[4].childNodes[0].data + ")" + "</th>";

document.getElementById("theadId").innerHTML = strTable;

},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}

/*****
* 단위 명칭 조회
*****/
function fnGetUnit () {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/statisticsData.do?method=getMeta&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=xml&type=UNIT&unitId=" + strUnitId,
data: "",
sync : true,
dataType: "xml",
success:function(object)
{

// 조회된 결과를 이용하여 단위 명칭 작성
var mapdata = object.documentElement.childNodes[1].childNodes[0].childNodes;

document.getElementById("unitId").innerHTML = "단위 : " + object.documentElement.childNodes[1].childNodes[0].childNodes[0].data;

},
error: function(xhr,status,error){
alert("error = " + error);
}
}
```

```
});
}

</script>
</head>
<body>
<div id="content" style="padding-top:30px;"></div>
</html>
```

■ 예제 소스(R)

```
library(httr)
library(rvest)
library(XML)

rm(list = ls()) # 모든변수 초기화

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청URL

res <-
  GET(
    url = baseurl,
    query = list(
      method = 'getList',
      format = 'sdmx',
      apiKey = v_apiKey %>% I(),
      userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1', # 시계열 조회 키값
      jsonVD = 'Y',
      type = 'StructureSpecific',
      prdSe = 'Y',
      newEstPrdCnt = 3, # 최근수록시점 개수
      prdInterval = 1, # 수록시점 간격
      version = 'v2_1'
    )
  )

docParse <- xmlIParse(res)

docList <- xmlToList(docParse)$DataSet$Series

df_lists <- data.frame()

for (i in 1:length(docList)) {
  if (length(docList[i]$Obs) == 2) {
    df_lists <-
      rbind(df_lists, cbind(paste(docList[i]$Obs[2]), paste(docList[i]$Obs[1])))
  }
}

colnames(df_lists) <- c("시점", "수치")
```

View(df_lists)

■ 예제 소스(Python)

```
import matplotlib.pyplot as plt
import requests
from bs4 import BeautifulSoup

# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 sdmx 데이터 가져오기
open_url = 'https://kosis.kr/openapi/statisticsData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&newEstPrdCnt=3&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

dataList = soup.find_all('obs')

#변수 지정 및 데이터 저장
data = []

for item in dataList:
    value = []
    value.append(item.get('time_period')) #년도 데이터
    value.append(item.get('obs_value')) #값 데이터

    data.append(value)

#Table 만들기
fig, ax = plt.subplots(1,1)
column_labels=["시점", "총인구(전국)"]
ax.axis('tight')
ax.axis('off')
ax.table(cellText=data,colLabels=column_labels,colColours=["yellow"] * 2, loc="center", cellLoc='center')

plt.show()
```

2.3 대용량 통계자료

- 통계표의 수치자료 및 메타정보 제공을 위한 OpenAPI로, 통계표 전체, 분류 전체(일부), 항목 전체(일부)를 선택적으로 요청합니다.

2.3.1 특징

- 통계표의 수치자료를 다중계열, 여러시점으로 제공
- 통계표의 수록정보, 분류/항목, 출처, 단위 등 메타정보 제공
- 자료 제공형태 : SDMX (DSD, Generic, StructureSpecific), CSV

2.3.2 서비스 활용

2.3.2.1 자료등록

- 개발가이드 > 대용량 통계자료 > 자료등록 URL생성 > 자료등록
'작성기관', '통계조사명', '통계표명' 등을 입력하여 사용하고자 하는 자료를 조회한 뒤 조회결과에서 등록할 자료의 ①사용여부 항목을 선택하고 ②통계표 등록 버튼을 누릅니다. ③통계표조회 버튼을 클릭하면 해당 자료의 통계표를 볼 수 있습니다.

대용량통계자료

Home > 개발 가이드 > 대용량통계자료

인쇄

대용량 통계자료 API를 이용하기 위한 요청번호, 출력정보, 샘플데이터를 제공합니다. 출력포맷으로 SDMX, XLS를 제공합니다. 데이터 출력건수가 40,000개 이상인 경우는 SDMX 제공은 불가능하고, XLS로만 가능합니다. 또한, 데이터 출력건수가 20만개 이상인 경우는 XLS로도 제공이 불가능하므로 이용건수를 줄여 사용하시기 바랍니다.

SDMX

XLS

자료등록 URL 생성

·자료 등록

·등록된 자료

작성기관

통계청

통계표명

인구밀도(인구주택총조사기준)

분류

통계청 (101)

분류값/항목

통계조사명

인구총조사

통계표ID

DT_1808024

주기

전체

등록한 표만 보기

검색

기관명	통계표명	통계표ID	통계조사명	수록기간	사용여부	통계표조회
통계청 (101)	인구밀도(인구주택총조사기준)	DT_1808024	인구총조사	5년 1966~2021	① <input checked="" type="checkbox"/>	③
통계청 (101)	행정구역/가구규모/성별 통근·통학인구(12...	DT_1890001	인구총조사	5년 1995~1995	<input checked="" type="checkbox"/>	
통계청 (101)	대도시/통근통학/성/이통고통수단별 통근·...	DT_1890003	인구총조사	5년 1995~1995	<input checked="" type="checkbox"/>	
통계청 (101)	시도/소도시간/이통고통수단별 통근·통학인...	DT_1890005	인구총조사	5년 1995~1995	<input type="checkbox"/>	
통계청 (101)	시도/절류형태/이통고통수단별 통근·통학인...	DT_1890007	인구총조사	5년 1995~1995	<input type="checkbox"/>	
통계청 (101)	시도/성/직업/이통고통수단별 통근·통학인...	DT_1890008	인구총조사	5년 1995~1995	<input type="checkbox"/>	
통계청 (101)	시도/성/산업/이통고통수단별 통근·통학인...	DT_1890009	인구총조사	5년 1995~1995	<input type="checkbox"/>	
통계청 (101)	시도/성/소도시간/학교별 통근·통학인구(12...	DT_189000A	인구총조사	5년 1995~1995	<input type="checkbox"/>	
통계청 (101)	현거주지/성/특적지별 통근·통학인구(서울,...	DT_189000D	인구총조사	5년 1995~1995	<input type="checkbox"/>	
통계청 (101)	현거주지/성/특적지별 통근·통학인구(부산,...	DT_189000E	인구총조사	5년 1995~1995	<input type="checkbox"/>	

◀ 1 2 3 4 5 6 7 8 9 10 ▶

②

통계표등록

2.3.2.2 등록된 자료

■ 개발가이드 > 대용량 통계자료 > 자료등록 URL생성 > 등록된 자료

자료등록의 통계표 등록을 마치고 등록된 자료 탭을 클릭하면, 이용자가 등록한 자료들의 목록이 나타나며, 등록된 자료 중 URL생성을 원하는 자료의 ①URL 생성 버튼을 누릅니다. ②통계표조회 버튼을 클릭하면 해당 자료의 통계표를 볼 수 있습니다.

대용량통계자료

Home > 개발 가이드 > 대용량통계자료

인쇄

대용량 통계자료 API를 이용하기 위한 요청변수, 출력정보, 샘플데이터를 제공합니다. 출력포맷으로 SDMX, XLS를 제공합니다. 데이터 출력건수가 40,000개 이상인 경우는 SDMX 제공은 불가하고, XLS로만 가능합니다. 또한, 데이터 출력건수가 20만개 이상인 경우는 XLS로도 제공이 불가하므로 이용건수를 줄여 사용하시기 바랍니다.

SDMX

XLS

자료등록 URL 생성

·자료 등록

·등록된 자료

작성기관

통계청

통계조사명

자동차등록현황보

통계표명

통계표ID

DT_MLTM_5498

분류

주기

전체

검색

기관명	통계표명	통계표ID	통계조사명	수록기간	신청일자	통계표조회	URL생성
국토교통부 (116)	자동차등록대수현황 시도별	DT_MLTM_5498	자동차등록현황보	2011.01~2022.08	2022-07-28 13:54:33	②	① URL생성
통계청 (101)	행정구역(시군구)별 주민등록세대수	DT_1804083	주민등록인구현황	1992~2022.08	2022-06-29 09:55:44		URL생성
국민연금공단 (322)	65세이상 인구대비 성별·지역별 수급...	DT_32202_B006	국민연금통계	년 2012~2021	2022-06-22 15:01:39		URL생성

■ 개발가이드 > 대용량 통계자료 > 자료등록 URL생성 > 등록된 자료 > 삭제

①버튼을 클릭한 후 하단에 조회되는 사용자 생성 URL 목록을 ②삭제할 수 있습니다.

대용량통계자료

Home > 개발 가이드 > 대용량통계자료

인쇄

대용량 통계자료 API를 이용하기 위한 요청변수, 출력정보, 샘플데이터를 제공합니다. 출력포맷으로 SDMX, XLS를 제공합니다. 데이터 출력건수가 40,000개 이상인 경우는 SDMX 제공은 불가하고, XLS로만 가능합니다. 또한, 데이터 출력건수가 20만개 이상인 경우는 XLS로도 제공이 불가하므로 이용건수를 줄여 사용하시기 바랍니다.

SDMX

XLS

자료등록 URL 생성

개발가이드 다운로드

·자료 등록

·등록된 자료

작성기관

통계조사명

통계표명

통계표ID

분류

주기

전체

검색

기관명	통계표명	통계표ID	통계조사명	수록기간	신청일자	통계표조회	URL생성
한국주택금융공사 (321)	주택담보대출 금액	DT_321003N_001	주택금융정보금자리론실태조사	년 2016~2016	2022-10-04 13:26:15		URL생성

사용자 생성 URL 목록

활용자료명	URL 상세 정보	삭제	URL보기
주택담보대출 금액	주택담보대출 금액 * 전체 * 사례수	② 삭제	URL보기

2.3.2.3 URL생성

■ 개발가이드 > 통계자료 > 자료등록 URL생성 > 등록된 자료 > URL 생성

URL생성 단계에서는 URL생성 조건 설정의 '활용 자료명', '분류/항목선택'을 입력 후 ①URL생성 버튼을 누르면 URL생성 상세조건 화면으로 이동 후 URL이 하단에 생성됩니다. URL생성 상세조건 화면에서 상세설정 후 ②URL보기, 결과값보기 버튼을 클릭하여 페이지 하단에서 결과를 확인 할 수 있고, ③URL복사 버튼을 누르면 생성된 URL이 클립보드에 복사됩니다. 또한 ④목록 버튼을 누르면 등록된 자료화면으로 돌아갑니다.

대용량통계자료

Home > 개발가이드 > 대용량통계자료

인쇄

대용량 통계자료 API를 이용하기 위한 요청변수, 출력정보, 샘플데이터를 제공합니다. 출력포맷으로 SDMX, XLS를 제공합니다. 데이터 출력건수가 40,000개 이상인 경우는 SDMX 제공은 불가하고, XLS로만 가능합니다. 또한, 데이터 출력건수가 20만개 이상인 경우는 XLS로도 제공이 불가하므로 이용건수를 줄여 사용하시기 바랍니다.

SDMX

XLS

자료등록 URL 생성

자료 등록

등록된 자료

URL생성

URL생성 조건 설정

통계표명	자동차등록대수현황 시도별	통계표 ID	DT_MLTM_5498
활용 자료명	자동차등록대수현황 시도별		

분류/항목선택

분류

항목

시도

개별

선택

레벨별

1

/1

시군구

개별

선택

레벨별

1

/1

레벨01

개별

선택

레벨별

1

/1

관통

자가용

영업용

계

전체선택

해제

1

URL생성

취소

대용량통계자료

[Home](#) > [개발 가이드](#) > [대용량통계자료](#) [인쇄](#)

대용량 통계자료 API를 이용하기 위한 요청변수, 출력정보, 샘플데이터를 제공합니다. 출력포맷으로 SDMX, XLS를 제공합니다.
 데이터 출력건수가 40,000개 이상인 경우는 SDMX 제공은 불가하고, XLS로만 가능합니다.
 또한, 데이터 출력건수가 20만개 이상인 경우는 XLS로도 제공이 불가하므로 이용건수를 줄여 사용하시기 바랍니다.

SDMX

XLS

자료등록 URL 생성

[*자료 등록](#)
[*등록된 자료](#)

URL생성

URL생성 조건 설정

통계표명	자동차등록대수현황 시도별	통계코드	DT_MLTM_5498
------	---------------	------	--------------

URL생성 상세조건 정보

출력형태 설정

데이터 포맷

☐ XLS
 ☒ SDMX

출력 형태

☐ DSD(자료구조정의)
 ☐ Generic(일반 수치자료)
 ☒ StructureSpecific(구조화된 수치자료)

조회기간 설정

☐ 기간설정
 ☒ 최근시점기준

최근시점부터

주기
 최근 개 자료

수색시점간격

최근시점 조정에 사용할 수 있는 항목으로 현재 시점부터 사용합기문시점까지의 간격을 입력
 ex) 2019, 2017, 2015 등 2개 시점 간격으로 추출시 [2] 입력

활동자르명	URL 상세 정보	조회건수	URL보기/결과값보기
자동차등록대수현황 시도별	계 + 서울 + 계 + 승용	대응중	URL보기 결과값보기

생성된 URL

https://kosis.kr/openapi/statisticsBigData.do?method=getList&apiKey=WWVnZDgyNDVhZTJhZTk2MmE0OGVhYjY0Yzg1MzowY20=&format=sdmx&userStatsId=bang3178/116/DT_MLTM_5498/3/5/20220921101508&pe=StructureSpecific&prdSe=M&newEstPrdCnt=38&version=V2_1

URL복사

데이터 보기

결과값보기

```
<?xml version='1.0' encoding='UTF-8' standalone='yes' ?><message:StructureSpecificTimeSeriesData
xmlns:extr='urn:sdmx:org.sdmx.infomodel.datastructure.DataStructure=116_DT_MLTM_5498:ObsLevelDim:TIME_PERIOD'
xmlns:message='http://www.sdmx.org/resources/sdmx/schemas/v2_1/message'
xmlns:data='http://www.sdmx.org/resources/sdmx/schemas/v2_1/data/structureSpecific'
xmlns:common='http://www.sdmx.org/resources/sdmx/schemas/v2_1/common' xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:schemaLocation='http://www.sdmx.org/resources/sdmx/schemas/v2_1/message ../../schemas/SDMXMessage.xsd
urn:sdmx:org.sdmx.infomodel.datastructure.DataStructure=116_DT_MLTM_5498:ObsLevelDim:TIME_PERIOD ec6_exr_ng.ts.xsd'>
<message:Header>
<message:ID>116_DT_MLTM_5498</message:ID>
<message:Name>자동차등록대수현황 시도별 (월 2011.01~2022.08)</message:Name>
<message:Test>false</message:Test>
<message:Prepared>2022-09-21 10:15:27</message:Prepared>
<message:Sender>KOSIS</message:Sender>
<message:Contact></message:Contact>
<message:Department>국토교통부 국토교통부 교통정책실
국립교통정책 자문위원회</message:Department>
<message:Telephone>044-201-3861</message:Telephone>
<message:Contact></message:Contact>
</message:Header>
<message:Structure>
<message:DataSet data:structureRef='116_DT_MLTM_5498' data:dataScope='DataStructure'>
<Series FREQ='M' ITEM='19103879449T4'
</message:DataSet>
</message>
```

4

복합

2.3.3 활용방법

2.3.3.1 SDMX(DSD)

■ 호출 URL : <http://kosis.kr/openapi/statisticsBigData.do>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
userStatsId	String	사용자 등록 통계표	필수
type	String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
format	String	결과 유형(json, sdmx)	필수
version	String	결과값 구분	생략시 구버전으로 데이터 출력

■ 출력 변수

항목명(영문)						항목설명	
Header	ID		기관코드_통계표ID				
	Name		통계표명				
	Prepared		전송시간				
	Sender	Id		전송기관			
		Name		전송기관명			
		Contact	Department		담당부서		
			Telephone		담당부서 연락처		
	Source		출처				
Codelist	Codelist	Id		코드리스트ID			
		Name		코드리스트명			
		Description		코드리스트영문명			
		Code	Id		코드ID		
			Name		코드명		
Concepts	ConceptsScheme	Id		컨셉스키마ID			
		Name		컨셉스키마명			
		Description		컨셉스키마영문명			
		concept	Id		컨셉ID		
			Name		컨셉명		
			Description		컨셉영문명		
DataStructures	DataStructure	Id		통계표ID			
		Name		통계표명			
		DataStructureComponents	Dimension	Id		디멘전Id	
				ConceptIdentity	Id	Id	

2.3.3.2 SDMX(Generic)

- 호출 URL : <http://kosis.kr/openapi/statisticsBigData.do>
- 입력 변수

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
userStatsId		String	사용자 등록 통계표	필수
type		String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
prdSe		String	수록주기	필수
시점기준	startPrdDe	String	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료	newEstPrdCnt	String	최근수록시점 개수	
기준	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수
version		String	결과값 구분	생략시 구버전으로 데이터 출력

- 출력 변수

항목명(영문)				항목설명
Header	ID			기관코드_통계표ID
	Name			통계 표명
	Prepared			전송시간
	Sender	Id		전송기관
		Name		전송기관명
		Contact	Department	담당부서
			Telephone	담당부서 연락처
Source			출처	
Series	SeriesKey	Value	Id	시리즈키ID
			value	시리즈키값
	Obs	ObsDimension	Value	시점
		ObsValue	Value	수치자료값

2.3.3.3 SDMX(StructureSpecific)

■ 호출 URL : <http://kosis.kr/openapi/statisticsBigData.do>

■ 입력 변수

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
userStatsId		String	사용자 등록 통계표	필수
type		String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
prdSe		String	수록주기	필수
시점기준	startPrdDe	String	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료	newEstPrdCnt	String	최근수록시점 개수	
기준	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수
version		String	결과값 구분	생략시 구버전으로 데이터 출력

■ 출력 변수

항목명(영문)					항목설명
Header	ID				기관코드_통계표ID
	Name				통계표명
	Prepared				전송시간
	Sender	Id			전송기관
		Name			전송기관명
		Contact	Department		담당부서
			Telephone		담당부서 연락처
Source				출처	
Series	UNIT				단위
	ITEM				항목
	FREQ				주기
	분류				분류
	Obs	ObsDimension	Value		시점
		ObsValue	Value		수치자료값

2.3.3.4 XLS

■ 호출 URL : <http://kosis.kr/openapi/statisticsBigData.do>

■ 입력 변수

항목명(영문)	변수타입	항목설명		비고
apiKey	String	발급된 인증키		필수
userStatsId	String	사용자 등록 통계표		필수
prdSe	String	수록주기		필수
시점기준	startPrdDe	String	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료기준	newEstPrdCnt	String	최근수록시점 개수	
	prdInterval	String	수록시점 간격	
format	String	결과 유형(json, sdmx)		필수

■ 출력 결과 예시

	A	B	C	D	E	F
1	행정구역별	연령별	항목	2038	2039	2040
2	전국	계	추계인구(명)	51,470,244	51,291,401	51,091,352
3	전국	계	추계인구(남)(명)	25,474,360	25,375,247	25,264,681
4	전국	계	추계인구(여)(명)	25,995,884	25,916,154	25,826,671
5	서울특별시	계	추계인구(명)	10,003,809	9,965,542	9,924,373
6	서울특별시	계	추계인구(남)(명)	4,842,375	4,822,073	4,800,386
7	서울특별시	계	추계인구(여)(명)	5,161,434	5,143,469	5,123,987
8	부산광역시	계	추계인구(명)	3,059,995	3,037,918	3,014,946
9	부산광역시	계	추계인구(남)(명)	1,483,856	1,472,777	1,461,266
10	부산광역시	계	추계인구(여)(명)	1,576,139	1,565,141	1,553,680
11	대구광역시	계	추계인구(명)	2,253,057	2,237,153	2,220,439
12	대구광역시	계	추계인구(남)(명)	1,102,911	1,094,925	1,086,547
13	대구광역시	계	추계인구(여)(명)	1,150,146	1,142,228	1,133,892
14	인천광역시	계	추계인구(명)	3,052,428	3,045,109	3,036,476
15	인천광역시	계	추계인구(남)(명)	1,517,059	1,512,525	1,507,274
16	인천광역시	계	추계인구(여)(명)	1,535,369	1,532,584	1,529,202
17	광주광역시	계	추계인구(명)	1,455,552	1,446,771	1,437,531
18	광주광역시	계	추계인구(남)(명)	722,971	718,565	713,901
19	광주광역시	계	추계인구(여)(명)	732,581	728,206	723,630
20	대전광역시	계	추계인구(명)	1,578,818	1,573,165	1,566,886
21	대전광역시	계	추계인구(남)(명)	786,630	783,787	780,638
22	대전광역시	계	추계인구(여)(명)	792,188	789,378	786,248
23	울산광역시	계	추계인구(명)	1,073,971	1,066,258	1,058,153
24	울산광역시	계	추계인구(남)(명)	554,948	550,901	546,671
25	울산광역시	계	추계인구(여)(명)	519,023	515,357	511,482
26	경기도	계	추계인구(명)	13,378,824	13,341,670	13,298,132
27	경기도	계	추계인구(남)(명)	6,679,141	6,658,192	6,633,623
28	경기도	계	추계인구(여)(명)	6,699,683	6,683,478	6,664,509
29	강원도	계	추계인구(명)	1,561,651	1,559,432	1,556,371
30	강원도	계	추계인구(남)(명)	782,552	781,129	779,285
31	강원도	계	추계인구(여)(명)	779,099	778,303	777,086
32	충청북도	계	추계인구(명)	1,714,070	1,712,815	1,710,679
33	충청북도	계	추계인구(남)(명)	855,054	853,850	852,172
34	충청북도	계	추계인구(여)(명)	859,016	858,965	858,507
35	충청남도	계	추계인구(명)	2,408,879	2,409,008	2,407,883
36	충청남도	계	추계인구(남)(명)	1,226,624	1,226,193	1,225,061
37	충청남도	계	추계인구(여)(명)	1,182,255	1,182,815	1,182,822
38	전라북도	계	추계인구(명)	1,782,831	1,778,867	1,774,171
39	전라북도	계	추계인구(남)(명)	870,262	867,693	864,708
40	전라북도	계	추계인구(여)(명)	912,569	911,174	909,463

2.4 통계설명

- 통계조사에 대한 설명자료 제공을 위한 OpenAPI입니다.

2.4.1 특징

- 자료 제공형태 : XML, JSON

2.4.2 서비스 활용

2.4.2.1 URL생성

- 개발가이드 > 통계설명 > URL생성

URL생성 단계에서는 URL생성 조건 설정의 '통계조사', '설명항목', '설명자료' 를 입력 후 ①URL 복사, 결과값보기 버튼을 클릭하여 페이지 하단에서 결과를 확인 할 수 있습니다.

통계설명

Home > 개발 가이드 > 통계설명
인쇄

통계설명 API를 이용하기 위한 요청변수, 출력정보, 샘플데이터를 제공합니다. 출력포맷으로 JSON과 XML을 제공합니다.

JSON

XML

URL 생성

URL생성 조건 설정

통계조사
인구총조사

☒ 조사명
☒ 조사목적
☐ 공표주기
☐ 조사대상 지역
☐ 공표방법 및 URL
☐ 주요 용어설명
☐ 승인일자

☒ 통계종류
☒ 조사주기
☐ 연혁지
☐ 조사단위 및 조사대상규모
☐ 조사대상기간 및 조사기호시법
☐ 지표 수집방법
☐ 통계종류

☒ 계속여부
☒ 조사체계
☐ 통계(활동)분야·실태
☐ 적용분류
☐ 공표주기
☐ 조사연혁

☒ 법적근거
☒ 공표법위
☐ 조사대상 범위
☐ 조사항목
☐ 자료이용자 유의사항
☐ 승인번호

☐ 전체

설명자료
xml

1
URL 복사
결과값보기

생성된 URL

```

https://kosis.kr/openapi/statisticsExpData.do?
method=getList&aoiKey=NWVhZDgyNDVhZTJhZTk2MmE0OGVhYjY0YzgtMzowY2Q=&statId=1962001&format=xml&jsonVD=Y&jsonMVD=Y&metaltn=statsNm+statsKind+statsContinue+bas
isLaw+writingPurpo+statsPeriod+writingSystem+pubExtent

```

항목	설명
조사명	인구총조사
통계종류	지정통계 / 가공통계
계속여부	계속통계
법적근거	인구주택총조사 규칙(기획재정부령 제810호, 2020.10.26. 일부개정)
조사목적	인구규모, 분포 및 구조와 주택에 관한 제 특성을 파악하여 각종 정책입안의 기초자료로 활용, 각종 가구관련 경상조사 표본틀로 활용
조사주기	1년
조사체계	전수 ; 공공기관 및 사법기관 → 통계청 표본 ; 조사원 → 구·시·군 → 시·도 → 통계청
공표법위	통계법

결과값보기

```

<?xml version="1.0" encoding="UTF-8" ?><response><Header><id>1962001</id><Prepared>2022-09-21 10:16:28</Prepared><Sender><id>K0813</id><Name>통계청</Name></Header><Header><Structures><statsNm>인구총조사</statsNm><writingPurpo>인구규모, 분포 및 구조와 주택에 관한 제 특성을 파악하여 각종 정책입안의 기초자료로 활용, 각종 가구관련 경상조사 표본틀로 활용</writingPurpo><statsKind>지정통계 / 가공통계</statsKind><pubExtent>통계연</pubExtent><statsPeriod>1년</statsPeriod><statsContinue>계속통계</statsContinue><basisLaw>인구주택총조사 규칙(기획재정부령 제810호, 2020.10.26. 일부개정)</basisLaw><writingSystem>전수 ; 공공기관 및 사법기관 → 통계청 표본 ; 조사원 → 구·시·군 → 시·도 → 통계청</writingSystem></Structures></response>

```

KOSIS 공유서비스 개발가이드

110

2.4.3 활용방법

2.4.3.1 JSON

- 호출 URL : <http://kosis.kr/openapi/statisticsExpData.do>
- 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
statId * orgId(기관ID) + tblId(통계표ID) 로도 가능	String	통계조사 ID	필수
metaltn	String	요청 항목	필수 전체 - All 조사명-statsNm 통계종류-statsKind 계속여부-statsContinue 법적근거-basisLaw 조사목적-writingPurps 조사주기-statsPeriod 조사체계-writingSystem 공표범위-pubExtent 공표주기-pubPeriod 연락처-writingTel 통계(활용)분야·실태-statsField 조사 대상범위-examinObjrange 조사 대상지역-examinObjArea 조사단위 및 조사대상규모-josaUnit 적용분류-applyGroup 조사항목-josaltn 공표주기-pubPeriod 공표범위-pubExtent 공표방법 및 URL-publicMth 조사대상기간 및 조사기준시점-examinTrgetPd 자료이용자 유의사항 -dataUserNote 주요 용어해설-mainTermExpl 자료 수집방법-dataCollectMth 조사연혁-examinHistory 승인번호-confmNo 승인일자-confmDt 통계종류-statsEnd
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
statsNm	조사명
statsKind	통계종류
statsContinue	계속여부
basisLaw	법적근거
writingPurps	조사목적
statsPeriod	조사주기
writingSystem	조사체계
pubExtent	공표범위
pubPeriod	공표주기
writingTel	연락처
statsField	통계(활용)분야·실태
examinObjrange	조사 대상범위
examinObjArea	조사 대상지역
josaUnit	조사단위 및 조사대상규모
applyGroup	적용분류
josaltn	조사항목
pubPeriod	공표주기
pubExtent	공표범위
publicMth	공표방법 및 URL
examinTrgetPd	조사대상기간 및 조사기준시점
dataUserNote	자료이용자 유의사항
mainTermExpl	주요 용어해설
dataCollectMth	자료 수집방법
examinHistory	조사연혁
confmNo	승인번호
confmDt	승인일자
statsEnd	통계종료

■ 예제 소스 결과

항목	설명
조사명	가계동향조사
통계종류	지정통계 / 조사통계
계속여부	계속통계
법적근거	승인번호:10106, 승인일자:1962.12
조사목적	가구의 생활수준실태와 그 변동사항을 파악하기 위해서 가계의 수입과 지출을 조사하여 국민소비수준 변화의 측정 및 분석·소비자 물가지수 편제에 필요한 가중치 모집단 자료 등 각종 경제, 사회정책에 필요한 자료를 제공·공공사업 시행으로 인한 세입자의 주거대책비 산정 및 국민주택 공급대상의 기준설정 자료

■ 예제 소스(JSP)

```

<!--
KOSIS OpenAPI를 이용하여 통계설명자료를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide04/js/dojo.js" ></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide04/js/json.js" ></script>
<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide04/js/ajax.js"></script>
<script type="text/javascript" language="JavaScript">

var mapData;

// window onload 되었을때 실행 함수
dojo.addOnLoad ( function() {

// 통계설명자료를 조회하기위해 함수를 호출한다.
getSubList("ALL", "1962009");
});

/*****
* 통계목록 리스트 조회 함수
* parameter : metaltm - 요청 항목
* statId - 통계조사 Id
*****/
function getSubList(metaltm, statId) {

// ajax 통신을 위한 파라미터를 변수에 담는다.
var paraObj = {
// 임의의 jsp 페이지를 호출함으로써 cross domain 제약을 우회할 수 있다.
url : "http://mgmk.kosis.kr/openapi_dev/devGuidePop.jsp?method=getList&key=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&metaltm=" + metaltm + "&statId=" + statId + "&type=json",

```

```

sync : true,
load : function(resObj, a, b) { mapData = resObj; },
error : function ( resObj, e ) { alert(dojotoJson(resObj)); }
}

sendPost( paraObj );

var nodeInfo="";

// 통계설명자료를 화면에 출력하기 위해 변수에 적재
nodeInfo= "<table cellpadding='0' cellspacing='0' class='TB02' summary=''>";
nodeInfo+=" <thead>"
nodeInfo+=" <tr>"
nodeInfo+=" <th class='fst al_C' scope='col' style='width:20%;>항목</th>"
nodeInfo+=" <th scope='col'>설명</th>"
nodeInfo+=" </tr>"
nodeInfo+=" </thead>"
nodeInfo+=" <tbody>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사명</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].statsNm+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>통계종류</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].statsKind+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>계속여부</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].statsContinue+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>법적근거</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].basisLaw+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사목적</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].writingPurps+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사주기</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].statsPeriod+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사체계</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].writingSystem+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표단위</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].pubExtent+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표주기</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].pubPeriod+"</td>"

```

```

nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>연락처</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].writingTel+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>통계(활용)분야·실태</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].statsField+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사대상범위</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].examinObjrange+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사대상지역</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].examinObjArea+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사단위및조사대상규모</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].josaUnit+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>적용분류</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].applyGroup+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사항목</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].josaltm+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표주기</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].pubPeriod+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표범위</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].pubExtent+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표방법및URL</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].publictMth+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사대상기간및조사기준시점</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].examinTrgetPd+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>자료이용자유의사항</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].dataUserNote+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>주요용어해설</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].mainTermExpl+"</td>"
nodeInfo+=" </tr>"

```

```

nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>자료수집 방법</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].dataCollectMth+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사연혁</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].examinHistory+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>승인번호</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].confmNo+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>승인일자</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].confmDt+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>통계종료</td>"
nodeInfo+=" <td class='al_C'>"+mapData[0].statsEnd+"</td>"
nodeInfo+=" </tr>"
nodeInfo+=" </tbody>"
nodeInfo+="</table>"

// 변수에 적재된 UI를 화면에 출력
var r_node = document.getElementById("content");
r_node.innerHTML =nodeInfo;
}

</script>
</head>
<body>
<div id="content" ></div>
</body>
</html>

// cross domain 제약을 우회하기 위한 jsp (devGuidePop.jsp)
<%@ page contentType="text/html; charset=utf-8" pageEncoding="utf-8" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>

<c:set var="method" ><%out.println(request.getParameter("method") == null ? "" : (request.getParameter("method"))); %></c:set>
<c:set var="key" ><%out.println(request.getParameter("key") == null ? "" : (request.getParameter("key"))); %></c:set>
<c:set var="metaltm" ><%out.println(request.getParameter("metaltm") == null ? "" : (request.getParameter("metaltm")));%></c:set>
<c:set var="statId" ><%out.println(request.getParameter("statId") == null ? "" : (request.getParameter("statId"))); %></c:set>
<c:set var="type" ><%out.println(request.getParameter("type") == null ? "" : (request.getParameter("type"))); %></c:set>

<c:import url="http://mgmk.kosis.kr/openapi_dev/Expt/statisticsExplData.do?method=${method}&apiKey=${key}&metaltm=${metaltm}&statId=${statId}&format=${type}" charEncoding="utf-8"/>

```

■ 예제 소스(R)

```
library(httr)
library(rvest)
library(jsonlite)
library(dplyr)

rm(list = ls()) # 모든변수 초기화

all_items <- data.frame(
  ename = c(
    'statsNm',
    'statsKind',
    'statsContinue',
    'basisLaw',
    'writingPurps',
    'statsPeriod',
    'writingSystem',
    'pubExtent',
    'pubPeriod',
    'writingTel',
    'statsField',
    'examinObjrange',
    'examinObjArea',
    'josaUnit',
    'applyGroup',
    'josaltm',
    'publictMth',
    'examinTrgetPd',
    'dataUserNote',
    'mainTermExpl',
    'dataCollectMth',
    'examinHistory',
    'confmNo',
    'confmDt',
    'statsEnd'
  ),
  hname = c(
    '조사명',
    '통계종류',
    '계속여부',
    '법적근거',
    '조사목적',
    '조사주기',
    '조사체계',
    '공표범위',
    '공표주기',
    '연락처',
    '통계(활용)분야·실태',
    '조사 대상범위',
    '조사 대상지역',
    '조사단위 및 조사대상규모',
    '적용분류',
  )
)
```

```

'조사항목',
'공표방법 및 URL',
'조사대상기간 및 조사기준시점',
'자료이용자 유의사항',
'주요 용어해설',
'자료 수집방법',
'조사연혁',
'승인번호',
'승인일자',
'통계종류'
)
)

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsExplData.do?'

res <- GET(
  url = baseurl,
  query = list(
    method = 'getList',
    format = 'json',
    jsonVD = 'Y',
    apiKey = v_apiKey,
    statId = '1962009',
    metaltm = 'ALL'
  )
)

res %>% content(as = 'text', encoding = 'UTF-8') %>% fromJSON() -> json
df_lists <- data.frame()
json_names <- names(json)

for (i in 1:length(all_items$ename)) {
  if (is.na(match(all_items$ename[i], json_names))==FALSE) {
    v_desc <- select(json, all_items$ename[i])

    df_lists <-
      rbind(df_lists, c(all_items$hname[i], paste(na.omit(v_desc))))
  }
}

colnames(df_lists) <- c("항목", "내용")

View(df_lists)

```

■ 예제 소스(Python)

```

import sys
import json

```

```

from urllib.request import urlopen
from PyQt5.QtWidgets import *
# 한글 폰트 사용을 위해서 세팅
from matplotlib import font_manager, rc
font_path = "C:/Windows/Fonts/malgun.ttf"
font = font_manager.FontProperties(fname=font_path).get_name()
rc('font', family=font)

#url을 통해 json 데이터 가져오기
with urlopen("http://kosis.kr/openapi/statisticsExpData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO  
WRjNjgxN2MzNDgwNmY=&format=json&jsonVD=Y&statId=1962009&metaltn=All") as url:
    json_file = url.read()

py_json = json.loads(json_file.decode('utf-8'))

#변수 지정 및 데이터 저장
data = []

for i, v in enumerate(py_json): #i는 인덱스를 나타냄
    #print(json.loads(json_file[i]))
    value = []
    if 'statsNm' in v:
        value.append("조사명")
        value.append(v['statsNm'])
        data.append(value)

    if 'statsKind' in v:
        value.append("통계종류")
        value.append(v['statsKind'])
        data.append(value)

    if 'statsContinue' in v:
        value.append("계속여부")
        value.append(v['statsContinue'])
        data.append(value)

    if 'basisLaw' in v:
        value.append("법적근거")
        value.append(v['basisLaw'])
        data.append(value)

    if 'writingPurps' in v:
        value.append("조사목적")
        value.append(v['writingPurps'])
        data.append(value)

    if 'statsPeriod' in v:
        value.append("조사주기")
        value.append(v['statsPeriod'])
        data.append(value)

    if 'writingSystem' in v:

```

```

value.append("조사체계")
value.append(v['writingSystem'])
data.append(value)

#if 'pubExtent' in v:
#value.append("공표단위")
#value.append(v['pubExtent'])
#data.append(value)

if 'pubPeriod' in v:
value.append("공표주기")
value.append(v['pubPeriod'])
data.append(value)

if 'writingTel' in v:
value.append("연락처")
value.append(v['writingTel'])
data.append(value)

if 'statsField' in v:
value.append("통계(활용)분야·실태")
value.append(v['statsField'])
data.append(value)

if 'examinObjrange' in v:
value.append("조사대상범위")
value.append(v['examinObjrange'])
data.append(value)

if 'examinObjArea' in v:
value.append("조사대상지역")
value.append(v['examinObjArea'])
data.append(value)

if 'josaUnit' in v:
value.append("조사단위및조사대상규모")
value.append(v['josaUnit'])
data.append(value)

if 'applyGroup' in v:
value.append("적용분류")
value.append(v['applyGroup'])
data.append(value)

if 'josaltm' in v:
value.append("조사항목")
value.append(v['josaltm'])
data.append(value)

if 'pubExtent' in v:
value.append("공표범위")
value.append(v['pubExtent'])
data.append(value)

```



```

if 'publicMth' in v:
    value.append("공표방법및URL")
    value.append(v['publicMth'])
    data.append(value)

if 'examinTrgetPd' in v:
    value.append("조사대상기간및조사기준시점")
    value.append(v['examinTrgetPd'])
    data.append(value)

if 'dataUserNote' in v:
    value.append("자료이용자유의사항")
    value.append(v['dataUserNote'])
    data.append(value)

if 'mainTermExpl' in v:
    value.append("주요용어해설")
    value.append(v['mainTermExpl'])
    data.append(value)

if 'dataCollectMth' in v:
    value.append("자료수집방법")
    value.append(v['dataCollectMth'])
    data.append(value)

if 'examinHistory' in v:
    value.append("조사연혁")
    value.append(v['examinHistory'])
    data.append(value)

if 'confmNo' in v:
    value.append("승인번호")
    value.append(v['confmNo'])
    data.append(value)

if 'confmDt' in v:
    value.append("승인일자")
    value.append(v['confmDt'])
    data.append(value)

if 'statsEnd' in v:
    value.append("통계종료")
    value.append(v['statsEnd'])
    data.append(value)

#데이터개수
count = len(data)

class MyWindow(QMainWindow):
    def __init__(self):
        super().__init__()
        self.setupUI()

```

```

def setupUI(self):
    self.setGeometry(800, 200, 600, 600)
    self.tableWidget = QTableWidgetItem(self)
    self.tableWidget.resize(600, 600)
    self.tableWidget.setRowCount(count)
    self.tableWidget.setColumnCount(2)
    self.tableWidget.setColumnWidth(1, 450)
    self.setTableWidgetData()
    self.tableWidget.resizeRowsToContents()

def setTableWidgetData(self):
    column_headers = ['항목', '설명']
    self.tableWidget.setHorizontalHeaderLabels(column_headers)

    for i in range(count):
        str_data = str(data[i])
        item = str_data.replace("'", "").replace("[", "").replace("]", "")
        total_count = len(item.split(","))

        #항목
        title = item.split(",")[0]
        #설명
        content = ""
        for j in range(1, total_count):
            content += item.split(",")[j]

        self.tableWidget.setItem(0, i*2, QTableWidgetItem(title))
        self.tableWidget.setItem(0, (i*2)+1, QTableWidgetItem(content))

if __name__ == "__main__":
    app = QApplication(sys.argv)
    mywindow = MyWindow()
    mywindow.show()
    app.exec_()

```

2.4.3.2 XML

- 호출 URL : <http://kosis.kr/openApi/StatsExplain.do?method=getList>
- 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
statId * orgId(기관ID) +tblId(통계표ID) 로도 가능	String	통계조사 ID	필수
metaltn	String	요청 항목	필수 전체 - All 조사명-statsNm 통계종류-statsKind 계속여부-statsContinue 법적근거-basisLaw 조사목적-writingPurps 조사주기-statsPeriod 조사체계-writingSystem 공표범위-pubExtent 공표주기-pubPeriod 연락처-writingTel 통계(활용)분야·실태-statsField 조사 대상범위-examinObjrange 조사 대상지역-examinObjArea 조사단위 및 조사대상규모-josaUnit 적용분류-applyGroup 조사항목-josaltm 공표주기-pubPeriod 공표범위-pubExtent 공표방법 및 URL-publicMth 조사대상기간 및 조사기준시점-examinTrgetPd 자료이용자 유의사항 -dataUserNote 주요 용어해설-mainTermExpl 자료 수집방법-dataCollectMth 조사연혁-examinHistory 승인번호-confmNo 승인일자-confmDt 통계종료-statsEnd
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
statsNm	조사명
statsKind	통계종류
statsContinue	계속여부
basisLaw	법적근거
writingPurps	조사목적
statsPeriod	조사주기
writingSystem	조사체계
pubExtent	공표범위
pubPeriod	공표주기
writingTel	연락처
statsField	통계(활용)분야·실태
examinObjrange	조사 대상범위
examinObjArea	조사 대상지역
josaUnit	조사단위 및 조사대상규모
applyGroup	적용분류
josaltn	조사항목
pubPeriod	공표주기
pubExtent	공표범위
publicMth	공표방법 및 URL
examinTrgetPd	조사대상기간 및 조사기준시점
dataUserNote	자료이용자 유의사항
mainTermExpl	주요 용어해설
dataCollectMth	자료 수집방법
examinHistory	조사연혁
confmNo	승인번호
confmDt	승인일자
statsEnd	통계종료

■ 예제 소스 결과

항목	설명
조사명	가계동향조사
통계종류	지정통계 / 조사통계
계속여부	계속통계
법적근거	승인번호:10106, 승인일자:1962.12
조사목적	가구의 생활수준실태와 그 변동사항을 파악하기 위해서 가계의 수입과 지출을 조사하여 국민소비수준 변화의 측정 및 분석·소비자 물가지수 편제에 필요한 가중치 모집단 자료 등 각종 경제, 사회정책에 필요한 자료를 제공·공공사업 시행으로 인한 세입자의 주거대책비 산정 및 국민주택 공급대상의 기준설정 자료

■ 예제 소스(JSP)

```
<!--
KOSIS OpenAPI를 이용하여 통계설명자료를 출력하는 예제입니다.
이 소스는 KOSIS API를 사용하는데 참고가 되도록 제공하는 것으로
사용자의 운영환경에 따라 수정작업이 필요합니다

* 유의사항 : Ajax를 활용하여 개발을 진행하실 때에는 CrossDomain으로 인한
통신문제가 발생 할 수 있습니다.
JSON 방식으로 제공받으실 때에는 개발홈페이지에 임의의 jsp를 생성하여
호출함으로써 CrossDomain에 대한 제약을 우회하실 수 있는 개발소스를 제공합니다.
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page contentType="text/html; charset=utf-8" %>

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>통계청 - 공유서비스</title>

<link type="text/css" rel="stylesheet" media="all" href="https://mgmk.kosis.kr/openapi_dev/ext/style/subCommon.css" />

<script type="text/javascript" language="JavaScript" src="https://mgmk.kosis.kr/openapi_dev/ext/script/jquery-1.6.1.min.js"></script>
<script type="text/javascript" language="JavaScript">

var mapData;

// window onload 되었을때 실행 함수
$(document).ready(function(){

// 통계설명자료를 조회하기위해 함수를 호출한다.
getSubList("ALL", "1962009");
});

/*****
* 통계목록 리스트 조회 함수
* parameter : metaltm - 요청 항목
*****/
```

```

* statId - 통계조사 Id
*****/
function getSubList(metaltn, statId) {

// ajax 통신을 위한 호출 함수
$.ajax({
type: "GET",
url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsExptData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&metaltn=" + metaltn + "&statId=" + statId + "&format=xml&version=v2_1",
data: "",
async : true,
dataType: "xml",
success:function(object)
{

// ajax 통신이 성공하였을 때 통계설명자료를 화면에 출력하기 위한 함수
var data = object.documentElement.childNodes[1].childNodes;

var nodeInfo="";

nodeInfo= "<table cellpadding='0' cellspacing='0' class='TB02' summary=''>";
nodeInfo+=" <thead>"
nodeInfo+=" <tr>"
nodeInfo+=" <th class='fst al_C' scope='col' style='width:20%;>항목</th>"
nodeInfo+=" <th scope='col'>설명</th>"
nodeInfo+=" </tr>"
nodeInfo+=" </thead>"
nodeInfo+=" <tbody>"
for (var i = 0; i < data.length; i++) {

if (data[i].tagName == "statsNm") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사명</td>"
nodeInfo+=" <td class='al_C'>" + data[i].childNodes[0].data + "</td>"
nodeInfo+=" </tr>"
}

if (data[i].tagName == "statsKind") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>통계종류</td>"
nodeInfo+=" <td class='al_C'>" + data[i].childNodes[0].data + "</td>"
nodeInfo+=" </tr>"
}

if (data[i].tagName == "statsContinue") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>계속여부</td>"
nodeInfo+=" <td class='al_C'>" + data[i].childNodes[0].data + "</td>"
nodeInfo+=" </tr>"
}

if (data[i].tagName == "basisLaw") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>법적근거</td>"
nodeInfo+=" <td class='al_C'>" + data[i].childNodes[0].data + "</td>"
}
}
}

```

```

nodeInfo+=" </tr>"
}
if (data[i].tagName == "writingPurps") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사목적</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "statsPeriod") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사주기</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "writingSystem") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사체계</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "pubExtent") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표단위</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "pubPeriod") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표주기</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "writingTel") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>연락처</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "statsField") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>통계(활용)분야·실태</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "examinObjrange") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사대상범위</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "examinObjArea") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사대상지역</td>"

```

```

nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "josaUnit") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사단위및조사대상규모</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "applyGroup") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>적용분류</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "josaltm") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사항목</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "pubPeriod") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표주기</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "pubExtent") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표범위</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "publicMth") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>공표방법및URL</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "examinTrgetPd") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사대상기간및조사기준시점</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "dataUserNote") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>자료이용자유의사항</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "mainTermExp") {
nodeInfo+=" <tr>"

```



```

nodeInfo+=" <td class='fst al_C' scope='row'>주요용어해설</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "dataCollectMth") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>자료수집방법</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "examinHistory") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>조사연혁</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "confmNo") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>승인번호</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "confmDt") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>승인일자</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
if (data[i].tagName == "statsEnd") {
nodeInfo+=" <tr>"
nodeInfo+=" <td class='fst al_C' scope='row'>통계종료</td>"
nodeInfo+=" <td class='al_C'>"+data[i].childNodes[0].data+"</td>"
nodeInfo+=" </tr>"
}
}

nodeInfo+=" </tbody>"
nodeInfo+="</table>"

var r_node = document.getElementById("content");
r_node.innerHTML =nodeInfo;
},
error: function(xhr,status,error){
alert("error = " + error);
}
});
}
</script>
</head>
<body>
<div id="content" ></div>
</body>
</html>

```

■ 예제 소스(R)

```
library(rvest) # api의 자료를 xml, html로 받은 경우
library(XML)   # API에서 XML데이터 수집
library(RCurl)
library(tidyverse)
library(xml2)

all_items <- data.frame(
  ename = c(
    'statsNm',
    'statsKind',
    'statsContinue',
    'basisLaw',
    'writingPurps',
    'statsPeriod',
    'writingSystem',
    'pubExtent',
    'pubPeriod',
    'writingTel',
    'statsField',
    'examinObjrange',
    'examinObjArea',
    'josaUnit',
    'applyGroup',
    'josaltm',
    'publictMth',
    'examinTrgetPd',
    'dataUserNote',
    'mainTermExpl',
    'dataCollectMth',
    'examinHistory',
    'confmNo',
    'confmDt',
    'statsEnd'
  ),
  hname = c(
    '조사명',
    '통계종류',
    '계속여부',
    '법적근거',
    '조사목적',
    '조사주기',
    '조사체계',
    '공표범위',
    '공표주기',
    '연락처',
    '통계(활용)분야·실태',
    '조사 대상범위',
    '조사 대상지역',
    '조사단위 및 조사대상규모',
    '적용분류',
    '조사항목',
  )
)
```

```

'공표방법 및 URL',
'조사대상기간 및 조사기준시점',
'자료이용자 유의사항',
'주요 용어해설',
'자료 수집방법',
'조사연혁',
'승인번호',
'승인일자',
'통계종료'
)
)

# 환경변수에 사용자 key 정의 (환경변수 편집 usethis::edit_r_environ() )
# KOSIS_TOKEN = ZjZOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseUrl <- 'https://kosis.kr/openapi/statisticsExplData.do?'

requestURL <- paste0(
  baseUrl,
  "method=getList",
  "&format=xml",
  "&apiKey=" ,
  v_apiKey %>% I(),
  "&statId=" ,
  '1962009',
  "&metaltm=",
  'ALL'
)

raw_xml <- read_xml(requestURL)

xml_nodeSet <- xml_find_all(raw_xml, "//Structures")

df_lists <- data.frame()

for (i in 1:xml_length(xml_nodeSet[1])) {
  j <- xml_length(xml_nodeSet[1]) - i + 1

  v_desc <- xml_child(xml_nodeSet[[1]], j) %>% xml_text()
  v_name <- xml_child(xml_nodeSet[[1]], j) %>% xml_name()

  all_items %>% filter(ename == v_name) %>% select(hname) -> i_hname

  df_lists <- rbind(df_lists, c(item = i_hname, desc = v_desc))
}

colnames(df_lists) <- c("항목", "내용")
View(df_lists)

```

■ 예제 소스(Python)

```
import sys
import requests
from bs4 import BeautifulSoup
from PyQt5.QtWidgets import *

open_url = 'http://kosis.kr/openapi/statisticsExpData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=xml&statId=1962009&metaltn=All&version=v2_1'

res = requests.get(open_url)
soup = BeautifulSoup(res.content, 'html.parser')

#변수 지정 및 데이터 저장
data = []
count = 0

if soup.find('statsnm') is not None:
    value = []
    value.append("조사명")
    value.append(soup.find('statsnm').text)
    data.append(value)
    count += 1

if soup.find('statskind') is not None:
    value = []
    value.append("통계종류")
    value.append(soup.find('statskind').text)
    data.append(value)
    count += 1

if soup.find('statscontinue') is not None:
    value = []
    value.append("계속여부")
    value.append(soup.find('statscontinue').text)
    data.append(value)
    count += 1

if soup.find('basislaw') is not None:
    value = []
    value.append("법적근거")
    value.append(soup.find('basislaw').text)
    data.append(value)
    count += 1

if soup.find('writingpurps') is not None:
    value = []
    value.append("조사목적")
    value.append(soup.find('writingpurps').text)
    data.append(value)
    count += 1
```

```

if soup.find('statsperiod') is not None:
    value = []
    value.append("조사주기")
    value.append(soup.find('statsperiod').text)
    data.append(value)
    count += 1

if soup.find('writingsystem') is not None:
    value = []
    value.append("조사체계")
    value.append(soup.find('writingsystem').text)
    data.append(value)
    count += 1

#if soup.find('pubextent') is not None:
#    #value = []
#    #value.append("공표단위")
#    #value.append(soup.find('pubextent').text)
#    #data.append(value)
#    #count += 1

if soup.find('pubperiod') is not None:
    value = []
    value.append("공표주기")
    value.append(soup.find('pubperiod').text)
    data.append(value)
    count += 1

if soup.find('writingtel') is not None:
    value = []
    value.append("연락처")
    value.append(soup.find('writingtel').text)
    data.append(value)
    count += 1

if soup.find('statsfield') is not None:
    value = []
    value.append("통계(활용)분야·실태")
    value.append(soup.find('statsfield').text)
    data.append(value)
    count += 1

if soup.find('examinobjrange') is not None:
    value = []
    value.append("조사대상범위")
    value.append(soup.find('examinobjrange').text)
    data.append(value)
    count += 1

if soup.find('examinobjArea') is not None:
    value = []
    value.append("조사대상지역")
    value.append(soup.find('examinobjArea').text)

```

```

data.append(value)
count += 1

if soup.find('josaunit') is not None:
    value = []
    value.append("조사단위및조사대상규모")
    value.append(soup.find('josaunit').text)
    data.append(value)
    count += 1

if soup.find('applygroup') is not None:
    value = []
    value.append("적용분류")
    value.append(soup.find('applygroup').text)
    data.append(value)
    count += 1

if soup.find('josaitm') is not None:
    value = []
    value.append("조사항목")
    value.append(soup.find('josaitm').text)
    data.append(value)
    count += 1

if soup.find('pubperiod') is not None:
    value = []
    value.append("공표주기")
    value.append(soup.find('pubperiod').text)
    data.append(value)
    count += 1

if soup.find('pubextent') is not None:
    value = []
    value.append("공표범위")
    value.append(soup.find('pubextent').text)
    data.append(value)
    count += 1

if soup.find('publictmth') is not None:
    value = []
    value.append("공표방법및URL")
    value.append(soup.find('publictmth').text)
    data.append(value)
    count += 1

if soup.find('examintrgetpd') is not None:
    value = []
    value.append("조사대상기간및조사기준시점")
    value.append(soup.find('examintrgetpd').text)
    data.append(value)
    count += 1

if soup.find('datausernote') is not None:

```

```

value = []
value.append("자료이용자유의사항")
value.append(soup.find('datausernote').text)
data.append(value)
count += 1

if soup.find('maintermexpl') is not None:
    value = []
    value.append("주요용어해설")
    value.append(soup.find('maintermexpl').text)
    data.append(value)
    count += 1

if soup.find('datacollectmth') is not None:
    value = []
    value.append("자료수집방법")
    value.append(soup.find('datacollectmth').text)
    data.append(value)
    count += 1

if soup.find('examinhistory') is not None:
    value = []
    value.append("조사연혁")
    value.append(soup.find('examinhistory').text)
    data.append(value)
    count += 1

if soup.find('confmno') is not None:
    value = []
    value.append("승인번호")
    value.append(soup.find('confmno').text)
    data.append(value)
    count += 1

if soup.find('confmdt') is not None:
    value = []
    value.append("승인일자")
    value.append(soup.find('confmdt').text)
    data.append(value)
    count += 1

if soup.find('statsend') is not None:
    value = []
    value.append("통계종료")
    value.append(soup.find('statsend').text)
    data.append(value)
    count += 1

class MyWindow(QMainWindow):

    def __init__(self):
        super().__init__()

```

```

self.setupUI()

def setupUI(self):
    self.setGeometry(800, 200, 600, 600)
    self.tableWidget = QTableWidgetItem(self)
    self.tableWidget.resize(600, 600)
    self.tableWidget.setRowCount(count)
    self.tableWidget.setColumnCount(2)
    self.tableWidget.setColumnWidth(1, 450)
    self.setTableWidgetData()
    self.tableWidget.resizeRowsToContents()
    #window title 설정
    self.setWindowTitle(soup.find('structures').find('statsnm').text)

def setTableWidgetData(self):
    column_headers = ['항목', '설명']
    self.tableWidget.setHorizontalHeaderLabels(column_headers)

    for i in range(count):
        str_data = str(data[i])
        item = str_data.replace("'", "").replace("[", "").replace("]", "")
        total_count = len(item.split(","))
        title = item.split(",")[0]
        content = ""
        for j in range(1, total_count):
            content += item.split(",")[j]

        self.tableWidget.setItem(0, i * 2, QTableWidgetItem(title))
        self.tableWidget.setItem(0, (i * 2) + 1, QTableWidgetItem(content))

if __name__ == "__main__":
    app = QApplication(sys.argv)
    mywindow = MyWindow()
    mywindow.show()
    app.exec_()

```


2.5 메타자료

- 통계자료에 대한 메타자료 제공을 위한 OpenAPI입니다.

2.5.1 특징

- 자료 제공형태 : XML, JSON

2.5.2 서비스 활용

2.5.2.1 URL생성

- 개발가이드 > 통계표 설명> URL생성

‘작성기관’, ‘통계조사명’, ‘통계표명’ 등을 입력하여 사용하고자 하는 자료를 조회한 뒤 조회결과에서 생성할 통계표 ①선택하고, ②통계표 설명 분류선택, ③세부정보 입력하여 ④URL생성 버튼을 누르면 결과 값을 제공받을 수 있는 URL이 생성됩니다.
⑤통계표조회 버튼을 클릭하면 해당 자료의 통계표를 볼 수 있습니다.

메타자료

Home > 개발 가이드 > 메타자료

통계자료 이용시 참고할 수 있도록 통계표설명자료를 제공합니다. 출력포맷으로 JSON과 XML을 제공합니다.

JSON

XML

URL 생성

통계표 선택

작성기관

대전광역시

통계조사명

주민등록인구통계

통계표명

통계표ID

검색

기관명	통계표명	통계표ID	통계조사명	수집기간	선택	통계표조회
대전광역시 (206)	구별 통별 세대 및 인구	DT_POPUL0001	주민등록인구통계	분기,년 2009~2022 2/4	<input checked="" type="checkbox"/>	
대전광역시 (206)	구별 연령별(5세계급) 외국인 현황	DT_POPUL0002	주민등록인구통계	분기,년 2009~2022 2/4	<input type="checkbox"/>	
대전광역시 (206)	구별 국적별 외국인 현황	DT_POPUL0003	주민등록인구통계	분기,년 2010~2022 2/4	<input type="checkbox"/>	
대전광역시 (206)	구별 연령별 인구(외국인 제외)	DT_POPUL01	주민등록인구통계	분기,년 2009~2022 2/4	<input type="checkbox"/>	

통계표 설명 분류선택

통계표명

기관명

수집정보

분류/항목

주석

단위

출처

가중치

자료경신일

세부정보입력

출력포맷

xml

기관코드

206

통계표ID

DT_POPUL0001

URL복사

URL생성

생성된 URL

https://kosis.kr/openapi/statisticsData.do?method=getMeta&apiKey=WWVhZDgyNDVhZTJhZTk2MmE0OGVhYjY0YzgzMzowY20=&format=xml&type=TBL&orgId=206&tblId=DT_POPUL0001

2.5.2 활용방법

2.5.2.1 JSON (통계표 명칭)

- 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=TBL>
- 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표 ID	필수
format	String	결과유형(JSON, SDMX)	필수

- 출력 변수

항목명(영문)	항목설명
TBL_NM	통계표 국문명
TBL_NM_ENG	통계표 영문명

2.5.2.2 JSON (기관 명칭)

- 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=ORG>
- 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
format	String	결과유형(JSON, SDMX)	필수

- 출력 변수

항목명(영문)	항목설명
ORG_NM	기관 국문명
TORG_NM_ENG	기관 영문명

2.5.2.3 JSON (수록정보)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=PRD>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
format	String	결과유형(JSON, SDMX)	필수
detail	String	전체시점 정보 제공	선택

■ 출력 변수

항목명(영문)	항목설명
PRD_SE	수록주기
PRD_DE	수록시점

2.5.2.4 JSON (분류/항목)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=ITM>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 코드	필수
tblId	String	통계표ID	필수
objId	String	분류코드	선택
itmId	String	자료코드	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
OBJ_ID	분류 ID
OBJ_NM	분류 국문명
OBJ_NM_ENG	분류 영문명
ITM_ID	자료코드 ID
ITM_NM	자료코드 국문명
ITM_NM_ENG	자료코드 영문명
UP_ITM_ID	상위 자료코드
OBJ_ID_SN	분류값 순번
UNIT_ID	단위ID
UNIT_NM	단위 국문명
UNIT_ENG_NM	단위 영문명

2.5.2.5 JSON (주석)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=CMMT>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
CMMT_NM	주석유형
CMMT_DC	주석
OBJ_ID	분류 ID
OBJ_NM	분류 명
ITM_ID	자료코드 ID
ITM_NM	자료코드 명

2.5.2.6 JSON (단위)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=UNIT>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관코드	필수
tblId	String	통계표ID	필수
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
UNIT_NM	단위 국문명
UNIT_NM_ENG	단위 영문명

2.5.2.7 JSON (출처)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=SOURCE>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
JOSA_NM	조사명
DEPT_NM	통계표 담당부서
DEPT_PHONE	단위 담당부서 전화번호
STAT_ID	통계조사ID

2.5.2.8 JSON (가중치)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=WGT>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
분류코드1~분류코드8	String	분류코드1~분류코드8	선택
ITEM	String	항목	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
C1 ~ C8	분류값 ID1 ~ 분류값 ID8
C1_NM ~ C8_NM	분류값 명1 ~ 분류값 명8
ITM_ID	항목 ID
ITM_NM	항목명
WGT_CO	가중치

2.5.2.9 JSON (자료갱신일)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=NCD>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
prdSe	String	수록주기	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
ORG_NM	기관명
TBL_NM	통계표명
PRD_SE	수록주기
PRD_DE	수록시점
SEND_DE	자료갱신일

2.5.2.10 XML (통계표 명칭)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=TBL>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
tblNm	통계표 국문명
tblNmEng	통계표 영문명

2.5.2.11 XML (기관 명칭)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=ORG>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
orgNm	기관 국문명
orgNmEng	기관 영문명

2.5.2.12 XML (수록정보)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=PRD>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
format	String	결과유형(JSON, SDMX)	필수
detail	String	전체시점 정보 제공	선택

■ 출력 변수

항목명(영문)	항목설명
prdSe	수록주기
PrdDe	수록시점

2.5.2.13 XML (분류/항목)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=ITM>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
objId	String	분류코드	선택
itmId	String	자료코드	선택
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
objId	분류 ID
objNm	분류 국문명
objNmEng	분류 영문명
itmId	자료코드 ID
itmNm	자료코드 국문명
itmNmEng	자료코드 영문명
upltnId	상위 자료코드
objIdSn	분류값 순번
unitId	단위ID
unitNm	단위 국문명
unitEngNm	단위 영문명

2.5.2.14 XML (주석)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=CMMT>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
CMMT_NM	주석유형
CMMT_DC	주석
OBJ_ID	분류 ID
OBJ_NM	분류 명
ITM_ID	자료코드 ID
ITM_NM	자료코드 명

2.5.2.15 XML (단위)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=UNIT>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
unitId	String	단위코드	필수
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
unitNm	단위 국문명
unitNmEng	단위 영문명

2.5.2.16 XML (출처)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=SOURCE>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
josaNm	조사명
deptNm	통계표 담당부서
deptPhone	통계표 담당부서 전화번호
statId	통계조사 ID

2.5.2.17 XML (가중치)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=WGT>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
분류코드1~분류코드8	String	분류코드1~분류코드8	선택
ITEM	String	항목	선택
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명
C1 ~ C8	분류값 ID1 ~ 분류값 ID8
C1_NM ~ C8_NM	분류값 명1 ~ 분류값 명8
ITM_ID	항목 ID
ITM_NM	항목명
WGT_CO	가중치

2.5.2.18 XML (자료갱신일)

■ 호출 URL : <http://kosis.kr/openapi/statisticsData.do?method=getMeta&type=NCD>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표ID	필수
prdSe	String	수록주기	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
orgNm	기관명
tblNm	통계표명
prdSe	수록주기
prdDe	수록시점
sendDe	자료갱신일

2.6 KOSIS통합검색

- 국가통계포털(www.kosis.kr)의 통합검색결과 제공을 위한 OpenAPI입니다.

2.6.1 특징

- 자료 제공형태 : JSON

2.6.2 서비스 활용

2.6.2.1 URL생성

- 개발가이드 > KOSIS 통합검색 > URL생성

‘검색어’, ‘정렬’, ‘페이지번호’ 등을 입력한 뒤 ‘URL복사’, ‘결과값보기’ 중 원하는 서비스에 해당하는 버튼을 누르면 결과값을 제공받을 수 있습니다.

KOSIS통합검색
Home > 개발 가이드 > KOSIS통합검색
인쇄

국가통계포털(www.kosis.kr)의 통합검색결과를 제공합니다. 출력포맷으로 JSON을 제공합니다.

JSON
URL 생성

URL생성 조건 설정

검색어 입력	인구
정렬	정확도순
기관검색(선택사항)	
페이지 번호	1
데이터 출력 개수	20
결과유형	JSON

결과값보기
URL복사

생성된 URL

```
https://kosis.kr/openapi/statisticsSearch.do?method=getList&apiKey=WVhZDgyNDYkZTJhZTk2MmE0OGVhYjY0YzgzMzowY2Q=&format=json&jsonV=D=Y&searchNm=인구&orgId=&startCount=1&resultCount=20&sort=rank
```

결과값보기

```
[{"ORG_ID":"101","ORG_NM":"통계청","TBL_ID":"DT_2KAA202","TBL_NM":"무양비 및 노령화지수","STAT_ID":"9805001","STAT_NM":"국제통계연
람","VW_CD":"MT_RTI","MT_ATITLE":"주제별 통계 > 영토/인구","FULL_PATH_ID":"101_003 > 101_003_A","CONTENTS":"국가별 인구(85세 이상)
무양비(노년)
무양비(0~14세)
노령화지수
인구
인구(15~64세)
총무양비(유년)
아프리카나탈
아제르바이잔
바레인
카자흐스탄
북한
라오스
레바논
모란
대한
튀르키예
영국
루만
키
프랑스
조지아
인도
싱가포르
중국
일본
우즈베키스탄
브루나이
캄보디아
미얀마
네팔
스리랑카
아르메니아
인도네시아
몰디브
사우디아라비아
한국
북마케도니아
이란
이스라엘
마카오
몰리브
라치카스
베트남
중국
가자지구
투르크메니스탄
파키스탄
시리아
아랍에미리트
대한민국
...","STAT_PRODE":"1950","END_PRODE":"2100","ITEMOS":"총정: 통계청, UN 자료: 중위(medium) 자료집 통계청 「장래인구추계」 2019. 3. UN
「https://population.un.org/wpp/ World Population Prospects 2019」 2019. 8. ② 무양비(Dependency Ratio) 총인구 중에서 생산가능연령(15~64세) 인구에 대한 비
생산 연령(0~14세, 65세 이상) 인구의 백분비로서 생산가능연령 인구가 무양해야 하는 경제적 부담을 나타내는 지표 ③ 노령화지수(Index of Ageing) 유소년(
0~14세) 인구에 대한 노령(65세 이상) 인구의 비","REQ_TBL_SE":"N","TBL_VIEWS_URL":"https://kosis.kr/statisticsList/statisticsListIndex.do?
menuId=M_02_01&vwcd=MT_RTI&parTabId=M_01_01&parentId=101_003_1:101_003_A:2","LINK_URL":"http://kosis.kr/statHtml/statHtml.do?
orgId=101&tId=DT_2KAA202","STAT_DE_CNT":"66231","QUERY":"인구","ORG_ID":"101","ORG_NM":"통계청","TBL_ID":"DT_2KAA202_OECD","TBL_NM":"무양비 및 노령화지
수(OECD)","STAT_ID":"9805001","STAT_NM":"국제통계연람","VW_CD":"MT_RTI","MT_ATITLE":"주제별 통계 > OECD 가입국 > 영토/인
구"
```

2.6.2 활용방법

2.6.2.1 JSON

■ 호출 URL : <http://kosis.kr/openapi/statisticsSearch.do?method=getList>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
searchNm	String	검색명	필수
sort	String	정렬 비고 : 정확도 RANK, 최신순 DATE ※ 호출 파라미터에 sort 없을 경우에는 자동으로 RANK 로 정렬	선택
startCount	String	페이지 번호	선택
resultCount	String	데이터 출력 개수 비고 : resultCount=20, startCount=1 : 1~20번 결과 리턴 resultCount=20, startCount=2 : 21~40번 결과 리턴	선택
format	String	결과유형(json)	필수

■ 출력 변수

항목명(영문)	항목설명
ORG_ID	기관코드
ORG_NM	기관명
TBL_ID	통계표ID
TBL_NM	통계표명
STAT_ID	조사코드
STAT_NM	조사명
VW_CD	KOSIS 목록구분
MT_ATITLE	KOSIS 통계표 위치
FULL_PATH_ID	통계표 위치
CONTENTS	통계표 주요내용
STRT_PRD_DE	수록기간 시작일
END_PRD_DE	수록기간 종료일
ITEM03	통계표 주석
REC_TBL_SE	추천통계표 여부
TBL_VIEW_URL	통계표 이동URL (KOSIS 목록으로 이동)
LINK_URL	통계표 이동URL (KOSIS 통계표로 이동)
STAT_DB_CNT	검색결과 건수
QUERY	검색어명

2.7 통계주요지표

- 지표 Open API 서비스를 이용하기 위한 JSON, XML 기반의 데이터형식, 요청변수, 반환되는 코드값을 제공합니다.

2.7.1 특징

- 자료 제공형태 : XML, JSON

2.7.2 서비스 활용

2.7.2.1 URL생성

- 개발가이드 > 통계주요지표 > URL생성
'서비스선택', '상세기능선택'을 선택한 뒤 'URL생성', 'URL복사' 중 원하는 서비스에 해당하는 버튼을 누르면 결과값을 제공받을 수 있습니다.

통계주요지표
Home > 개발 가이드 > 통계주요지표
인쇄

지표 Open API 서비스를 이용하기 위한 JSON, XML 기반의 데이터형식, 요청변수, 반환되는 코드값을 제공합니다.

JSON
XML
URL 생성

※ 지표ID 및 지표명에 관한 정보는 개발가이드-통계주요지표의 코드정보를 참조해주세요.

서비스선택

☒ 지표고유번호별 설명자료조회
☐ 지표명별 설명자료조회
☐ 목록별 지표 조회서비스
☐ 지표정보 조회서비스

상세기능선택

☒ 고유번호별 개별조회
☐ 고유번호별 산정방법 및 출처조회
☐ 고유번호별 전체 설명자료조회

세부정보입력

출력포맷: xml
지표ID: 166

URL 생성
URL 복사

생성된 URL

https://kosis.kr/openapi/pkNumberService.do?method=getList&apiKey=WWVhZDgyNDVhZTk2MmE0OGVhYjY0YzgzMzowY2Q=&format=xml&jiyoId=166&service=1&serviceDetail=pkNotion

2.7.2 활용방법

2.7.2.1 JSON, XML (지표 고유번호별 설명자료조회)

■ 호출 URL :

<http://kosis.kr/openapi/pkNumberService.do?method=getList&service=1&serviceDetail=pkAll>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
jipyold	String	지표 ID	필수
pageNo	String	페이지 번호	선택
numOfRows	String	데이터 출력 개수 비고 : numOfRows=20, pageNo=1 : 1~20번 결과 리턴 numOfRows=20, pageNo=2 : 21~40번 결과 리턴 ※ 호출 파라미터에 pageNo, numOfRows 없을 경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
statJipyold	지표ID
statJipyoNm	지표명
jipyoExplan	설명자료 제목
jipyoExplan1	개념

2.7.2.2 JSON, XML (지표명별 설명자료조회)

■ 호출 URL :

<http://kosis.kr/openapi/indExpService.do?method=getList&service=2&serviceDetail=indAll>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
jipyoNm	String	지표명	필수
pageNo	String	페이지 번호	선택
numOfRows	String	데이터 출력 개수 비고 : numOfRows=20, pageNo=1 : 1~20번 결과 리턴 numOfRows=20, pageNo=2 : 21~40번 결과 리턴 ※ 호출 파라미터에 pageNo, numOfRows 없을 경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
statJipyoId	지표ID
statJipyoNm	지표명
jipyoExplan	설명자료 제목
jipyoExplan1	개념

2.7.2.3 JSON, XML (목록별 지표조회)

■ 호출 URL : <http://kosis.kr/openapi/indilistService.do?method=getList&service=3>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
listId	String	목록ID	필수
pageNo	String	페이지 번호	선택
numOfRows	String	데이터 출력 개수 비고 : numOfRows=20, pageNo=1 : 1~20번 결과 리턴 numOfRows=20, pageNo=2 : 21~40번 결과 리턴 ※ 호출 파라미터에 pageNo, numOfRows 없을 경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
listId	세부목록ID
listNm	세부분명
statJipyold	지표ID
statJipyoNm	지표명
unit	단위
areaTypeName	지역구분명
prdSeName	수록주기명
strtPrdDe	수록시작시점
endPrdDe	수록종료시점
rn	수록시점개수
listSn1	목록순서1
listSn2	목록순서2
prdDe	시점
repJipyold	대표지표ID
repJipyoNm	대표지표명
repJipyoUrl	대표지표URL
explainUrl	지표설명URL

2.7.2.4 JSON, XML (지표명별 목록조회)

■ 호출 URL :

<http://kosis.kr/openapi/indListSearchRequest.do?method=getList&service=4&serviceDetail=indList>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
jipyoNm	String	지표명	필수
pageNo	String	페이지 번호	선택
numOfRows	String	데이터 출력 개수 비고 : numOfRows=20, pageNo=1 : 1~20번 결과 리턴 numOfRows=20, pageNo=2 : 21~40번 결과 리턴 ※ 호출 파라미터에 pageNo, numOfRows 없을 경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
statJipyoId	지표ID
statJipyoNm	지표명
unit	단위
areaTypeName	지역구분명
prdSeName	수록주기명
strtPrdDe	수록시작시점
endPrdDe	수록종료시점
rn	수록시점개수
prdDe	종료시점+주기명

2.7.2.5 JSON,XML (고유번호별 목록조회)

■ 호출 URL :

<http://kosis.kr/openapi/indListSearchRequest.do?method=getList&service=4&serviceDetail=indList>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
jipyold	String	지표ID	필수
pageNo	String	페이지 번호	선택
numOfRows	String	데이터 출력 개수 비고 : numOfRows=20, pageNo=1 : 1~20번 결과 리턴 numOfRows=20, pageNo=2 : 21~40번 결과 리턴 ※ 호출 파라미터에 pageNo, numOfRows 없을 경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
statJipyold	지표ID
statJipyoNm	지표명
unit	단위
areaTypeName	지역구분명
prdSeName	수록주기명
strtPrdDe	수록시작시점
endPrdDe	수록종료시점
rn	수록시점개수
prdDe	종료시점+주기명

2.7.2.6 JSON,XML (고유번호별 지표 상세조회)

■ 호출 URL : <http://kosis.kr/openapi/indIdDetailSearchRequest.do?method=getList&service=4&serviceDetail=indIdDetail>

■ 입력 변수

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
jipyold		String	지표ID	필수
시점기준	startPrdDe	String	조회 시작 시점	선택
	endPrdDe	String	조회 종료 시점	
최신자료기준	rn	String	조회 기준 시점	선택
	srvRn	String	조회 시점 개수	
pageNo		String	페이지 번호	선택
numOfRows		String	데이터 출력 개수 비고 : numOfRows=20, pageNo=1 : 1~20번 결과 리턴 numOfRows=20, pageNo=2 : 21~40번 결과 리턴 ※ 호출 파라미터에 pageNo, numOfRows 없을 경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format		String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
statJipyold	지표ID
statJipyoNm	지표명
prdSe	수록주기
prdDe	시점
itmNm	항목
val	통계수치

2.7.2.7 JSON,XML (수록주기별 목록조회)

■ 호출 URL :

<http://kosis.kr/openapi/prListSearchRequest.do?method=getList&service=4&serviceDetail=prList>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
prdSe	String	수록주기	필수
pageNo	String	페이지 번호	선택
numOfRows	String	데이터 출력 개수 비고 : numOfRows=20, pageNo=1 : 1~20번 결과 리턴 numOfRows=20, pageNo=2 : 21~40번 결과 리턴 ※ 호출 파라미터에 pageNo, numOfRows 없을 경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
statJipyold	지표ID
statJipyoNm	지표명
unit	단위
areaTypeName	지역구분명
prdSeName	수록주기명
strtPrdDe	수록시작시점
endPrdDe	수록종료시점
rn	수록시점개수
prdDe	종료시점+주기명

2.7.2.8 JSON,XML (지표명별 상세조회)

■ 호출 URL : [http://kosis.kr/openapi/indDetailSearchRequest.do?method=getList&service=4
&serviceDetail=indDetail](http://kosis.kr/openapi/indDetailSearchRequest.do?method=getList&service=4&serviceDetail=indDetail)

■ 입력 변수

항목명(영문)		변수타입	항목설명	비고
apiKey		String	발급된 인증키	필수
jipyoNm		String	지표명	필수
시점기준	startPrdDe	String	조회 시작 시점	선택
	endPrdDe	String	조회 종료 시점	
최신자료기준	rn	String	조회 기준 시점	선택
	srvRn	String	조회 시점 개수	
pageNo		String	페이지 번호	선택
numOfRows		String	데이터 출력 개수 비고 : numOfRows=20, pageNo=1 : 1~20번 결과 리턴 numOfRows=20, pageNo=2 : 21~40번 결과 리턴 ※ 호출 파라미터에 pageNo, numOfRows 없을 경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format		String	결과유형(json, xml)	필수

■ 출력 변수

항목명(영문)	항목설명
statJipyold	지표ID
statJipyoNm	지표명
prdSe	수록주기
prdDe	시점
itmNm	항목
val	통계수치

<참고> SDMX

KOSIS 공유서비스 자료 제공형태 중에서 SDMX는 XML의 일종으로서 통계에 특화된 XML로 보시면 됩니다.

SDMX는 Statistical Data and Metadata eXchange의 약어로 통계작성기구(기관)간의 다양한 형태의 통계 자료를 XML 기반으로 제공하여 교환과 공유를 효율적으로 지원합니다.

SDMX 표준은 국제결제은행(BIS), 유럽중앙은행, 유럽통계처(Eurostat), 국제통화기금(IMF), 경제협력개발기구(OECD), UN 통계국 및 세계은행의 국제기구들로부터 후원을 받고 있습니다.

현재 Version 2.1은 2013년 1월에 ISO(국제표준화기구) 17369로 국제표준 인증을 받았으며 각 국가의 통계청과 여러 국제기구에 이르기까지 사용범위가 점차 확대되고 있습니다.

SDMX 표준 및 지침, 개발도구(software), 새로운 소식 등은 현재 sdmx.org 웹사이트(링크)를 통해 제공되고 있습니다.

※ SDMX 파일에 대한 설명(SDMX Version 2.1)

구 분	설 명
DSD	Data Structure Definition의 약어로 통계자료에 대한 의미와 구조를 정의 통계표를 예를 들면 통계표에 대한 설명과 통계표의 형태를 파악할 수 있는 통 계표의 구성정보와 분류, 분류값, 단위, 항목에 대한 상세정보(코드 및 명칭)를 담 고 있음
DATA	DATA 파일은 DSD에서 정의한 구성정보에 주기, 시점에 따른 수치정보를 정의 SDMX version 2.1에서는 Generic, StructureSpecific 두 가지 포맷으로 나뉘어짐 <ul style="list-style-type: none"> • Generic : 데이터를 담는 XML 구성요소가 구조를 정의하는 메시지와 독립적인 형태로 이루어져 있으며 통계구성정보 및 수치정보가 각 Element(요소, 항 목)로 구성되어 있어 StructureSpecific에 비해 파일 용량이 큼 • StructureSpecific : 데이터를 담는 XML 구성요소는 구조를 정의하는 메시지에 의존적인 형태로 이루어져 있으며 통계구성정보 및 수치정보는 하나의 Element(요소, 항목)에 Attribute(속성)로 나열되어 있어 Generic에 비해 파일 용량이 작음