

---

# **KOSIS 공유서비스(OpenAPI)**

## **개발가이드**

---



# 목 차

<b>1. KOSIS 공유서비스(OpenAPI) 개요</b>	<b>5</b>
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
<b>2. KOSIS 공유서비스(OpenAPI) 주요기능 및 활용방법</b>	<b>17</b>
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
<b>2.3 대용량 통계자료 .....</b>	<b>102</b>
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
<b>2.4 통계설명 .....</b>	<b>110</b>
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
<b>2.5 메타자료 .....</b>	<b>137</b>
2.5.1 특징 .....	137
2.5.2 서비스활용 .....	137
2.5.2.1 URL생성 .....	137
2.5.3 활용방법 .....	138
2.5.3.1 JSON(통계표 명칭) .....	138
2.5.3.2 JSON(기관 명칭) .....	138
2.5.3.3 JSON(수록정보) .....	139
2.5.3.4 JSON(분류/항목) .....	139
2.5.3.5 JSON(주석) .....	140
2.5.3.6 JSON(단위) .....	140
2.5.3.7 JSON(출처) .....	141
2.5.3.8 JSON(가중치) .....	141
2.5.3.9 JSON(자료갱신일) .....	142
2.5.3.10 XML(통계표 명칭) .....	142
2.5.3.11 XML(기관 명칭) .....	143
2.5.3.12 XML(수록정보) .....	143
2.5.3.13 XML(분류/항목) .....	144
2.5.3.14 XML(주석) .....	144
2.5.3.15 XML(단위) .....	145
2.5.3.16 XML(출처) .....	145
2.5.3.17 XML(가중치) .....	146

2.5.3.18 XML(자료갱신일) .....	146
<b>2.6 KOSIS통합검색 .....</b>	<b>147</b>
2.6.1 특징 .....	147
2.6.2 서비스활용 .....	147
2.6.2.1 URL생성 .....	147
2.6.3 활용방법 .....	148
2.6.3.1 JSON .....	148
<b>2.7 통계주요지표 .....</b>	<b>149</b>
2.7.1 특징 .....	149
2.7.2 서비스활용 .....	149
2.7.2.1 URL생성 .....	149
2.7.3 활용방법 .....	150
2.7.3.1 JSON,XML(지표고유번호별 설명자료조회) .....	150
2.7.3.2 JSON,XML(지표명별 설명자료조회) .....	151
2.7.3.3 JSON,XML(목록별 지표조회) .....	152
2.7.3.4 JSON,XML(지표명별 목록조회) .....	153
2.7.3.5 JSON,XML(고유번호별 목록조회) .....	154
2.7.3.6 JSON,XML(고유번호별 지표 상세조회) .....	155
2.7.3.7 JSON,XML(수록주기별 목록조회) .....	156
2.7.3.8 JSON,XML(지표명별 상세조회) .....	157
<b>&lt;참고&gt; SDMX .....</b>	<b>158</b>

## 1. KOSIS 공유서비스(OpenAPI) 개요

KOSIS 공유서비스는 외부(기관, 개인)에서 국가통계포털(KOSIS)의 통계정보를 서비스 또는 컨텐츠 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, 작성증지통계, 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

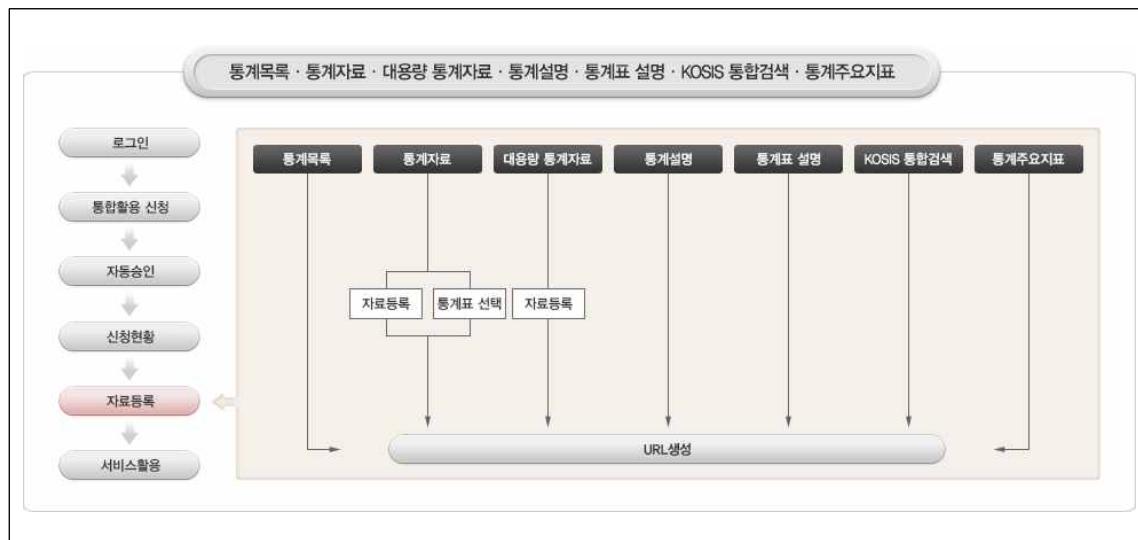
KOSIS 공유서비스를 활용하려면 국가통계포털(KOSIS) 회원으로 가입되어야 합니다. 기존 국가통계포털에 회원으로 가입된 사용자는 그대로 사용하시면 됩니다.

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.

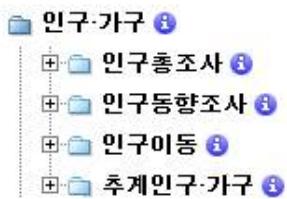


## 1.1 제공 콘텐츠

### 1.1.1 통계목록

#### ■ 개요

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



#### ■ 서비스 내용

서비스별	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	지방자치단체외국인주민현황		

### 1.1.2 통계자료

#### ■ 개요

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

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)	10,003,803	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.

① 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 >

## < 주석 >

단위 국문명	단위 영문명
명	Person

### < 단위 >

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

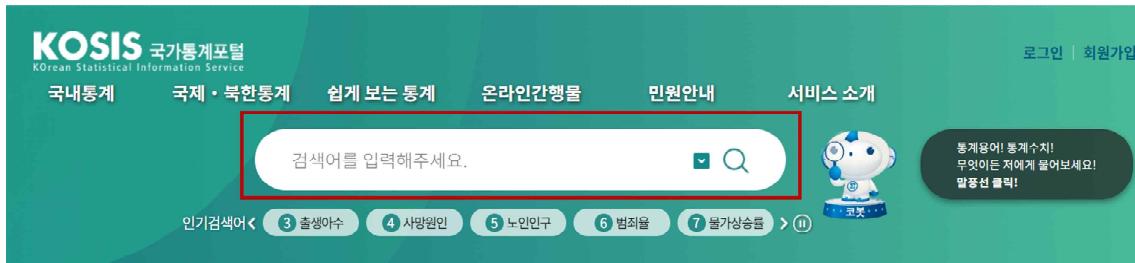
### < 출처 >

## 1.1.6 KOSIS통합검색

### ■ 개요

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

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



### ■ 서비스 내용

결과번호	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 농생활불가지수 농산물및석유류제외지수 생활불가지수 식료품및에너지제외지수 신선식품지수 전년비 총지수
수록기간 시작일	1966
수록기간 종료일	2018
통계표 주석	- 생활불가지수 전체 소비자물가 460개 품목 중 체감률을 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, 재소류, 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 균원인플레이션 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)	<a href="http://kosis.kr/statisticsList/statisticsListIndex.do?menuId=M_01_01&amp;vwcd=MT_ZTITLE&amp;parmTabId=M_01_01&amp;parentId=C.1;C1.2;C1_15.3;">http://kosis.kr/statisticsList/statisticsListIndex.do?menuId=M_01_01&amp;vwcd=MT_ZTITLE&amp;parmTabId=M_01_01&amp;parentId=C.1;C1.2;C1_15.3;</a>
통계표 이동URL (KOSIS 통계표로 이동)	<a href="http://kosis.kr/statHtml/statHtml.do?orgId=101&amp;tblId=DT_1J17042">http://kosis.kr/statHtml/statHtml.do?orgId=101&amp;tblId=DT_1J17042</a>
검색결과 건수	2399
검색어명	물가

### 1.1.7 통계주요지표

#### ■ 개요

통계표의 목록구성 정보 제공을 위한 OpenAPI입니다. 통계목록 단위로 호출하고 서비스별(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(15~64 세)] × 100	통계청, 장래 인구추계

< 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(증위추계, 총인구)	명	전국	년	1960	2067	108	2067년
94	추계인구(증위추계, 남자)	명	전국	년	1960	2067	108	2067년

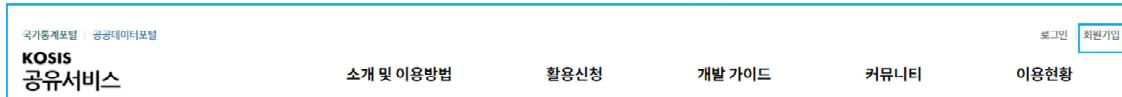
< 수록주기별 목록조회 >

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.



- 페이지 상단의 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 ①등록버튼을 클릭합니다.
- ②의 [약관보기]를 클릭하면 통계정보 활용약관 팝업이 호출됩니다.

\* 는 필수입력 항목입니다.

신청자명	김순영	전화번호	
서비스 유형	REST	심의여부	자동승인
활용용도 *	<input type="radio"/> 웹개발 <input type="radio"/> 앱개발 <input checked="" type="radio"/> 기타 <b>테스트</b>	활용기간	인증기 만료일까지 활용 가능
상업적활용여부 *	<input type="radio"/> 예 <input checked="" type="radio"/> 아니오		
<small>* 사용목적을 불성실하게 작성시 활용신청 승인이 취소될 수 있습니다. 상세히 작성해 주시기 바랍니다. (2,000자 이내)</small> <small>사용목적을 반드시 작성해 주십시오.</small>			
사용목적 *			
첨부파일	<small>* 첨부파일 용량은 10Mb이하로 한 개의 파일만 첨부할 수 있습니다. [xls, doc, hwp, txt, gif, jpg, jpeg, png, bmp 파일만 가능합니다.]</small> <input type="button" value="파일 선택"/> <input type="button" value="선택된 파일 없음"/> <input type="button" value="X"/>		
약관동의	<b>2</b> <input type="checkbox"/> 통계정보 활용약관에 동의합니다. <input type="button" value="약관보기"/>		
<b>1</b> <input type="button" value="등록"/> <input type="button" value="취소"/>			

### 1.3.2 신청현황

- 활용신청이 완료되면 신청현황 단계로 넘어가며, 기본정보에서는 사용자 인증키가 발급된 것을 확인이 가능하고, 서비스정보, 활용정보를 확인 할 수 있습니다. ①신청정보 수정을 누르면 신청 정보 수정 단계로 넘어갑니다.

기본정보			
서비스유형	REST	활용기간	인증기 만료일까지 활용 가능
설의여부	자동승인	상업적활용유무	비활용
사용자 인증키	NzRIY2l3NTZkNTM0NGI[REDACTED]		

활용정보	
활용용도	기타 (테스트) 테스트용
활용목적	
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
40	호출가능건수 제한	관리자에게 문의
41	호출가능ROW수 제한	관리자에게 문의
42	사용자별 이용 제한	관리자에게 문의
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**

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

상세기능 +
**URL 생성 -**
[개발가이드 다운로드](#)

**URL생성 조건 설정**

목록 구분	<input style="border: 1px solid #ccc; padding: 2px; width: 100%; height: 15px;" type="button" value="국내통계 주제별"/>
목록 조회	<input style="width: 150px; height: 25px; border: 1px solid #ccc; border-radius: 5px; padding: 2px; margin-right: 10px;" type="text" value="목록ID A"/> <input style="border: 1px solid #ccc; padding: 2px; width: 100px; height: 25px;" type="button" value="목록 검색"/>
목록명	<input style="width: 150px; height: 25px; border: 1px solid #ccc; border-radius: 5px; padding: 2px;" type="text" value="인구"/>
결과유형	<input style="border: 1px solid #ccc; padding: 2px; width: 50px; height: 25px;" type="button" value="SDMX"/>

[URL 복사](#)
[결과값보기](#)

**생성된 URL**

```
https://kosis.kr/openapi/statisticsList.do?
method=list&apiKey=NzRIV2j3NTZkNTM0N[REDACTED]&vwCd=MT_ZTITLE&parentListId=A&format=sdmx&jsonVd=Y&version=v2_1
```

**데이터 보기**

서비스별	목록ID	목록명	기관코드	등계표ID	등계조사ID	최종갱신일	추천통계표여부
MT_ZTITLE	A_4	인구총조사	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_7	주민등록인구현황	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_3	인구동향조사	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_6	장래인구추계	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_1	국내인구이동통계	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_9	출입국자및체류외국인통계	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_8	지방자치단체외국인주민현황	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_5	장래가구추계	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_2	국제인구이동통계	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>
MT_ZTITLE	A_001	고령친화용품제조업실태조사	<해당없음>	<해당없음>	<해당없음>	<해당없음>	<해당없음>

**결과값보기**

```
<xml version="1.0" encoding="UTF-8" ?><message:Structure xmlns:messages="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:><message:Test><false></message:Test><message:Prepared>2023-10-26T13:14:55</message:Prepared>
<message:Sender id="KOSIS" /><common:Name>통계청</common:Name></message:Sender><common:Name>국내통계 주제별</common:Name></message:Header>
<message:Structures><structure:OrganisationSchemes><structure:AgencyScheme id="AGENCIES" agencyId="SDMX" version="1.0"><common:Name xml:lang="en">SDMX Agency Scheme</common:Name><common:Name xml:lang="en">KOSIS</common:Name></structure:Agency><structure:CategorySchemes>
```

[통계서비스 기획과](#)

### 2.1.3 활용방법

#### 2.1.3.1 JSON

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

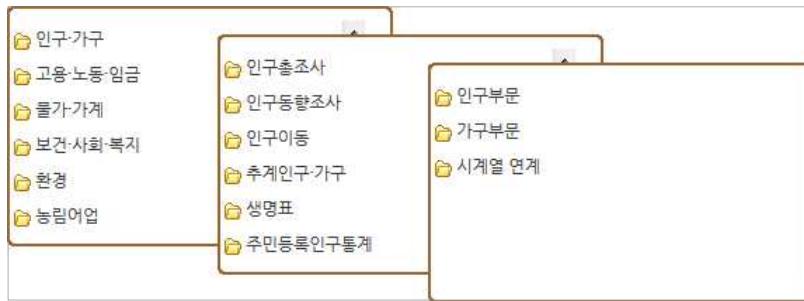
■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
vwCd	String	서비스류 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
VW_CD	서비스류ID	VARCHAR2(40)
VW_NM	서비스류명	VARCHAR2(300)
LIST_ID	목록ID	VARCHAR2(40)
LIST_NM	목록명	VARCHAR2(300)
ORG_ID	기관코드	VARCHAR2(40)
TBL_ID	통계표ID	VARCHAR2(40)
TBL_NM	통계표명	VARCHAR2(300)
STAT_ID	통계조사ID	VARCHAR2(40)
SEND_DE	최종갱신일	VARCHAR2(8)
REC_TBL_SE	추천 통계표 여부	VARCHAR2(10)

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

<script type="text/javascript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide01/js/dojo.js" ></script>
<script type="text/javascript" src="https://mgmk.kosis.kr/openapi_dev/devGuide/devGuide01/js/json.js" ></script>
<script type="text/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 - 서비스류 코드 (통계목록구분)
 * listLev - 목록 레벨
 * parentId - 시작목록 Id
*/
function getSubList(vwcd, listLev, parentId) {

    // ajax 통신을 위한 파라메터를 변수에 담는다.
    var paraObj = {
        // 임의의 jsp 페이지를 호출함으로써 cross domain 제약을 우회할 수 있다. (devGuidePop.jsp 소스는 소스 하단에 제공)
        url : "http://[개발자 주소]/devGuidePop.jsp?method=getList&key=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&vwcd=MT_ZTITLE&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 href=W'http://kosis.kr/start.jsp?orgId='"+mapData[cnt].ORG_ID+"&tBId='"+mapData[cnt].TBL_ID+"&vw_cd='"+mapData[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=javascript:getSubList('"+mapData[cnt].VW_CD+", "" + listLev + "", "+mapData[cnt].LIST_ID+");W">" +mapData[cnt].LIST_NM+"</a> ";
    }
    nodeInfo = nodeInfo + "</li>";
}
nodeInfo = nodeInfo+"
```

```

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="parentId" ><%out.println(request.getParameter("parentId") == null ? "" : (request.getParameter("parentId")));%></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=${vwcd}&
parentListId=${parentId}&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'           # 서비스별코드
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 %>% !()
    )
  )
```

```

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

df_lists <- data.frame(서비스류 = 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 설정 함수
        def Setting(self, parentId):
            #url을 통해 json 데이터 가져오기
            with urlopen(
                "https://kosis.kr/openapi/statisticsList.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRj"

```

```
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	서비스별 코드 · MT_ZTITLE : 주제별 통계 · MT_OTITLE : 기관별 통계 · MT_CHOSUN_TITLE : 광복이전통계(1908~1943) · MT_HANKUK_TITLE : 대한민국통계연감 · MT_STOP_TITLE : 작성종지통계 · 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			서비스별ID
	Name			서비스별명
	Prepared			전송시간
	Sender	Id		전송기관
		Name		전송기관명
Structures	Category Schemes	Category Scheme	Id	상위목록 ID
			Name	상위목록 명
			Category	Id
			Category	Name
			Description	Description
			StatId	StatId
			SendDe	SendDe
			RecTbISe	RecTbISe

## ■ 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 - 서비스류 코드 (통계목록구분)  

* listLev - 목록 레벨  

* parentId - 시작목록 Id  

*****/  

function getSubList(vwcd, listLev, parentId) {  
  

    // ajax 통신을 위한 호출 함수  

$.ajax({  

    type: "GET",  

    u :  

    "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsList.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOW  

    RjNjgxN2MzNDgwNmY=&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='_balnk' 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'                      # 서비스별코드
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 함으로서 이용할 자료를 미리 등록하는 과정이 불필요한 방식입니다.(등록한 자료의 통계표 메타정보가 변경될 경우 확인 불가능)
- URL 자료등록 방식은 생성내역의 통계표 및 URL이 저장되어 등록한 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

동계자료 API를 이용하는 요청번호, 출력변수, 생플데이터를 제공하며 통계표선택(URL 생성) 방법 (분류와 항목정보가 파라메터로 URL에 포함되어 파라메터를 변경해가며 이용 가능)과 자료등록(자료등록 URL 생성) 방법 (자료등록 후 이용하여 하며 URL 변경 불가) 2가지 방법으로 활용 가능합니다. 출력포맷으로 JSON과 SDMX를 제공합니다.

상세기능 +
**URL 생성** -
[개발가이드 다운로드](#)

**URL 생성** **자료등록**

#### 1 통계표 선택

작성기관	<input type="text"/>	<a href="#">자성기관 검색</a>	통계조사명	<input type="text"/>	<a href="#">통계조사명 검색</a>
통계표명	<input type="text"/>	통계표ID	DT_1B08024		
분류	<input type="text"/>	주기	<a href="#">전체</a> ▾		
분류값/항목					

**검색**

기관명	통계표명	통계표ID	통계조사명	수록기간	선택	통계표조회
통계청 (101)	인구밀도(인구주택총조사기준)	DT_1B08024	인구총조사	년 1966-2022	<input checked="" type="checkbox"/> 선택	

**1**
**2**

#### 3 URL생성 조건 설정

통계표명	<input type="text" value="인구밀도(인구주택총조사기준)"/>	통계표ID	<input type="text" value="DT_1B08024"/>						
분류/항목선택	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">분류</td> <td style="width: 50%;">항목 <a href="#">전체선택</a> <a href="#">해제</a></td> </tr> <tr> <td><input type="text" value="행정구역별"/></td> <td><input checked="" type="radio"/> 개별 <input type="radio"/> 선택 <input checked="" type="checkbox"/> 인구밀도</td> </tr> <tr> <td colspan="2">선택한 분류값 : 전국</td> </tr> </table>			분류	항목 <a href="#">전체선택</a> <a href="#">해제</a>	<input type="text" value="행정구역별"/>	<input checked="" type="radio"/> 개별 <input type="radio"/> 선택 <input checked="" type="checkbox"/> 인구밀도	선택한 분류값 : 전국	
분류	항목 <a href="#">전체선택</a> <a href="#">해제</a>								
<input type="text" value="행정구역별"/>	<input checked="" type="radio"/> 개별 <input type="radio"/> 선택 <input checked="" type="checkbox"/> 인구밀도								
선택한 분류값 : 전국									

출력형태 설정	<table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td>데이터 포맷 <input checked="" type="radio"/> JSON <input type="radio"/> SDMX</td> </tr> <tr> <td>사용자 인증키 <input type="text" value="NzRlY2l3NT..."/></td> </tr> </table>	데이터 포맷 <input checked="" type="radio"/> JSON <input type="radio"/> SDMX	사용자 인증키 <input type="text" value="NzRlY2l3NT..."/>
데이터 포맷 <input checked="" type="radio"/> JSON <input type="radio"/> SDMX			
사용자 인증키 <input type="text" value="NzRlY2l3NT..."/>			

조회기간 설정	<table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td><input type="radio"/> 기간설정 <input checked="" type="radio"/> 최근시점기준</td> </tr> <tr> <td>최근시점부터 주기 년 ▾ 최근 3 개 자료</td> </tr> <tr> <td>수록시점간격 <small>최근시점 조정에 사용할 수 있는 철자로 현재 시점부터 사용할 기준시점까지의 간격을 입력 ex) 2019, 2017, 2015 등 2개 시점 간격으로 추출시 [2] 입력</small></td> </tr> </table>	<input type="radio"/> 기간설정 <input checked="" type="radio"/> 최근시점기준	최근시점부터 주기 년 ▾ 최근 3 개 자료	수록시점간격 <small>최근시점 조정에 사용할 수 있는 철자로 현재 시점부터 사용할 기준시점까지의 간격을 입력 ex) 2019, 2017, 2015 등 2개 시점 간격으로 추출시 [2] 입력</small>
<input type="radio"/> 기간설정 <input checked="" type="radio"/> 최근시점기준				
최근시점부터 주기 년 ▾ 최근 3 개 자료				
수록시점간격 <small>최근시점 조정에 사용할 수 있는 철자로 현재 시점부터 사용할 기준시점까지의 간격을 입력 ex) 2019, 2017, 2015 등 2개 시점 간격으로 추출시 [2] 입력</small>				

응답필드 설정 [선택사항]	<table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td><input type="checkbox"/> 기관코드</td> <td><input type="checkbox"/> 통계표ID</td> <td><input type="checkbox"/> 통계표명</td> <td><input type="checkbox"/> 분류값 ID</td> <td><input type="checkbox"/> 분류명</td> </tr> <tr> <td><input type="checkbox"/> 분류 명문명</td> <td><input type="checkbox"/> 분류값 영</td> <td><input type="checkbox"/> 분류값 영문</td> <td><input type="checkbox"/> 항목 ID</td> <td><input type="checkbox"/> 항목명</td> </tr> <tr> <td><input type="checkbox"/> 항목명문명</td> <td><input type="checkbox"/> 단위명</td> <td><input type="checkbox"/> 단위명문명</td> <td><input type="checkbox"/> 수록주기</td> <td><input type="checkbox"/> 수록시점</td> </tr> <tr> <td><input type="checkbox"/> 자료갱신일</td> <td colspan="4"></td> </tr> </table>	<input type="checkbox"/> 기관코드	<input type="checkbox"/> 통계표ID	<input type="checkbox"/> 통계표명	<input type="checkbox"/> 분류값 ID	<input type="checkbox"/> 분류명	<input type="checkbox"/> 분류 명문명	<input type="checkbox"/> 분류값 영	<input type="checkbox"/> 분류값 영문	<input type="checkbox"/> 항목 ID	<input type="checkbox"/> 항목명	<input type="checkbox"/> 항목명문명	<input type="checkbox"/> 단위명	<input type="checkbox"/> 단위명문명	<input type="checkbox"/> 수록주기	<input type="checkbox"/> 수록시점	<input type="checkbox"/> 자료갱신일				
<input type="checkbox"/> 기관코드	<input type="checkbox"/> 통계표ID	<input type="checkbox"/> 통계표명	<input type="checkbox"/> 분류값 ID	<input type="checkbox"/> 분류명																	
<input type="checkbox"/> 분류 명문명	<input type="checkbox"/> 분류값 영	<input type="checkbox"/> 분류값 영문	<input type="checkbox"/> 항목 ID	<input type="checkbox"/> 항목명																	
<input type="checkbox"/> 항목명문명	<input type="checkbox"/> 단위명	<input type="checkbox"/> 단위명문명	<input type="checkbox"/> 수록주기	<input type="checkbox"/> 수록시점																	
<input type="checkbox"/> 자료갱신일																					

**3**
**4**

**4 URL보기**

- 생성된 URL

```
https://kosis.kr/openapi/Param/statisticsParameterData.do?
method=list&apiKey=NzRlY2l3NTkNTMONGRMzU3ZTVhN2E5NDaxNzYwZG=&itmlId=17+&objJ1=00&objJ2=&objJ3=&objJ4=&objJ5=&objJ6=&objJ7=&objJ8=&format=json&jsonVd=Y&prdSe=Y&newEstPrdCnt=3&orgId=101&tblId=DT_1B08024
```

**5**
**URL복사**

[통계서비스기획과](#)

KOSIS 공유서비스 개발가이드

34

## 2.2.2.2 자료등록 URL 생성

### 2.2.2.2.1 자료 등록

#### ■ 개발가이드 > 통계자료 > URL생성 > 자료등록

'작성기관', '통계조사명', '통계표명' 등을 입력하여 사용하고자 하는 자료를 조회한 뒤 조회결과에서 등록할 자료의 ①사용여부 항목을 선택하고 ②통계표 등록 버튼을 누른다.  
 ③통계표조회 버튼을 클릭하면 해당 자료의 통계표를 볼 수 있습니다.

### 통계자료

통계자료 API를 이용하는 요청변수, 출력변수, 생莼데이터를 제공하며 통계표선택(URL 생성) 방법 (분류와 항목정보가 파라메터로 URL에 포함되어 파라메터를 변경해가며 이용 가능)과 자료등록(자료등록 URL 생성) 방법 (자료등록 후 이용하여야 하며 URL 변경 불가) 2가지 방법으로 활용 가능합니다. 출력포맷으로 JSON과 SDMX를 제공합니다.

상세기능 +
**URL 생성** -


URL 생성
**자료등록**

**I 통계표 선택**

작성기관	통계청	작성기관 검색	통계조사명	인구총조사	통계조사명 검색
통계표명			통계표ID		
분류			주기	전체	
분류값/항목			등록한 표만 보기	<input type="checkbox"/>	

**검색**

Total : 1875 | Page : 1/188

기관명	통계표명	통계표ID	통계조사명	수록기간	선택	통계표조회
등계청 (101)	인구밀도(인구주택총조사기준)	DT_1B08024	인구총조사	년 1966-2022	<input checked="" type="checkbox"/>	
등계청 (101)	행정구역/기구규모/성별 등근-동학인구(12세이상)	DT_1B90001	인구총조사	5년 1995-1995	<input checked="" type="checkbox"/>	
등계청 (101)	대도시/등근등학/성/이용교동수단별 등근-동학인구(12세이상)	DT_1B90003	인구총조사	5년 1995-1995	<input checked="" type="checkbox"/>	
등계청 (101)	시도/소요시간/이용교동수단별 등근-동학인구(12세이상)	DT_1B90005	인구총조사	5년 1995-1995	<input type="checkbox"/>	
등계청 (101)	시도/성/등학/이용교동수단별 등근-동학인구(12세이상)	DT_1B90007	인구총조사	5년 1995-1995	<input type="checkbox"/>	
등계청 (101)	시도/성/직업/이용교동수단별 등근-동학인구(12세이상)	DT_1B90008	인구총조사	5년 1995-1995	<input type="checkbox"/>	
등계청 (101)	시도/성/산업/이용교동수단별 등근-동학인구(12세이상)	DT_1B90009	인구총조사	5년 1995-1995	<input type="checkbox"/>	
등계청 (101)	시도/성/소요시간/학교별 등근-동학인구(12세이상)	DT_1B9000A	인구총조사	5년 1995-1995	<input type="checkbox"/>	
등계청 (101)	현거주지/성/목적지별 등근-동학인구(서울,인천,경기,12세…)	DT_1B9000D	인구총조사	5년 1995-1995	<input type="checkbox"/>	
등계청 (101)	현거주지/성/목적지별 등근-동학인구(부산,경남,12세이상)	DT_1B9000E	인구총조사	5년 1995-1995	<input type="checkbox"/>	

< 1 2 3 4 5 6 7 8 9 10 >

2 등록

KOSIS 공유서비스 개발가이드

35

### 2.2.2.2 등록된 자료

#### ■ 마이페이지 > 등록한 자료 > 통계자료 > 등록된 자료

자료등록의 통계표 등록을 마치면, 이용자가 등록한 자료들의 목록이 나타나며, 등록된 자료 중 URL생성을 원하는 자료의 ①URL생성 버튼을 누른다.

②통계표조회 버튼을 클릭하면 해당 자료의 통계표를 볼 수 있습니다.

**등록한 자료**

[활용신청현황](#) [나의 Q&A](#) [나의 활용사례](#) [나의 통계표변동내역](#) [나의 통계표URL변동내역](#) **등록한 자료**

**통계자료** -
대용량통계자료 +

**통계표 선택**

작성기관	<input type="text"/>	작성기관 검색	통계조사명	<input type="text"/>	통계조사명 검색
통계표명	<input type="text"/>	통계표ID	<input type="text"/>		
분류	<input type="text"/>	주기	전체		
분류값/항목	<input type="text"/>				

**검색**

Total : 10 | Page : 1/1

기관명	통계표명	통계표ID	통계조사명	수록기간	신청일자	통계표조회	URL생성
통계청 (101)	경지규모별 농가 및 경지면적	DT_INJ101	농림어업총조사	5년 2005~2005	2023-09-27 08:19:01	 2	 1
통계청 (101)	 인간개발지수	DT_2KAAB01	국제농계연감	년 1990~2021	2022-10-04 16:51:23		

#### ■ 마이페이지 > 등록한 자료 > 통계자료 > 등록된 자료 > 삭제

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

**등록한 자료**

[활용신청현황](#) [나의 Q&A](#) [나의 활용사례](#) [나의 통계표변동내역](#) [나의 통계표URL변동내역](#) **등록한 자료**

**통계자료** -
대용량통계자료 +

**통계표 선택**

작성기관	<input type="text"/>	작성기관 검색	통계조사명	<input type="text"/>	통계조사명 검색
통계표명	<input type="text"/>	통계표ID	<input type="text"/>		
분류	<input type="text"/>	주기	전체		
분류값/항목	<input type="text"/>				

**검색**

Total : 10 | Page : 1/1

기관명	통계표명	통계표ID	통계조사명	수록기간	신청일자	통계표조회	URL생성
통계청 (101)	경지규모별 농가 및 경지면적	DT_INJ101	농림어업총조사	5년 2005~2005	2023-09-27 08:19:01		
통계청 (101)	 인간개발지수	DT_2KAAB01	국제농계연감	년 1990~2021	2022-10-04 16:51:23		

사용자 생성 URL 목록

활용자료명	URL 상세 정보	삭제	URL보기
인간개발지수	기대수명(세)/HDI 순위/인간개발지수/평균교육기간(... 	 2	

### 2.2.2.2.3 URL생성

- 마이페이지 > 등록한 자료 > 통계자료 > 등록된 자료 > 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.

활용신청현황
나의 Q&A
나의 활용사례
나의 통계표변동내역
나의 통계표URL변동내역
등록한 자료

통계자료
-
+
내용량통계자료

**URL생성 조건 설정**

통계표명	경지규모별 농가 및 경지면적	통계표ID	DT_INJ101
활용 자료명	경지규모별 농가 및 경지면적		
분류/항목선택	<div style="display: flex; justify-content: space-between;"> <div style="flex: 1;"> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>행정구역별</p> <p>&gt; 선택된 분류값: 전국</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>특성별</p> <p>&gt; 선택된 분류값: 계</p> </div> </div> <div style="flex: 1;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <span style="border: 1px solid #ccc; padding: 2px 5px; border-radius: 5px;">분류</span> <span style="border: 1px solid #ccc; padding: 2px 5px; border-radius: 5px;">항목</span> <span style="border: 1px solid #ccc; padding: 2px 5px; border-radius: 5px;">전체선택</span> <span style="border: 1px solid #ccc; padding: 2px 5px; border-radius: 5px;">해제</span> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <input checked="" type="checkbox"/> 계  <input type="checkbox"/> 경지없는농가  <input type="checkbox"/> 경지규모별농가:소계  <input type="checkbox"/> 0.1ha미만  <input type="checkbox"/> 0.1~0.2         </div> </div> </div>		

1
URL생성
취소

**등록한 자료**

활용신청현황
나의 Q&A
나의 활용사례
나의 통계표변동내역
나의 통계표URL변동내역
등록한 자료

등계자료
-
대용량등계자료
+

**1 URL생성 조건 설정**

통계표명	<input type="text" value="행정구역(시군구)별, 성별 인구수"/>	통계표ID	<input type="text" value="DT_1B040A3"/>
<b>출력형태 설정</b>			
데이터 포맷	<input checked="" type="radio"/> JSON <input type="radio"/> SDMX		
<b>조회기간 설정</b>			
최근시점부터	주기	월	최근 3 개 자료
<b>수록시점간격</b> <small>최근시점 조정에 사용할 수 있는 항목으로 현재 시점부터 사용할 기준시점까지의 간격을 입력 ex) 2019, 2017, 2015 등 2개 시점 간격으로 추출시 [2] 입력</small>			
활용자료명	URL 상세 정보	<span style="border: 1px solid #0070C0; border-radius: 50%; padding: 2px 5px; margin-right: 5px;">2 URL보기</span> <span style="border: 1px solid #0070C0; border-radius: 50%; padding: 2px 5px; margin-right: 5px;">3 결과값보기</span>	
행정구역(시군구)별, 성별 인구수	총인구수 * 전국	<span style="border: 1px solid #0070C0; border-radius: 50%; padding: 2px 5px; margin-right: 5px;">URL보기</span> <span style="border: 1px solid #0070C0; border-radius: 50%; padding: 2px 5px; margin-right: 5px;">결과값보기</span>	

재신청

**- 생성된 URL**

```
https://kosis.kr/openapi/statisticsData.do?
method=getList&apiKey=NtI0Yjk1MGJ3[REDACTED]&format=json&jsonVD=Y&userStatsId=rIagyrI92/101/DT_1B040A3/2/5/20231026134415&prdSe=M&newEstPrdCnt=3
```

**- 데이터 보기**

행정구역(시군구)별, 성별 인구수(202309)	총인구수 (명)
행정구역(시군구)별	51370001
전국	

3 URL복사

## 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	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	종료수록시점	
최신자료 기준	newEstPrd Cnt	최근수록시점 개수	
	prdInterval	수록시점 간격	
format	String	결과 유형(json, sdmx)	필수
content	String	헤더 유형(html, json)	선택

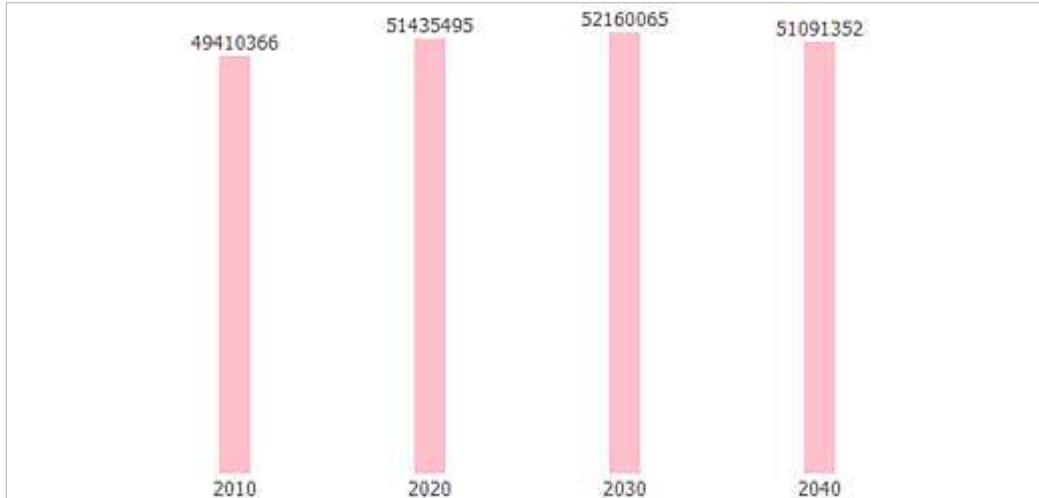
#### ● 통계표선택 방법

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	시작수록시점	선택 (시점기준 또는 최신자료기준 택1)
	endPrdDe	종료수록시점	
최신자료 기준	newEstPrd Cnt	최근수록시점 개수	
	prdInterval	수록시점 간격	
format	String	결과 유형(json, sdmx)	필수

■ 출력 변수

항목명(영문)	항목설명	형식	비고
ORG_ID	기관코드	VARCHAR2(40)	
TBL_ID	통계표ID	VARCHAR2(40)	
TBL_NM	통계표명	VARCHAR2(300)	
C1 ~ C8	분류값 ID1 ~ 분류값 ID8	VARCHAR2(40)	2~8 분류값은 없을 경우 생략
C1_OBJ_NM ~ C8_OBJ_NM	분류명1 ~ 분류명8	VARCHAR2(3000)	
C1_OBJ_NM_ENG ~ C8_OBJ_NM_ENG	분류 영문명1 ~ 분류 영문명8	VARCHAR2(3000)	
C1_NM ~ C8_NM	분류값 명1 ~ 분류값 명8	VARCHAR2(3000)	
C1_NM_ENG ~ C8_NM_ENG	분류값 영문명1 ~ 분류값 영문명8	VARCHAR2(3000)	
ITM_ID	항목 ID	VARCHAR2(40)	
ITM_NM	항목명	VARCHAR2(3000)	
ITM_NM_ENG	항목영문명	VARCHAR2(3000)	
UNIT_ID	단위ID	VARCHAR2(40)	
UNIT_NM	단위명	VARCHAR2(1000)	
UNIT_NM_ENG	단위영문명	VARCHAR2(1000)	
PRD_SE	수록주기	VARCHAR2(20)	
PRD_DE	수록시점	VARCHAR2(8)	
DT	수치값	VARCHAR2(100)	
LST_CHN_DE	최종수정일	VARCHAR2(8)	

■ 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 파라메터를 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&n
ewEstPrdCnt=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 그리기위한 옵션을 정의
var options = {'BarSize': '20px', 'BarSpace': '2px', 'type' : 'horizontal', 'Font': '2px'};

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

    alert(dojo.toJson(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()) # 모든 변수 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 %>% !(),
      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
jNjgxN2MzMzNDgwNmY=&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

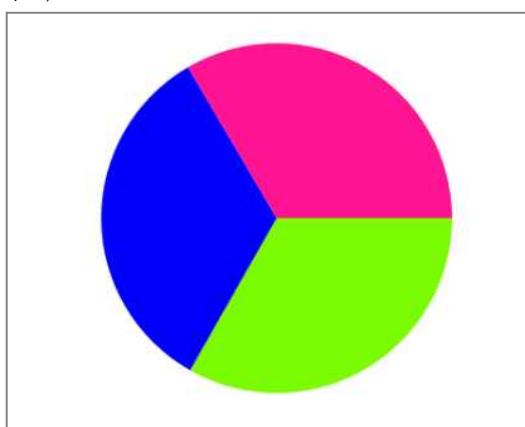
#Bar차트 그리기
plt.bar(xAxis, yAxis)
plt.title(title)

#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 파라메터를 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=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgx
N2MzNDgwNmY=&format=json&userStatsId=openapisample/101/DT_1IN1502/2/1/20191106094026_1&prdSe=Y&n
ewEstPrdCnt=3",
sync : true,
load : function(resObj, a, b) {
}
}

```

```

// 차트에 보여주기위해 값을 담고있을 배열변수를 선언
mapData = resObj;
// 차트를 생성하기위한 함수 호출
chartShow();
},
error : function ( resObj, e ) {

alert(dojo.toJson(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 = ["#7cfcc00", "#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 출력하기위한 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 %>% !(),
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)
)

# 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=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
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)



#### ■ 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/xhtml

```

```

1-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/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">

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

getList();
});

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

// ajax 통신을 위한 파라메터를 변수에 담는다. (통계자료)
var paraObj = {
// 임의의 jpg 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) {

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:500px;'>";
strTable += "<thead>";
strTable += "<tr>";
strTable += "<th class='fst al_C' 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='al_C'>" + mapData[i].PRD_DE + "</td>";
    strTable += "<td class='al_C'>" + 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 파라메터를 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=ZjZjOTI3MjRjNmU1YzdhZTMwOW
RjNjgxN2MzNDgwNmY=&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()) # 모든 변수 초기화
v_apiKey = Sys.getenv('KOSIS_TOKEN')
baseurl <- 'https://kosis.kr/openapi/statisticsData.do' # 요청 URL

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

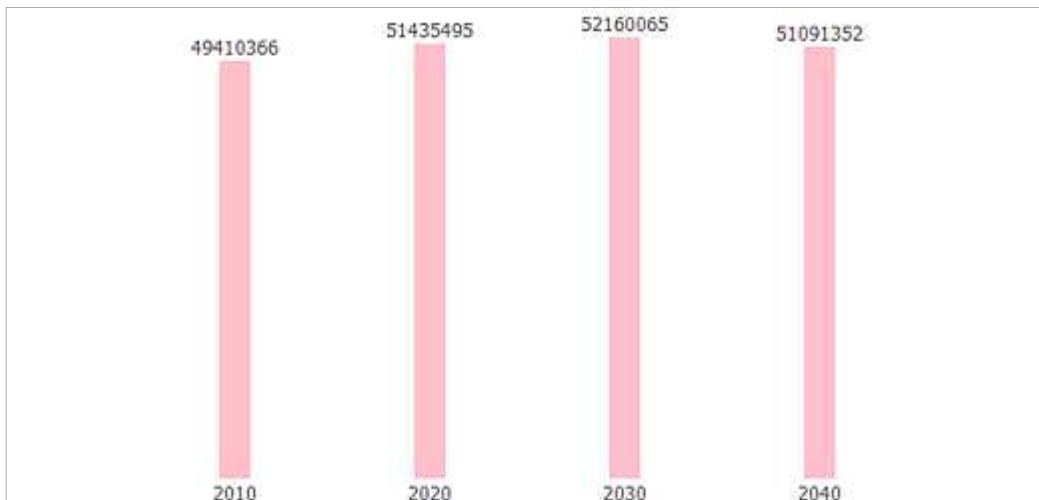
항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
orgId	String	기관 ID	필수
tblId	String	통계표 ID	필수
objL2 ~ objL8	String	분류2(두번째 분류코드) ~ 분류8(여덟째 분류코드)	선택
itmId	String	항목	필수
type	String	SDMX의 유형(DSD, Generic, StructureSpecific)	필수
format	String	결과 유형(json, sdmx)	필수
version	String	결과값 구분	생략시 구버전으로 데이터 출력

■ 출력 변수

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

			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	디멘션ID
				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/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()) # 모든 변수 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 %>% !(),
    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(st
at = "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=ZjZjOTI3MjRjNmU1YzdhZTMwOWRj
NjgxN2MzNDgwNmY=&format=sdmx&type=StructureSpecific&userStatsId=openapisample/101/DT_1IN1502/2/1/201
91106094026_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)

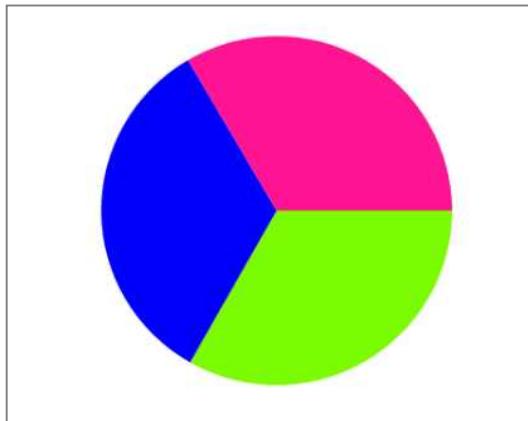
#y축 수치를 안보이게 하는 코드. 필요에 따라 선택하여 사용
plt.gca().axes.yaxis.set_visible(False)

#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 = ["#7fcfc00", "#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 출력하기위한 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 %>% !(),
    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)

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)



#### ■ 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, strItmId, 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") {

strItmId = 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=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&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 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=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNigxN2MzNDgwNmY=&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()) # 모든 변수 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 %>% l(),
    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)

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 기준	newEstPrdCnt	String	최근수록시점 개수	최신자료기준 택1)
	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수
version		String	결과값 구분	생략시 구버전으로 데이터 출력

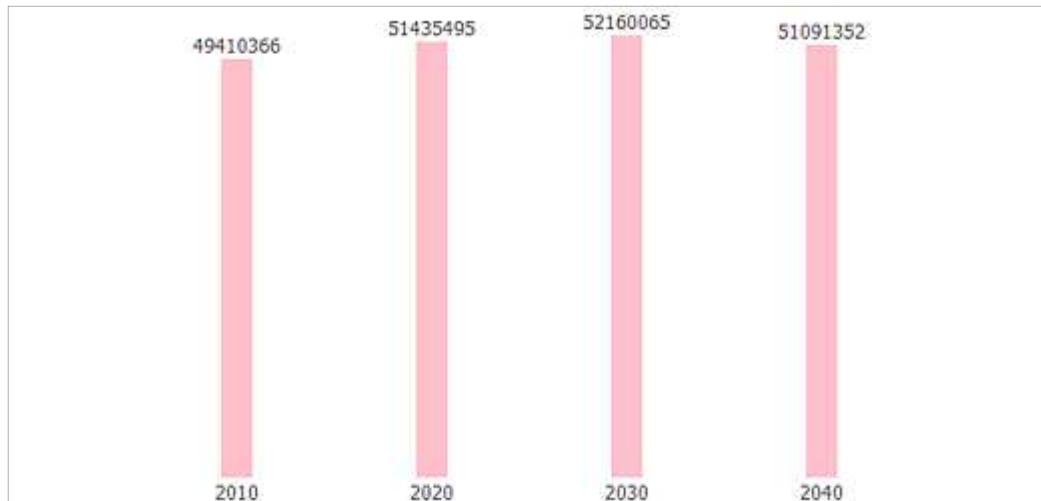
#### ● 통계표선택 방법

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 기준	newEstPrdCnt	String	최근수록시점 개수	최신자료기준 택1)
	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	수치자료값
		LstChnDe	Value	최종수정일

■ 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 # 통계 표명
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)

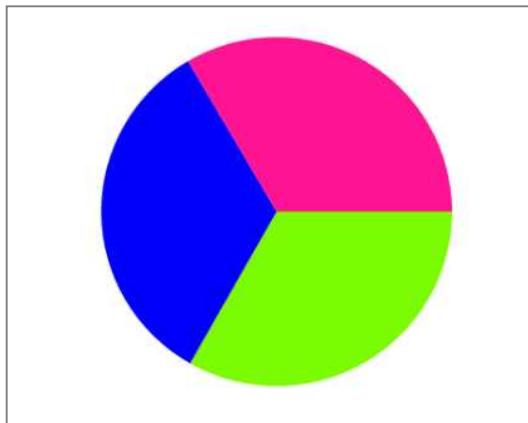
#y축 수치를 안보이게 하는 코드. 필요에 따라 선택하여 사용
plt.gca().axes.yaxis.set_visible(False)

#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=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzMzNDgwNmY=&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 = ["#7fcfc00", "#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 출력하기위한 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 %>% l(),
    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)

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)



■ 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, stritmld, 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=ZjZjOTI3MjRjNmU1YzdhZTM
wOWRjNjgxN2MzNDgwNmY=&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") {

        strItmId = 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=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 al_C' scope='col'>시점</th>";
        strTable += "<th scope='col'>" + itmNm + "(" + object.documentElement.childNodes[1].childNodes[0].childNod
es[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 %>% !(),  
      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)  
  
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 기준	newEstPrdCnt	String	최근수록시점 개수	최신자료기준 택1)
	prdInterval	String	수록시점 간격	
format		String	결과 유형(json, sdmx)	필수
version		String	결과값 구분	생략시 구버전으로 데이터 출력

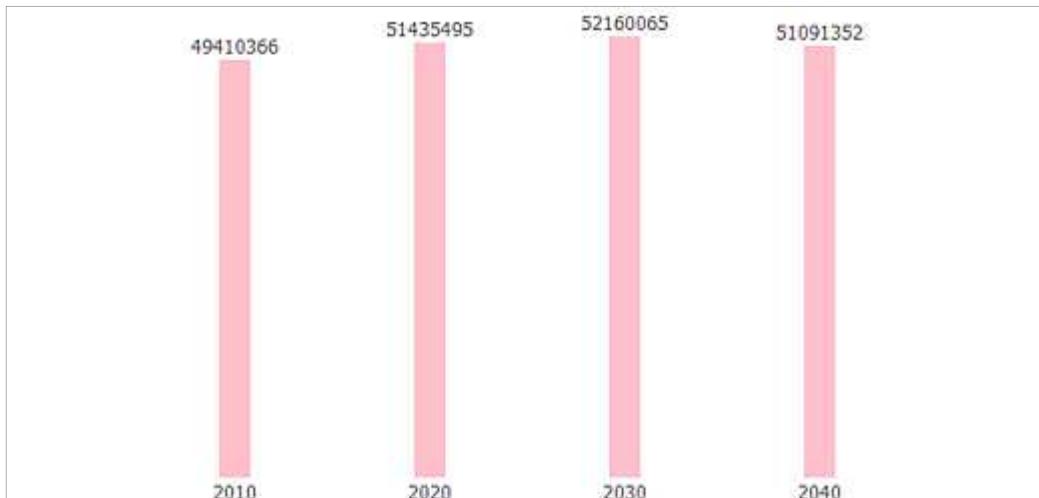
- 통계표선택 방법

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 기준	newEstPrdCnt	String	최근수록시점 개수	최신자료기준 택1)
	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	담당부서			
	Telephone	담당부서 연락처			
	Source			출처	
	Series			단위	
				항목	
				주기	
	C_분류	C_분류 (8개 분류까지 가능)			
	OBS	시점			
		수치자료			
					TIME_PERIOD
					OBS_VALUE

■ 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 # 통계 표명
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)

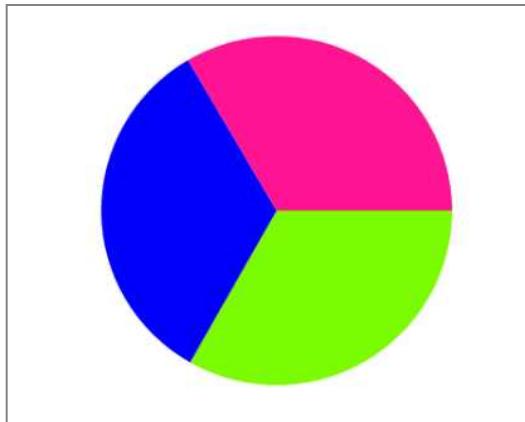
#y축 수치를 안보이게 하는 코드. 필요에 따라 선택하여 사용
plt.gca().axes.yaxis.set_visible(False)

#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/xhtml

```

```

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=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 = ["#7fcfc00", "#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 출력하기위한 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 %>% !(),
  userStatsId = 'openapisample/101/DT_1IN1502/2/1/20191106094026_1',
  # 시계열 조회 키값
  jsonVD = 'Y',
  type = 'StructureSpecific',
  prdSe = 'Y',
  newEstPrdCnt = 3,
  # 최근수록시점 개수
  prdlInterval = 1,
  # 수록시점 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)



### ■ 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, strItmId, 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") {

        strItmId = 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=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 al_C' scope='col'>시점</th>";
    strTable += "<th scope='col'>" + itmNm + "(" + object.documentElement.childNodes[1].childNodes[0].childNod
es[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 <- 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()
```

## 2.3 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 > 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

대용량통계자료 API를 이용하기 위한 요청번호, 출력정보, 셱플데이터를 제공합니다. 출력포맷으로 SDMX, XLS를 제공합니다.  
데이터 출력건수가 40,000개 이상인 경우는 SDMX 제공을 불가하고, XLS로만 가능합니다.  
또한, 데이터 출력건수가 20만개 이상인 경우는 XLS도 제공이 불가하므로 이용건수를 줄이 사용하시기 바랍니다.

상세기능 +
URL 생성 -
개발가이드 다운로드

URL 생성
자료등록

#### 통계표 선택

작성기간	동계정	작성기관 검색	통계조사명	인구총조사	통계조사명 검색
통계표명					
분류					
분류값/항목					
	주기	전체			
	등록한 표면 보기	<input type="checkbox"/>			

검색

Total : 1875 | Page : 1/188

기원명	통계표명	통계표ID	통계조사명	수록기간	선택	통계표조회
등 계정 (101)	인구밀도(인구주택총조사기준)	DT_1B08024	인구총조사	년 1966-2022	<input checked="" type="checkbox"/> 	
등 계정 (101)	행정구역/가구규모/상별 등근-동학연구(12세이상)	DT_1B90001	인구총조사	5년 1995-1995	<input checked="" type="checkbox"/> 	
등 계정 (101)	대도시/동근동학/성/아동교동수단별 등근-동학연구(12세이상)	DT_1B90003	인구총조사	5년 1995-1995	<input checked="" type="checkbox"/> 	
등 계정 (101)	시도/소요시간/이용교통수단별 등근-동학연구(12세이상)	DT_1B90005	인구총조사	5년 1995-1995	<input type="checkbox"/> 	
등 계정 (101)	시도/점유면적/이용교통수단별 등근-동학연구(12세이상)	DT_1B90007	인구총조사	5년 1995-1995	<input type="checkbox"/> 	
등 계정 (101)	시도/성/직업/이용교통수단별 등근-동학연구(12세이상)	DT_1B90008	인구총조사	5년 1995-1995	<input type="checkbox"/> 	
등 계정 (101)	시도/성/산업/이용교통수단별 등근-동학연구(12세이상)	DT_1B90009	인구총조사	5년 1995-1995	<input type="checkbox"/> 	
등 계정 (101)	시도/성/소요시간/학교별 등근-동학연구(12세이상)	DT_1B9000A	인구총조사	5년 1995-1995	<input type="checkbox"/> 	
등 계정 (101)	현거주지/성/국적지별 등근-동학연구(서울,인천,경기,12세이상)	DT_1B9000D	인구총조사	5년 1995-1995	<input type="checkbox"/> 	
등 계정 (101)	현거주지/성/국적지별 등근-동학연구(부산,경남,12세이상)	DT_1B9000E	인구총조사	5년 1995-1995	<input type="checkbox"/> 	

< 1 2 3 4 5 6 7 8 9 10 >

등록 2

KOSIS 공유서비스 개발가이드

102

### 2.3.2.2 등록된 자료

#### ■ 마이페이지 > 등록한 자료 > 대용량통계자료

자료등록의 통계표 등록을 마치고 등록된 자료 템을 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.

활용신청현황
나의 Q&A
나의 활용사례
나의 통계표변동내역
나의 통계표URL변동내역
등록한 자료

통계자료 +
대용량통계자료 -

**통계표 선택**

작성기관	<input type="text"/>	<input type="button" value="작성기관 검색"/>	통계조사명	<input type="text"/>	<input type="button" value="통계조사명 검색"/>
통계표명	<input type="text"/>		통계표ID	<input type="text"/>	
분류	<input type="text"/>		주기	<input type="button" value="전체"/> <input type="button" value="선택"/>	
분류값/항목	<input type="text"/>				

Total : 10 | Page : 1/1

기관명	통계표명	통계표ID	통계조사명	수록기간	신청일자	통계표조회	URL생성
통계청 (101)	경지규모별 농가 및 경지면적	DT_1NJ101	농림어업총조사	5년 2005~2005	2023-09-27 08:19:01	<input type="button" value="조회"/> 2	<input type="button" value="생성"/> 1
통계청 (101)	<input checked="" type="checkbox"/> ❶ 인간개발지수	DT_2KAA01	국제통계연감	년 1990~2021	2022-10-04 16:51:23	<input type="button" value="조회"/>	<input type="button" value="생성"/>

#### ■ 마이페이지 > 등록한 자료 > 대용량통계자료 > 삭제

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

활용신청현황
나의 Q&A
나의 활용사례
나의 통계표변동내역
나의 통계표URL변동내역
등록한 자료

통계자료 +
대용량통계자료 -

**통계표 선택**

작성기관	<input type="text"/>	<input type="button" value="작성기관 검색"/>	통계조사명	<input type="text"/>	<input type="button" value="통계조사명 검색"/>
통계표명	<input type="text"/>		통계표ID	<input type="text"/>	
분류	<input type="text"/>		주기	<input type="button" value="전체"/> <input type="button" value="선택"/>	
분류값/항목	<input type="text"/>				

Total : 10 | Page : 1/1

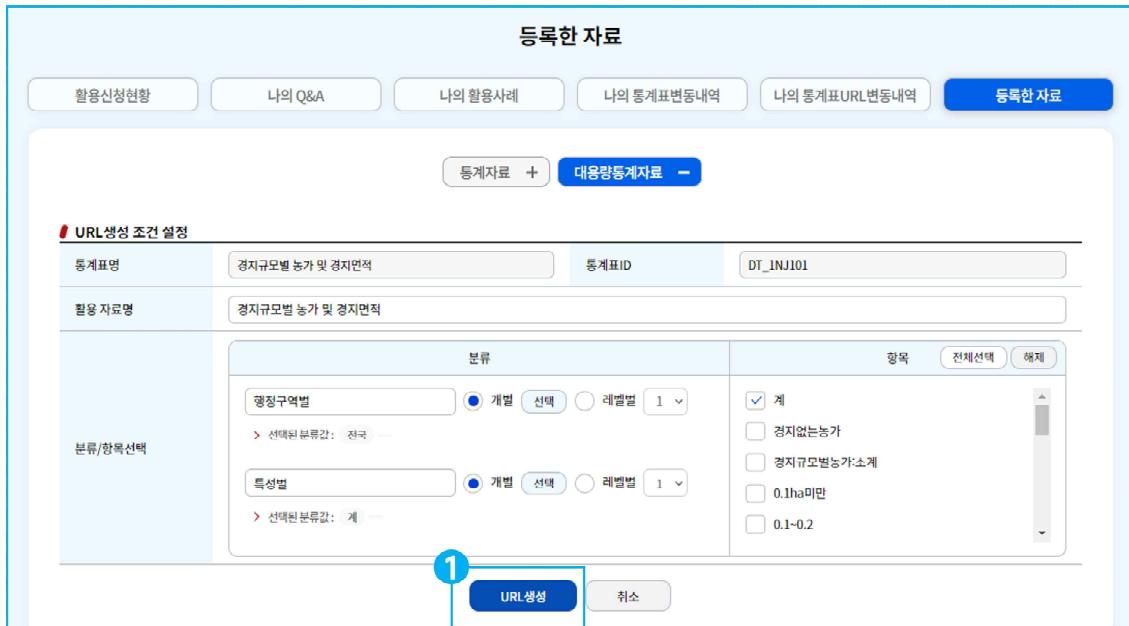
기관명	통계표명	통계표ID	통계조사명	수록기간	신청일자	통계표조회	URL생성
통계청 (101)	경지규모별 농가 및 경지면적	DT_1NJ101	농림어업총조사	5년 2005~2005	2023-09-27 08:19:01	<input type="button" value="조회"/> 1	<input type="button" value="생성"/>
통계청 (101)	<input checked="" type="checkbox"/> ❶ 인간개발지수	DT_2KAA01	국제통계연감	년 1990~2021	2022-10-04 16:51:23	<input type="button" value="조회"/>	<input type="button" value="생성"/>

활용자 생성 URL 목록	<input type="button" value="삭제"/> 2 <input type="button" value="URL보기"/>
인간개발지수	<input type="button" value="삭제"/> 2 <input type="button" value="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.



The screenshot shows the '등록한 자료' (Registered Data) section of the portal. At the top, there are tabs for '활용신청현황', '나의 Q&A', '나의 활용사례', '나의 통계표변동내역', '나의 통계표URL변동내역', and '등록한 자료'. Below these tabs, there are two buttons: '등록자료 +' and '대용량등록자료 -'. The main area is titled 'URL생성 조건 설정' and contains fields for '통계표명' (Statistical Table Name) set to '경지규모별 농가 및 경지면적', '통계표ID' (Statistical Table ID) set to 'DT\_INJ101', and '활용 자료명' (Utilization Data Name) set to '경지규모별 농가 및 경지면적'. On the left, there is a sidebar labeled '분류/항목선택' (Category/Item Selection) which includes dropdown menus for '행정구역별' (Administrative Region Level) and '특성별' (Characteristic Level), both set to '개별' (Individual). To the right, there is a '항목' (Item) section with checkboxes for categories like '계' (Total), '경지없는농가' (Non-farm household), etc. At the bottom, there are three buttons: 'URL생성' (Generate URL) with a blue circle containing the number 1, '취소' (Cancel), and '취소' (Cancel).

**등록한 자료**

활용신청현황      나의 Q&A      나의 활용사례      나의 통계표변동내역      나의 통계표URL변동내역      등록한 자료

통계자료 +      대용량통계자료 -

**URL생성 조건 설정**

통계표명	행정구역(시군구)별, 성별 인구수	통계표ID	DT_1B040A3
------	--------------------	-------	------------

**출력형태 설정**

데이터 포맷	<input checked="" type="radio"/> JSON <input type="radio"/> SDMX
--------	--

**조회기간 설정**

최근시점부터	주기    월    최근 3 개 자료
--------	----------------------

수록시시점간격

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**      **URL보기**      **결과값보기**

행정구역(시군구)별, 성별 인구수      종인구수 \* 전국

[ 재신청 ]

**- 생성된 URL**

```
https://kosis.kr/openapi/statisticsData.do?
method=list&apiKey=NTI0Yjk1MGJ3...&format=json&jsonVD=Y&userStatsId=flagyrf92/101/DT_1B040A3/2/5/20231026134415&prdSe=M&newEstPrdC
nt=3
```

3      [ URL복사 ]

**- 데이터 보기**

행정구역(시군구)별, 성별 인구수(202309)

행정구역(시군구)별	종인구수 (영)
전국	51370001

### 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
		Name
		Contact
		Department
		Telephone
	Source	출처
Codelist	Codelist	Id
		Name
		Description
		Code
		Name
Concepts	ConceptsSche me	Id
		Name
		Description
		concept
		Id
		Name
DataStructures	DataStructure	Description
		Id
		Name
		DataStructureComponents
		Dimension
		Conceptl identity
		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	(시점기준 또는 최신자료기준 택1)
	endPrdDe	String	종료수록시점	
최신자료 기준	newEstPrdCnt	String	최근수록시점 개수	
	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_통계 표ID	
	Name			통계표명	
	Prepared			전송시간	
	Sender	Id		전송기관	
		Name		전송기관명	
		Contact	Department	담당부서	
			Telephone	담당부서 연락처	
Source				출처	
Series	SeriesKey		Value	시리즈키ID	
			value	시리즈키값	
	Obs		ObsDimension	시점	
			ObsValue	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	시작수록시점	선택 (시점기준 또는 최신자료 기준 택1)
	endPrdDe	종료수록시점	
최신자료 기준	newEstPrdCnt	최근수록시점 개수	
	prdInterval	수록시점 간격	
format	String	결과 유형(json, sdmx)	필수
version	String	결과값 구분	생략시 구버전으로 데이터 출력

■ 출력 변수

항목명(영문)	항목설명		
Header	ID		
	Name		
	Prepared		
	Sender	Id	
		Name	
		Contact	
		Department	
		Telephone	
Series	Source		
	UNIT		
	ITEM		
	FREQ		
	분류		
	Obs	ObsDimension	Value
		ObsValue	Value

### 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	1,182,622
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 ①URL복사, 결과값보기 버튼을 클릭하여 페이지 하단에서 결과를 확인 할 수 있습니다.

**통계설명**

통계설명 API를 이용하기 위한 요청번호, 헤더정보, 샘플데이터를 제공합니다. 출력포맷으로 JSON과 XML을 제공합니다.

(상세기능 +) URL 생성 -( ) 개발가이드 다운로드

**UR 생성 조건 설정**

통계조사	인구총조사	작성기간	선택
설명항목	<input type="checkbox"/> 전체	<input checked="" type="checkbox"/> 조사명	<input checked="" type="checkbox"/> 통계종류
	<input checked="" type="checkbox"/> 계속여부	<input checked="" type="checkbox"/> 법적근거	<input checked="" type="checkbox"/> 조사목적
	<input checked="" type="checkbox"/> 조사주기	<input checked="" type="checkbox"/> 조사체계	<input checked="" type="checkbox"/> 공표범위
	<input type="checkbox"/> 공표주기	<input type="checkbox"/> 연락처	<input type="checkbox"/> 통계(활동)분야-실내
	<input type="checkbox"/> 조사대상 범위	<input type="checkbox"/> 조사대상 지역	<input type="checkbox"/> 조사단위 및 조사대상규모
	<input type="checkbox"/> 적용분류	<input type="checkbox"/> 조사항목	<input type="checkbox"/> 공표방법 및 URL
	<input type="checkbox"/> 조사대상기간 및 조사기준시범	<input type="checkbox"/> 공표주기	<input type="checkbox"/> 자료이용자 유의사항
	<input type="checkbox"/> 주요 용어설명	<input type="checkbox"/> 자료 수집방법	<input type="checkbox"/> 조사연혁
	<input type="checkbox"/> 승인번호	<input type="checkbox"/> 송인일자	<input type="checkbox"/> 통계종료

설명자료

1 URL복사 결과값보기

**생성된 URL**

```
https://kosis.kr/openapi/statistics/xptlData.do?method=list&apiKey=NTI0YjU1MGJzWYx0[REDACTED]&statId=1962001&format=xml&jsonMvD=Y&metaNm=statsNm+statsKind+statsContinue+basicStat+writingPurp+statsPeriod+writingSystem+pubExtent
```

**데이터 보기**

항목	설명
조사명	인구총조사
통계종류	지정통계 / 가공통계
개속여부	계속동계
법적근거	인구주택총조사 규칙(기획재정부령 제810호, 2020.10.26. 일부개정)
조사주기	○ 인구규모, 본포 및 구조, 가구, 주택에 관한 특성을 파악하여 각종 정책입안의 기초자료로 활용 ○ 각종 표본조사의 표본률로 활용
조사체계	전수(등록선서스 방식, 1년 주기) : 공공기관 및 사법기관 → 통계청 표본(조사 방식, 5년 주기) : 조사원 → 구·군 ·군구→ 시·도 → 통계청
공표범위	시군구

**결과값보기**

```
<?xml version="1.0" encoding="UTF-8" ?><response><Header><id>1962001</id><Prepared>2023-10-26 14:01:44</Prepared><Sender><id>KOSIS</id><Name>통계청</Name></Sender></Header><Structures><statsNm>인구총조사</statsNm><statsKind>writingPurp</statsKind><statsPeriod>1</statsPeriod><statsContinue>false</statsContinue><basisStat>인구주택총조사 규칙(기획재정부령 제810호, 2020.10.26. 일부개정)</basisStat><writingSystem>한글</writingSystem><Structures></Structures></response>
```

KOSIS 공유서비스 개발가이드

110

### 2.4.3 활용방법

#### 2.4.3.1 JSON

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

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
statId * orgId(기관ID) + tblId(통계표ID) 로도 가능	String	통계조사 ID	필수
metaltm	String	요청 항목	필수 전체 – All 조사명–statsNm 작성유형–statsKind 통계종류–statsEnd 계속여부–statsContinue 법적근거–basisLaw 조사목적–writingPurps 조사기간–examinPd 조사주기–statsPeriod 조사체계–writingSystem 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 공표시기–pubDate 공표방법 및 URL–publicMth 조사대상기간 및 조사기준시점–examinTrgetPd 자료이용시 유의사항 –dataUserNote 주요 용어해설–mainTermExpl 자료 수집방법–dataCollectMth 조사연혁–examinHistory 승인번호–confmNo 승인일자–confmDt
format	String	결과유형(JSON, SDMX)	필수
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
statsNm	조사명	VARCHAR2(4000)
statsKind	작성유형	VARCHAR2(4000)
statsEnd	통계종류	VARCHAR2(4000)
statsContinue	계속여부	VARCHAR2(4000)
basisLaw	법적근거	VARCHAR2(4000)
writingPurps	조사목적	VARCHAR2(4000)
examinPd	조사기간	VARCHAR2(4000)
statsPeriod	조사주기	VARCHAR2(4000)
writingSystem	조사체계	VARCHAR2(4000)
writingTel	연락처	VARCHAR2(8000)
statsField	통계(활용)분야·실태	VARCHAR2(4000)
examinObjrange	조사 대상범위	VARCHAR2(4000)
examinObjArea	조사 대상지역	VARCHAR2(4000)
josaUnit	조사단위 및 조사대상규모	VARCHAR2(4000)
applyGroup	적용분류	VARCHAR2(4000)
josaltm	조사항목	VARCHAR2(4000)
pubPeriod	공표주기	VARCHAR2(4000)
pubExtent	공표범위	VARCHAR2(4000)
pubDate	공표시기	VARCHAR2(4000)
publicMth	공표방법 및 URL	VARCHAR2(4000)
examinTrgetPd	조사대상기간 및 조사기준시점	VARCHAR2(4000)
dataUserNote	자료이용자 유의사항	VARCHAR2(4000)
mainTermExpl	주요 용어해설	VARCHAR2(4000)
dataCollectMth	자료 수집방법	VARCHAR2(4000)
examinHistory	조사연혁	VARCHAR2(4000)
confmNo	승인번호	VARCHAR2(4000)
confmDt	승인일자	VARCHAR2(4000)

■ 예제 소스 결과

항목	설명
조사명	가계등황조사
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 : metalm – 요청 항목
* statId – 통계조사 Id
*****************/
function getSubList(metalm, statId) {

// ajax 통신을 위한 파라메터를 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=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzMzNDgwNmY=&metalm=" + metalm + "&statId=" + statId + "&type=json",
}

```

```

sync : true,
load : function(resObj, a, b) { mapData = resObj; },
error : function ( resObj, e ) { alert(dojo.toJson(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 +"

```

```
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].josalm + "</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'>공표시기</td>"  
nodeInfo+=" <td class='al_C'>" + mapData[0].pubDate + "</td>"  
nodeInfo+=" </tr>"  
nodeInfo+=" <tr>"  
nodeInfo+=" <td class='fst al_C' scope='row'>공표방법및URL</td>"  
nodeInfo+=" <td class='al_C'>" + mapData[0].publicMth + "</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'>자료수집방법</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+=" </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="metalm" ><%out.println(request.getParameter("metalm")) == null ? "" : (request.getParameter("metalm"));%></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}&metalm=${metalm}&statId=${statId}&format=${type}" charEncoding="utf-8"/>

```

## ■ 예제 소스(R)

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

rm(list = ls()) # 모든 변수 초기화

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

```

'조사단위 및 조사대상규모',
'적용분류',
'조사항목',
'공표주기',
'공표범위',
'공표시기',
'공표방법 및 URL',
'조사대상기간 및 조사기준시점',
'자료이용자 유의사항',
'주요 용어해설',
'자료 수집방법',
'조사연혁',
'승인번호',
'승인일자'
)
)

# 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',
    metalm = '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/statisticsExplData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwO
WRjNjgxN2MzNDgwNmY=&format=json&jsonVD=Y&statId=1962009&metaitm>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 'josalm' in v:
    value.append("조사항목")
    value.append(v['josalm'])
    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 = QTableWidget(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.domethod=getList>

■ 입력 변수

항목명(영문)	변수타입	항목설명	비고
apiKey	String	발급된 인증키	필수
statId * orgId(기관ID) +tblId(통계표ID) 로도 가능	String	통계조사 ID	필수
metaltm	String	요청 항목	필수 전체 - All 조사명-statsNm 작성유형-statsKind 통계종류-statsEnd 계속여부-statsContinue 법적근거-basisLaw 조사목적-writingPurps 조사기간-examinPd 조사주기-statsPeriod 조사체계-writingSystem 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 공표시기-pubDate 공표방법 및 URL-publictMth 조사대상기간 및 조사기준시점-examinTrgetPd 자료이용시 유의사항 -dataUserNote 주요 용어해설-mainTermExpl 자료 수집방법-dataCollectMth 조사연혁-examinHistory 승인번호-confmNo 승인일자-confmDt
format	String	결과유형(JSON, SDMX)	필수

■ 출력 변수

항목명(영문)	항목설명	형식
statsNm	조사명	VARCHAR2(4000)
statsKind	작성유형	VARCHAR2(4000)
statsEnd	통계종류	VARCHAR2(4000)
statsContinue	계속여부	VARCHAR2(4000)
basisLaw	법적근거	VARCHAR2(4000)
writingPurps	조사목적	VARCHAR2(4000)
examinPd	조사기간	VARCHAR2(4000)
statsPeriod	조사주기	VARCHAR2(4000)
writingSystem	조사체계	VARCHAR2(4000)
writingTel	연락처	VARCHAR2(8000)
statsField	통계(활용)분야·실태	VARCHAR2(4000)
examinObjrange	조사 대상범위	VARCHAR2(4000)
examinObjArea	조사 대상지역	VARCHAR2(4000)
josaUnit	조사단위 및 조사대상규모	VARCHAR2(4000)
applyGroup	적용분류	VARCHAR2(4000)
josalm	조사항목	VARCHAR2(4000)
pubPeriod	공표주기	VARCHAR2(4000)
pubExtent	공표범위	VARCHAR2(4000)
pubDate	공표시기	VARCHAR2(4000)
publicMth	공표방법 및 URL	VARCHAR2(4000)
examinTrgetPd	조사대상기간 및 조사기준시점	VARCHAR2(4000)
dataUserNote	자료이용자 유의사항	VARCHAR2(4000)
mainTermExpl	주요 용어해설	VARCHAR2(4000)
dataCollectMth	자료 수집방법	VARCHAR2(4000)
examinHistory	조사연혁	VARCHAR2(4000)
confmNo	승인번호	VARCHAR2(4000)
confmDt	승인일자	VARCHAR2(4000)

### ■ 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 mData;

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

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

/*
* 통계목록 리스트 조회 함수
* parameter : metalm - 요청 항목
*/

```

```

* statId - 통계조사 Id
*****
function getSubList(metalm, statId) {

    // ajax 통신을 위한 호출 함수
    $.ajax({
        type: "GET",
        url: "http://mgmk.kosis.kr/openapi_dev/Expt/statisticsExplData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1Yzdh
ZTMwOWRjNjgxN2MzNDgwNmY=&metalm=" + metalm + "&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'>작성 유형</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'>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 == "examinPd") {
    nodeInfo+=" <tr>" 
    nodeInfo+=" <td class='fst al_C' scope='row'>조사기간</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 == "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 == "josalm") {
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 == "pubDate") {
nodeInfo+=" <tr>
nodeInfo+=" <td class='fst al_C' scope='row'>공표시기</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 == "mainTermExpl") {
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>"
}
}

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',
    'statsEnd',
    'statsContinue',
    'basisLaw',
    'writingPurps',
    'examinPd',
    'statsPeriod',
    'writingSystem',
    'writingTel',
    'statsField',
    'examinObjrange',
    'examinObjArea',
    'josaUnit',
    'applyGroup',
    'josalm',
    'pubPeriod',
    'pubExtent',
    'pubDate',
    'publictMth',
    'examinTrgetPd',
    'dataUserNote',
    'mainTermExpl',
    'dataCollectMth',
    'examinHistory',
    'confmNo',
    'confmDt'
  ),
  hname = c(
    '조사명',
    '작성 유형',
    '통계종류',
    '계속여부',
    '법적근거',
    '조사목적',
    '조사주기',
    '조사체계',
    '공표범위',
    '공표주기',
    '연락처',
    '통계(활용)분야·실태',
    '조사 대상범위',
    '조사 대상지역',
    '조사단위 및 조사대상규모',
    '작용분류',
  )
)
```

```

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

# 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?'

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/statisticsExplData.do?method=getList&apiKey=ZjZjOTI3MjRjNmU1YzdhZTMwOWRjNjgxN2MzNDgwNmY=&format=xml&statId=1962009&metaitm=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("작성유형")
    value.append(soup.find('statskind').text)
    data.append(value)
    count += 1

if soup.find('statsEnd') is not None:
    value = []
    value.append("통계종류")
    value.append(soup.find('statsEnd').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('examinPd') is not None:  
    value = []  
    value.append("조사기간")  
    value.append(soup.find('examinPd').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('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('pubDate') is not None:
    value = []
    value.append("공표시기")
    value.append(soup.find('pubDate').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('examintretpd') 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

class MyWindow(QMainWindow):

    def __init__(self):
        super().__init__()
```

```
self.setupUI()

def setupUI(self):
    self.setGeometry(800, 200, 600, 600)
    self.tableWidget = QTableWidget(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’ 등 을 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.

⑤ 통계표 조회 버튼을 클릭하면 해당 자료의 통계표를 볼 수 있습니다.

# 통계표설명

동계자를 이용시 참고할 수 있도록 통계표설명자를 함께 공니다. 출력포맷으로 JSON과 XML을 제공합니다.

상세기능 +

URL 생성 -

[개발가이드 다운로드](#)

## 1 통계표 선택

작성기관	대전광역시	작성기관 검색	통계조사명	주민등록인구통계	통계조사명 검색
통계표명			통계표ID		

검색

기관명	통계표명	통계표ID	통계조사명	수록기간	선택	제표 조회
대전광역시 (206)	구별 동별 세대 및 인구	DT_POPUL0001	주민등록인구통계	분기,년 2009-2023 2/4	<input checked="" type="checkbox"/>	
대전광역시 (206)	구별 연령별(5세계급) 외국인 현황	DT_POPUL0002	주민등록인구통계	분기,년 2009-2023 2/4	<input type="checkbox"/>	
대전광역시 (206)	구별 국적별 외국인 현황	DT_POPUL0003	주민등록인구통계	분기,년 2010-2023 2/4	<input type="checkbox"/>	
대전광역시 (206)	구별 연령별 인구(외국인 제외)	DT_POPUL01	주민등록인구통계	분기,년 2009-2023 2/4	<input type="checkbox"/>	

< 1 >

2

## 2 통계표 설명 분류선택

통계표명정  기관명정  수록정보  분류/항목  주석  단위  출처  가중치  자료갱신일

3

## 3 세부정보 입력

출력포맷	xml	기관코드	206	통계표ID	DT_POPUL0001
URL복사			URL생성		

## 4 생성된 URL

<https://kosis.kr/openapi/statisticsData.do?method=getMeta&apiKey=NTI0YjklMGJ3ZWYxOTA...>

## 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
content	String	헤더 유형(html, json)	선택

- 출력 변수

항목명(영문)	항목설명	형식
TBL_NM	통계표 국문명	VARCHAR2(300)
TBL_NM_ENG	통계표 영문명	VARCHAR2(300)

### 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
content	String	헤더 유형(html, json)	선택

- 출력 변수

항목명(영문)	항목설명	형식
ORG_NM	기관 국문명	VARCHAR2(300)
TORG_NM_ENG	기관 영문명	VARCHAR2(300)

### 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
PRD_SE	수록주기	VARCHAR2(300)
PRD_DE	수록시점	VARCHAR2(8)

### 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
OBJ_ID	분류 ID	VARCHAR2(40)
OBJ_NM	분류 국문명	VARCHAR2(3000)
OBJ_NM_ENG	분류 영문명	VARCHAR2(3000)
ITM_ID	자료코드 ID	VARCHAR2(40)
ITM_NM	자료코드 국문명	VARCHAR2(3000)
ITM_NM_ENG	자료코드 영문명	VARCHAR2(3000)
UP_ITM_ID	상위 자료코드	VARCHAR2(40)
OBJ_ID_SN	분류값 순번	NUMBER(3)
UNIT_ID	단위ID	VARCHAR2(40)
UNIT_NM	단위 국문명	VARCHAR2(1000)
UNIT_ENG_NM	단위 영문명	VARCHAR2(1000)

### 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
CMMT_NM	주석유형	VARCHAR2(40)
CMMT_DC	주석	VARCHAR2(3000)
OBJ_ID	분류 ID	VARCHAR2(40)
OBJ_NM	분류 명	VARCHAR2(3000)
ITM_ID	자료코드 ID	VARCHAR2(40)
ITM_NM	자료코드 명	VARCHAR2(3000)

### 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
UNIT_NM	단위 국문명	VARCHAR2(1000)
UNIT_NM_ENG	단위 영문명	VARCHAR2(1000)

### 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
JOSA_NM	조사명	VARCHAR2(300)
DEPT_NM	통계표 담당부서	VARCHAR2(300)
DEPT_PHONE	단위 담당부서 전화번호	VARCHAR2(100)
STAT_ID	통계조사ID	VARCHAR2(40)

### 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
C1 ~ C8	분류값 ID1 ~ 분류값 ID8	VARCHAR2(40)
C1_NM ~ C8_NM	분류값 명1 ~ 분류값 명8	VARCHAR2(3000)
ITM_ID	항목 ID	VARCHAR2(40)
ITM_NM	항목명	VARCHAR2(3000)
WGT_CO	가중치	NUMBER(23,10)

### 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
content	String	헤더 유형(html, json)	선택

#### ■ 출력 변수

항목명(영문)	항목설명	형식
ORG_NM	기관명	VARCHAR2(300)
TBL_NM	통계표명	VARCHAR2(300)
PRD_SE	수록주기	VARCHAR2(300)
PRD_DE	수록시점	VARCHAR2(8)
SEND_DE	자료갱신일	VARCHAR2(20)

### 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	형식
tblNm	통계표 국문명	VARCHAR2(300)
tblNmEng	통계표 영문명	VARCHAR2(300)

### 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	형식
orgNm	기관 국문명	VARCHAR2(300)
orgNmEng	기관 영문명	VARCHAR2(300)

### 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	형식
prdSe	수록주기	VARCHAR2(300)
PrdDe	수록시점	VARCHAR2(8)

### 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	형식
objId	분류 ID	VARCHAR2(40)
objNm	분류 국문명	VARCHAR2(3000)
objNmEng	분류 영문명	VARCHAR2(3000)
itmId	자료코드 ID	VARCHAR2(40)
itmNm	자료코드 국문명	VARCHAR2(3000)
itmNmEng	자료코드 영문명	VARCHAR2(3000)
uplItemId	상위 자료코드	VARCHAR2(40)
objIdSn	분류값 순번	NUMBER(3)
unitId	단위ID	VARCHAR2(40)
unitNm	단위 국문명	VARCHAR2(1000)
unitEngNm	단위 영문명	VARCHAR2(1000)

### 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	형식
CMMT_NM	주석유형	VARCHAR2(40)
CMMT_DC	주석	VARCHAR2(3000)
OBJ_ID	분류 ID	VARCHAR2(40)
OBJ_NM	분류 명	VARCHAR2(3000)
ITM_ID	자료코드 ID	VARCHAR2(40)
ITM_NM	자료코드 명	VARCHAR2(3000)

### 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	형식
unitNm	단위 국문명	VARCHAR2(1000)
unitNmEng	단위 영문명	VARCHAR2(1000)

### 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	형식
josaNm	조사명	VARCHAR2(300)
deptNm	통계표 담당부서	VARCHAR2(300)
deptPhone	통계표 담당부서 전화번호	VARCHAR2(100)
statId	통계조사 ID	VARCHAR2(40)

### 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	형식
C1 ~ C8	분류값 ID1 ~ 분류값 ID8	VARCHAR2(40)
C1_NM ~ C8_NM	분류값 명1 ~ 분류값 명8	VARCHAR2(3000)
ITM_ID	항목 ID	VARCHAR2(40)
ITM_NM	항목명	VARCHAR2(3000)
WGT_CO	가중치	NUMBER(23,10)

### 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	형식
orgNm	기관명	VARCHAR2(300)
tblNm	통계표명	VARCHAR2(300)
prdSe	수록주기	VARCHAR2(300)
prdDe	수록시점	VARCHAR2(8)
sendDe	자료갱신일	VARCHAR2(20)

## 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’ 등 을 입력한 뒤 ‘URL복사’, ‘결과값보기’ 중 원하는 서비스에 해당하는 버튼을 누르면 결과값을 제공받을 수 있습니다.

**KOSIS통합검색**

국가통계포털(www.kosis.kr)의 통합검색결과를 제공합니다. 출력포맷으로 JSON을 제공합니다.

상세기능 +
URL 생성 -
개발가이드 다운로드

**URL 생성 조건 설정**

검색어 입력	<input type="text" value="인구"/>
정렬	<input type="button" value="정확도 ▾"/>
기관검색(선택사항)	<input type="button" value="기관검색"/>
페이지 번호	<input type="text" value="1"/>
데이터 출력 개수	<input type="text" value="20"/>
결과유형	<input type="button" value="JSON ▾"/>

검색
URL복사

**- 생성된 URL**

```
https://kosis.kr/openapi/statisticsSearch.do?method=getList&apiKey=&format=json&jsonVd=Y&searchNm=인구&orgId=&startCount=1&resultCount=20&sort=RANK
```

**- 결과값보기**

```
[{"ORG_ID": "101", "ORG_NM": "통계청", "TBL_ID": "DT_XN50011", "TBL_NM": "부양비 및 노령화지수 - 남부·동남아시아", "STAT_ID": "A10120210517153124", "STAT_NM": "신남방신북방통계", "VW_CD": "MT_RTITLE", "MT_ATITLE": "(아주지역 통계 > 남부·동남아시아 통계 > 주제별 통계 > 경도/인구", "FULL_PATH_ID": "101_001>101_001_NS>101_001_NS_A>101_001_NS_A_NS_TT_01", "CONTENTS": "국가별 통부양인구비(유소년) 인구(0-14세) 인구(15-64세) 통부양인구비 인구(65세이상) 노령화지수 브루나이 인도네시아 라오스 태국 베트남 싱가포르 인도 필리핀 캄보디아 미얀마 말레이시아", "START_PRD_DE": "1990", "END_PRD_DE": "2100", "ITEM03": "출처: UN 자료: UN https://population.un.org/wpp, World Population Prospects 2022, 2022.7 ◎ 중위(medium) 자료임 ◎ 부양비(Dependency Ratio) 통인구 중에서 생산기능인구(15~64세)에 대한 유소년인구(0~14세)와 고령인구(65세 이상)의 합의 백분비로 인구의 연령구조를 나타내는 지표 ◎ 노령화지수(Index of Ageing) 유소년비(0~14세) 인구 100명에 대한 고령(65세 이상) 인구의
```

✉ 동계서비스기획과

## 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 RANK로 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
ORG_ID	기관코드	VARCHAR2(40)
ORG_NM	기관명	VARCHAR2(300)
TBL_ID	통계표ID	VARCHAR2(40)
TBL_NM	통계표명	VARCHAR2(300)
STAT_ID	조사코드	VARCHAR2(40)
STAT_NM	조사명	VARCHAR2(400)
VW_CD	KOSIS 목록구분	VARCHAR2(40)
MT_ATITLE	KOSIS 통계표 위치	VARCHAR2(4000)
FULL_PATH_ID	통계표 위치	VARCHAR2(400)
CONTENTS	통계표 주요내용	CLOB
STRT_PRD_DE	수록기간 시작일	VARCHAR2(20)
END_PRD_DE	수록기간 종료일	VARCHAR2(20)
ITEM03	통계표 주석	CLOB
REC_TBL_SE	추천통계표 여부	VARCHAR2(10)
TBL_VIEW_URL	통계표 이동URL (KOSIS 목록으로 이동)	VARCHAR2(4000)
LINK_URL	통계표 이동URL (KOSIS 통계표로 이동)	VARCHAR2(4000)
STAT_DB_CNT	검색결과 건수	VARCHAR2(4000)
QUERY	검색어명	VARCHAR2(4000)

## 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

지표 Open API 서비스를 이용하기 위한 JOSN, XML 기반의 데이터형식, 요청변수, 반환되는 코드값을 제공합니다.

상세기능 + URL 생성 - [개발가이드 다운로드](#)

**서비스선택**  
 지표고유번호 설명자료조회  지표명별 설명자료조회  목록별 지표 조회서비스  지표정보 조회서비스

**상세기능선택**  
 고유번호별 개념조회  고유번호별 산정방법 및 출처조회  고유번호별 전체 설명자료조회

**세부정보입력**  
출력포맷  지표ID

**생성된 URL**  
https://kosis.kr/openapi/pkNumberService.do?  
method=getList&apiKey=NTI0Yjk1MGJ3ZWY[REDACTED]&format=xml&jipyold=166&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 없을경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수
content	String	헤더 유형(html, json)	선택

- 출력 변수

항목명(영문)	항목설명	형식
statJipyold	지표ID	NUMBER(20)
statJipyoNm	지표명	VARCHAR2(300)
jipyoExplan	설명자료 제목	VARCHAR2(100)
jipyoExplan1	개념	CLOB

### 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 없을경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
statJipyold	지표ID	NUMBER(20)
statJipyoNm	지표명	VARCHAR2(300)
jipyoExplan	설명자료 제목	VARCHAR2(100)
jipyoExplan1	개념	CLOB

### 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 없을경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
listId	세부목록ID	VARCHAR2(40)
listNm	세부부문명	VARCHAR2(300)
statJipyold	지표ID	NUMBER(20)
statJipyoNm	지표명	VARCHAR2(300)
unit	단위	VARCHAR2(200)
areaTypeName	지역구분명	VARCHAR2(10)
prdSeName	수록주기명	VARCHAR2(10)
strtPrdDe	수록시작시점	VARCHAR2(8)
endPrdDe	수록종료시점	VARCHAR2(8)
rn	수록시점개수	NUMBER(5)
listSn1	목록순서1	NUMBER(22)
listSn2	목록순서2	NUMBER(22)
prdDe	시점	VARCHAR2(100)
repJipyold	대표지표ID	NUMBER(20)
repJipyoNm	대표지표명	VARCHAR2(300)
repJipyoUrl	대표지표URL	VARCHAR2(3000)
explainUrl	지표설명URL	VARCHAR2(3000)

#### 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
statJipyold	지표ID	NUMBER(20)
statJipyoNm	지표명	VARCHAR2(300)
unit	단위	VARCHAR2(200)
areaTypeName	지역구분명	VARCHAR2(10)
prdSeName	수록주기명	VARCHAR2(10)
strtPrdDe	수록시작시점	VARCHAR2(8)
endPrdDe	수록종료시점	VARCHAR2(8)
rn	수록시점개수	NUMBER(5)
prdDe	종료시점+주기명	VARCHAR2(100)

### 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
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
statJipyold	지표ID	NUMBER(20)
statJipyoNm	지표명	VARCHAR2(300)
unit	단위	VARCHAR2(200)
areaTypeName	지역구분명	VARCHAR2(10)
prdSeName	수록주기명	VARCHAR2(10)
strtPrdDe	수록시작시점	VARCHAR2(8)
endPrdDe	수록종료시점	VARCHAR2(8)
rn	수록시점개수	NUMBER(5)
prdDe	종료시점+주기명	VARCHAR2(100)

### 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	조회 시작 시점	선택
	endPrdDe	조회 종료 시점	
최신자료기준	rn	조회 기준 시점	선택
	srvRn	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 없을경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수
content	String	헤더 유형(html, json)	선택

- 출력 변수

항목명(영문)	항목설명	형식
statJipyold	지표ID	NUMBER(20)
statJipyoNm	지표명	VARCHAR2(300)
prdSe	수록주기	VARCHAR2(20)
prdDe	시점	VARCHAR2(8)
itmNm	항목	VARCHAR2(300)
val	통계수치	NUMBER(25,10)

### 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 없을경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수
content	String	헤더 유형(html, json)	선택

■ 출력 변수

항목명(영문)	항목설명	형식
statJipyold	지표ID	NUMBER(20)
statJipyoNm	지표명	VARCHAR2(300)
unit	단위	VARCHAR2(200)
areaTypeName	지역구분명	VARCHAR2(10)
prdSeName	수록주기명	VARCHAR2(10)
strtPrdDe	수록시작시점	VARCHAR2(8)
endPrdDe	수록종료시점	VARCHAR2(8)
rn	수록시점개수	NUMBER(5)
prdDe	종료시점+주기명	VARCHAR2(100)

### 2.7.2.8 JSON,XML (지표명별 상세조회)

- 호출 URL :<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	조회 시작 시점	선택
	endPrdDe	조회 종료 시점	
최신자료기준	rn	조회 기준 시점	선택
	srvRn	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 없을경우에는 자동으로 pageNo=1, numOfRows=10으로 결과 리턴	선택
format	String	결과유형(json, xml)	필수
content	String	헤더 유형(html, json)	선택

#### ■ 출력 변수

항목명(영문)	항목설명	형식
statJipyold	지표ID	NUMBER(20)
statJipyoNm	지표명	VARCHAR2(300)
prdSe	수록주기	VARCHAR2(20)
prdDe	시점	VARCHAR2(8)
itmNm	항목	VARCHAR2(300)
val	통계수치	NUMBER(25,10)

## <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 국의 통계청과 여러 국제기구에 이르기까지 사용범위가 점차 확대되고 있습니다.

SDMX 표준 및 지침, 개발도구(software), 새로운 소식 등은 현재 [sdmx.org](http://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 정의하는 통계표를 예를 들면 통계표에 대한 설명과 통계표의 형태를 파악할 수 있는 통계표의 구성정보와 분류, 분류값, 단위, 항목에 대한 상세정보(코드 및 명칭)를 담고 있음
DATA	DATA 파일은 DSD에서 정의한 구성정보에 주기, 시점에 따른 수치정보를 정의하는 SDMX version 2.1에서는 Generic, StructureSpecific 두 가지 포맷으로 나뉘어짐 <ul style="list-style-type: none"><li>Generic : 데이터를 담는 XML 구성요소가 구조를 정의하는 메세지와 독립적인 형태로 이루어져 있으며 통계구성정보 및 수치정보가 각 Element(요소, 항목)로 구성되어 있어 StructureSpecific에 비해 파일 용량이 큼</li><li>StructureSpecific : 데이터를 담는 XML 구성요소는 구조를 정의하는 메세지에 의존적인 형태로 이루어져 있으며 통계구성정보 및 수치정보는 하나의 Element(요소, 항목)에 Attribute(속성)로 나열되어 있어 Generic에 비해 파일 용량이 작음</li></ul>