IMFData.jl

stephenbnicar

Code

Examples

```
#import Pkg; Pkg.add("IMFData")
using CSV, DataFrames, DataFramesMeta, IMFData

wd = @__DIR__
```

```
"C:\\Directorio_Trabajo\\Julia\\IMFData"
```

General Functions

Get a list of datasets accessible from the API

Note: While the function `get_imf_datasets()` returns a list of all available datasets, currently the module only supports data requests from the International Financial Statistics (IFS) dataset.

```
# Get a list of datasets accessible from the API:
data = IMFData.get_imf_datasets()
names(data)
data = DataFrames.DataFrame(
    dataset_id = data.dataset_id,
    dataset_name = data.dataset_name)

df = DataFrames.DataFrame([[],[]], ["dataset_id", "dataset_name"])
for i in 1:size(data)[1]
    push!(df, (data[i,1],data[i,2]))
end
```

```
df.dataset_id = string.(df.dataset_id)
df.dataset_name = string.(df.dataset_name)
CSV.write(
    wd * "/IMFData.csv",
    delim = ";",
    df)
```

```
"C:\\Directorio_Trabajo\\Julia\\IMFData/IMFData.csv"
```

Get the list of parameters ("dimensions") for a dataset and their values

```
ifs_structure = get_imf_datastructure("IFS")
```

```
Dict{String, Any} with 2 entries:
  "Parameter Names"  => 5×2 DataFrame…
  "Parameter Values" => Dict{Any, Any}("CL_INDICATOR_IFS"=>1679×2 DataFrame…
```

Parameter Names

```
collect(values(ifs_structure))[1]
```

	parameter_id	parameter_name
	String	String
1	CL_UNIT_MULT	Scale
2	CL_FREQ	Frequency
3	CL_AREA_IFS	Geographical Areas
4	CL_INDICATOR_IFS	Indicator
5	CL_TIME_FORMAT	Time format

Parameter Values

```
collect(values(ifs_structure))[2]
```

```
Dict{Any, Any} with 5 entries:
  "CL_INDICATOR_IFS" => 1679×2 DataFrame…
  "CL_UNIT_MULT"     => 31×2 DataFrame…
  "CL_TIME_FORMAT"   => 6×2 DataFrame…
```

```
  "CL_AREA_IFS"       => 278×2 DataFrame…
  "CL_FREQ"           => 6×2 DataFrame…
```

1. “CL_UNIT_MULT”

```
collect(collect(values(ifs_structure))[2])[2]
```

```
Pair{Any, Any}("CL_UNIT_MULT", 31×2 DataFrame
 Row   parameter_value  description
       String           String

   1   0                Units
   2   2                Hundreds
   3   3                Thousands
   4   6                Millions
   5   9                Billions
   6   12               Trillions
   7   N15              Quadrillionths
   8   N14              Hundred Trillionths
   9   N13              Ten Trillionths
  10   N12              Trillionths
  11   N11              Hundred Billionths

  22   1                Tens
  23   4                Ten Thousands
  24   5                Hundred Thousands
  25   7                Ten Millions
  26   8                Hundred Millions
  27   10               Ten Billions
  28   11               Hundred Billions
  29   13               Ten Trillions
  30   14               Hundred Trillions
  31   15               Quadrillions
                              10 rows omitted)
```

2. “CL_FREQ”

```
collect(collect(values(ifs_structure))[2])[5]
```

```
Pair{Any, Any}("CL_FREQ", 6×2 DataFrame
 Row   parameter_value  description
```

```
      String           String

   1  A                Annual
   2  B                Bi-annual
   3  Q                Quarterly
   4  M                Monthly
   5  D                Daily
   6  W                Weekly)
```

3. “CL_AREA_IFS”

```
collect(collect(values(ifs_structure))[2])[4]
```

```
Pair{Any, Any}("CL_AREA_IFS", 278×2 DataFrame
 Row   parameter_value  description
       String           String

   1   AF               Afghanistan
   2   AL               Albania
   3   DZ               Algeria
   4   AD               Andorra
   5   AO               Angola
   6   AI               Anguilla
   7   AG               Antigua and Barbuda
   8   5M               AMF (Arab Monetary Fund)
   9   AR               Argentina
  10   AM               Armenia
  11   AW               Aruba

 269   XR21             Oil-exporting Countries (IMF)
 270   XA69             Other Central America not alloca…
 271   1C_970           Other Holders
 272   1C_SRF_less_EMU  SRF (excluding EA)
 273   1C_SRF_plus_EMU  SRF (including EA)
 274   1C_SRF           SRF Countries
 275   F6               Sub-Saharan Africa
 276   7A               WAEMU (West African Economic and…
 277   A10              Western Hemisphere
 278   W00              All Countries, excluding the IO
                                        257 rows omitted)
```

4. “CL_INDICATOR_IFS”

```
collect(collect(values(ifs_structure))[2])[1]
```

```
Pair{Any, Any}("CL_INDICATOR_IFS", 1679×2 DataFrame
  Row   parameter_value    description
        String             String

    1   NFIAXD_XDC         Acquisitions less Disposals of V…
    2   NFIAXD_SA_XDC      Acquisitions less Disposals of V…
    3   NFIAXD_NSA_XDC     Acquisitions less Disposals of V…
    4   NFIAXD_R_XDC       Acquisitions less Disposals of V…
    5   NFIAXD_R_SA_XDC    Acquisitions less Disposals of V…
    6   NFIAXD_R_NSA_XDC   Acquisitions less Disposals of V…
    7   IAFR_BP6_USD       International Investment Positio…
    8   IADD_BP6_USD       International Investment Positio…
    9   IADE_BP6_USD       International Investment Positio…
   10   IAD_BP6_USD        International Investment Positio…
   11   IADF_BP6_USD       International Investment Positio…

 1670   BTCC_BP6_USD       Balance of Payments, Supplementa…
 1671   RAFA_G_XDR         Total International Reserves, SD…
 1672   RAFA_G_USD         Total International Reserves, US…
 1673   RAFA_MV_USD        Total Reserves, US Dollars (Gold…
 1674   AOTV_PE_NUM        Tourism, Number of Visitors, Per…
 1675   LU_PE_NUM          Unemployment, Persons, Number of
 1676   LU_PE_PC_CP_A_PT   Unemployment, Persons, Percentag…
 1677   LU_PE_PC_PP_PT     Unemployment, Persons, Percentag…
 1678   1C_ALL_INDICATORS  All Indicators
 1679   MFS                Monetary and Financial Statistic…
                                          1658 rows omitted)
```

5. “CL_TIME_FORMAT”

```
collect(collect(values(ifs_structure))[2])[3]
```

```
Pair{Any, Any}("CL_TIME_FORMAT", 6×2 DataFrame
 Row   parameter_value  description
       String           String

   1   P1Y              Annual
   2   P6M              Bi-annual
   3   P3M              Quarterly
```

```
4   P1M             Monthly
5   P7D             Weekly
6   P1D             Daily)
```

Filter Datasets by Key

```julia
datasets = IMFData.get_imf_datasets()
# Use @where macro from DataFramesMeta to filter based on dataset name
#ds_ifs = @where(datasets, occursin.("IFS", :dataset_id))
ds_ifs = DataFramesMeta.@subset(datasets, occursin.("IFS", :dataset_id))
```

	dataset_id	dataset_name
	String	String
1	IFS	International Financial Statistics (IFS)
2	IFS_2017M08	International Financial Statistics (IFS), 2017 M08
3	IFS_2017M09	International Financial Statistics (IFS), 2017 M09
4	IFS_2017M10	International Financial Statistics (IFS), 2017 M10
5	IFS_2017M11	International Financial Statistics (IFS), 2017 M11
6	IFS_2017M12	International Financial Statistics (IFS), 2017 M12
7	IFS_2018M01	International Financial Statistics (IFS), 2018 M01
8	IFS_2018M02	International Financial Statistics (IFS), 2018 M02
9	IFS_2018M03	International Financial Statistics (IFS), 2018 M03
10	IFS_2018M04	International Financial Statistics (IFS), 2018 M04
11	IFS_2018M05	International Financial Statistics (IFS), 2018 M05
12	IFS_2018M06	International Financial Statistics (IFS), 2018 M06
13	IFS_2018M07	International Financial Statistics (IFS), 2018 M07
14	IFS_2018M08	International Financial Statistics (IFS), 2018 M08
15	IFS_2018M09	International Financial Statistics (IFS), 2018 M09
16	IFS_2018M10	International Financial Statistics (IFS), 2018 M10
17	IFS_2018M11	International Financial Statistics (IFS), 2018 M11
18	IFS_2018M12	International Financial Statistics (IFS), 2018 M12
19	IFS_2019M01	International Financial Statistics (IFS), 2019 M01
20	IFS_2019M02	International Financial Statistics (IFS), 2019 M02
21	IFS_2019M03	International Financial Statistics (IFS), 2019 M03
22	IFS_2019M04	International Financial Statistics (IFS), 2019 M04
23	IFS_2019M05	International Financial Statistics (IFS), 2019 M05
24	IFS_2019M06	International Financial Statistics (IFS), 2019 M06
25	IFS_2019M07	International Financial Statistics (IFS), 2019 M07
26	IFS_2019M08	International Financial Statistics (IFS), 2019 M08
27	IFS_2019M09	International Financial Statistics (IFS), 2019 M09
28	IFS_2019M10	International Financial Statistics (IFS), 2019 M10
29	IFS_2019M11	International Financial Statistics (IFS), 2019 M11
30	IFS_2019M12	International Financial Statistics (IFS), 2019 M12
...	...	...

```
#ds_dot = @where(datasets, occursin.("DOT", :dataset_id))
ds_dot = DataFramesMeta.@subset(datasets, occursin.("DOT", :dataset_id))
```

	dataset_id	dataset_name
	String	String
1	DOT	Direction of Trade Statistics (DOTS)
2	DOT_2017Q1	Direction of Trade Statistics (DOTS), 2017 Q1
3	DOT_2017Q2	Direction of Trade Statistics (DOTS), 2017 Q2
4	DOT_2017Q3	Direction of Trade Statistics (DOTS), 2017 Q3
5	DOT_2017Q4	Direction of Trade Statistics (DOTS), 2017 Q4
6	DOT_2018Q1	Direction of Trade Statistics (DOTS), 2018 Q1
7	DOT_2018Q2	Direction of Trade Statistics (DOTS), 2018 Q2
8	DOT_2018Q3	Direction of Trade Statistics (DOTS), 2018 Q3
9	DOT_2018Q4	Direction of Trade Statistics (DOTS), 2018 Q4
10	DOT_2019Q1	Direction of Trade Statistics (DOTS), 2019 Q1
11	DOT_2019Q2	Direction of Trade Statistics (DOTS), 2019 Q2
12	DOT_2019Q3	Direction of Trade Statistics (DOTS), 2019 Q3
13	DOT_2019Q4	Direction of Trade Statistics (DOTS), 2019 Q4
14	DOT_2020Q1	Direction of Trade Statistics (DOTS), 2020 Q1
15	DOT_2020Q2	Direction of Trade Statistics (DOTS), 2020 Q2
16	DOT_2020Q3	Direction of Trade Statistics (DOTS), 2020 Q3
17	DOT_2020Q4	Direction of Trade Statistics (DOTS), 2020 Q4
18	DOT_2021Q1	Direction of Trade Statistics (DOTS), 2021 Q1
19	DOT_2021Q2	Direction of Trade Statistics (DOTS), 2021 Q2
20	DOT_2021Q3	Direction of Trade Statistics (DOTS), 2021 Q3
21	DOT_2021Q4	Direction of Trade Statistics (DOTS), 2021 Q4
22	DOT_2022Q1	Direction of Trade Statistics (DOTS), 2022 Q1
23	DOT_2022Q2	Direction of Trade Statistics (DOTS), 2022 Q2
24	DOT_2022Q3	Direction of Trade Statistics (DOTS), 2022 Q3
25	DOT_2022Q4	Direction of Trade Statistics (DOTS), 2022 Q4
26	DOT_2023Q1	Direction of Trade Statistics (DOTS), 2023 Q1

Retrieve data from the IFS dataset

With one request:

```
us_gdp = get_ifs_data("US", "NGDP_SA_XDC", "Q", 1900, 2100)
```

```
IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA_XDC
Description:
```

```
Frequency: Q
Time Period: 1950 to 2023
Data: 293 x 2 DataFrame
```

```
us_gdp.series
```

	date	value
	Date	Float64
1	1950-03-01	70207.0
2	1950-06-01	72595.8
3	1950-09-01	77038.3
4	1950-12-01	79986.3
5	1951-03-01	84000.0
6	1951-06-01	86022.5
7	1951-09-01	87846.3
8	1951-12-01	89044.5
9	1952-03-01	89955.0
10	1952-06-01	90257.5
11	1952-09-01	91925.3
12	1952-12-01	95203.0
13	1953-03-01	96995.0
14	1953-06-01	97937.3
15	1953-09-01	97792.8
16	1953-12-01	96492.5
17	1954-03-01	96336.3
18	1954-06-01	96530.3
19	1954-09-01	97749.0
20	1954-12-01	99933.5
21	1955-03-01	1.03268e5
22	1955-06-01	105383.0
23	1955-09-01	1.07555e5
24	1955-12-01	109273.0
25	1956-03-01	1.09936e5
26	1956-06-01	1.11502e5
27	1956-09-01	1.12798e5
28	1956-12-01	1.15116e5
29	1957-03-01	1.17445e5
30	1957-06-01	1.18006e5
...	...	...

With multiple requests:

```
countries = ["US","CA"]
indicators = ["NGDP_SA_XDC"]
us_ca_gdp = get_ifs_data(countries, indicators, "Q", 1900, 2100)
```

```
2-element Vector{IMFSeries}:
 IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1950 to 2023
Data: 293 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: CA
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1961 to 2023
Data: 249 x 2 DataFrame
```

```
us_ca_gdp[1].series
```

	date	value
	Date	Float64
1	1950-03-01	70207.0
2	1950-06-01	72595.8
3	1950-09-01	77038.3
4	1950-12-01	79986.3
5	1951-03-01	84000.0
6	1951-06-01	86022.5
7	1951-09-01	87846.3
8	1951-12-01	89044.5
9	1952-03-01	89955.0
10	1952-06-01	90257.5
11	1952-09-01	91925.3
12	1952-12-01	95203.0
13	1953-03-01	96995.0
14	1953-06-01	97937.3
15	1953-09-01	97792.8
16	1953-12-01	96492.5
17	1954-03-01	96336.3
18	1954-06-01	96530.3
19	1954-09-01	97749.0
20	1954-12-01	99933.5
21	1955-03-01	1.03268e5
22	1955-06-01	105383.0
23	1955-09-01	1.07555e5
24	1955-12-01	109273.0
25	1956-03-01	1.09936e5
26	1956-06-01	1.11502e5
27	1956-09-01	1.12798e5
28	1956-12-01	1.15116e5
29	1957-03-01	1.17445e5
30	1957-06-01	1.18006e5
...	...	...

```
us_ca_gdp[2].series
```

	date	value
	Date	Float64
1	1961-03-01	10168.0
2	1961-06-01	10392.0
3	1961-09-01	10670.0
4	1961-12-01	10869.0
5	1962-03-01	11174.0
6	1962-06-01	11300.0
7	1962-09-01	11536.0
8	1962-12-01	11806.0
9	1963-03-01	11917.0
10	1963-06-01	12161.0
11	1963-09-01	12361.0
12	1963-12-01	12863.0
13	1964-03-01	13163.0
14	1964-06-01	13379.0
15	1964-09-01	13669.0
16	1964-12-01	13856.0
17	1965-03-01	14341.0
18	1965-06-01	14691.0
19	1965-09-01	15043.0
20	1965-12-01	15583.0
21	1966-03-01	16107.0
22	1966-06-01	16648.0
23	1966-09-01	16866.0
24	1966-12-01	17189.0
25	1967-03-01	17396.0
26	1967-06-01	17960.0
27	1967-09-01	18121.0
28	1967-12-01	18435.0
29	1968-03-01	18852.0
30	1968-06-01	19440.0
...	...	...

```
# Pasar a DataFrame
df = []
dftemp = []
i = 1
for i in 1:size(countries)[1]
    dftemp = us_ca_gdp[i].series;
    dftemp[:,:country] .= countries[i];
    dftemp[:,:indicator] .= indicators[1];
```

```
    df = push!(df,dftemp)
end
df = vcat(df...)
```

	date	value	country	indicator
	Date	Float64	String	String
1	1950-03-01	70207.0	US	NGDP_SA_XDC
2	1950-06-01	72595.8	US	NGDP_SA_XDC
3	1950-09-01	77038.3	US	NGDP_SA_XDC
4	1950-12-01	79986.3	US	NGDP_SA_XDC
5	1951-03-01	84000.0	US	NGDP_SA_XDC
6	1951-06-01	86022.5	US	NGDP_SA_XDC
7	1951-09-01	87846.3	US	NGDP_SA_XDC
8	1951-12-01	89044.5	US	NGDP_SA_XDC
9	1952-03-01	89955.0	US	NGDP_SA_XDC
10	1952-06-01	90257.5	US	NGDP_SA_XDC
11	1952-09-01	91925.3	US	NGDP_SA_XDC
12	1952-12-01	95203.0	US	NGDP_SA_XDC
13	1953-03-01	96995.0	US	NGDP_SA_XDC
14	1953-06-01	97937.3	US	NGDP_SA_XDC
15	1953-09-01	97792.8	US	NGDP_SA_XDC
16	1953-12-01	96492.5	US	NGDP_SA_XDC
17	1954-03-01	96336.3	US	NGDP_SA_XDC
18	1954-06-01	96530.3	US	NGDP_SA_XDC
19	1954-09-01	97749.0	US	NGDP_SA_XDC
20	1954-12-01	99933.5	US	NGDP_SA_XDC
21	1955-03-01	1.03268e5	US	NGDP_SA_XDC
22	1955-06-01	105383.0	US	NGDP_SA_XDC
23	1955-09-01	1.07555e5	US	NGDP_SA_XDC
24	1955-12-01	109273.0	US	NGDP_SA_XDC
25	1956-03-01	1.09936e5	US	NGDP_SA_XDC
26	1956-06-01	1.11502e5	US	NGDP_SA_XDC
27	1956-09-01	1.12798e5	US	NGDP_SA_XDC
28	1956-12-01	1.15116e5	US	NGDP_SA_XDC
29	1957-03-01	1.17445e5	US	NGDP_SA_XDC
30	1957-06-01	1.18006e5	US	NGDP_SA_XDC
...	...	...	...	...

Examples

IFS indicators

Indicators: "CL_INDICATOR_IFS"

```
ifs_structure  = get_imf_datastructure("IFS")
ifs_indicators = ifs_structure["Parameter Values"]["CL_INDICATOR_IFS"]
```

	parameter_value	description
	String	String
1	NFIAXD_XDC	Acquisitions less Disposals of Valuables, Nomin
2	NFIAXD_SA_XDC	Acquisitions less Disposals of Valuables, Nominal, Seasona
3	NFIAXD_NSA_XDC	Acquisitions less Disposals of Valuables, Nominal, Un
4	NFIAXD_R_XDC	Acquisitions less Disposals of Valuables, Real
5	NFIAXD_R_SA_XDC	Acquisitions less Disposals of Valuables, Real, Seasonall
6	NFIAXD_R_NSA_XDC	Acquisitions less Disposals of Valuables, Real, Unac
7	IAFR_BP6_USD	International Investment Positions, Net acquisition of financial
8	IADD_BP6_USD	International Investment Positions, Assets, Direct investm
9	IADE_BP6_USD	International Investment Positions, Assets, Direct investment, Equi
10	IAD_BP6_USD	International Investment Positions, Assets, Direc
11	IADF_BP6_USD	International Investment Positions, Financial derivatives (other than res
12	IAOE_BP6_USD	International Investment Positions, Assets, Other invest
13	IAO_BP6_USD	International Investment Positions, Assets, Other
14	IAOD_BP6_USD	International Investment Positions, Other investment: Net acquisition o
15	IAPD_BP6_USD	International Investment Positions, Assets, Portfolio invest
16	IAPE_BP6_USD	International Investment Positions, Assets, Portfolio investment, Equ
17	IAP_BP6_USD	International Investment Positions, Assets, Portfol
18	IARFR_BP6_USD	International Investment Positions, Reserve Assets, Net
19	BK_DB_BP6_USD	Balance of Payments, Capital Account, Tota
20	FASAGEA_EUR	Central Bank Survey, Claims on Central Government (Eu
21	FASAFEA_EUR	Central Bank Survey, Claims on Nonresidents (Euro
22	FASABEA_EUR	Central Bank Survey, Claims on Other Depository Corporation
23	FASAOFEA_EUR	Central Bank Survey, Claims on Other Financial Corporation
24	FASAOEA_EUR	Central Bank Survey, Claims on Other Sectors (Euro
25	FASAOPEA_EUR	Central Bank Survey, Claims on Private Sector (Euro
26	FASAONEA_EUR	Central Bank Survey, Claims on Public Nonfinancial Corporatio
27	FASAOSEA_EUR	Central Bank Survey, Claims on State and Local Government
28	FASMBCEA_EUR	Central Bank Survey, Currency in Circulation (Euro
29	FASDOEA_EUR	Central Bank Survey, Demand deposits of other sectors (E
30	FASDSSEA_EUR	Central Bank Survey, Deposits and Securities Other than Shares excl. from
...	...	...

```
ifs_indicators.parameter_value
```

```
1679-element Vector{String}:
 "NFIAXD_XDC"
 "NFIAXD_SA_XDC"
 "NFIAXD_NSA_XDC"
 "NFIAXD_R_XDC"
 "NFIAXD_R_SA_XDC"
 "NFIAXD_R_NSA_XDC"
 "IAFR_BP6_USD"
 "IADD_BP6_USD"
 "IADE_BP6_USD"
 "IAD_BP6_USD"
 "IADF_BP6_USD"
 "IAOE_BP6_USD"
 "IAO_BP6_USD"

 "BTRUE_BP6_USD"
 "BXISXF_BP6_USD"
 "BTCC_BP6_USD"
 "RAFA_G_XDR"
 "RAFA_G_USD"
 "RAFA_MV_USD"
 "AOTV_PE_NUM"
 "LU_PE_NUM"
 "LU_PE_PC_CP_A_PT"
 "LU_PE_PC_PP_PT"
 "1C_ALL_INDICATORS"
 "MFS"
```

```
ifs_indicators.description
```

```
1679-element Vector{String}:
 "Acquisitions less Disposals of Valuables, Nominal, Domestic Currency"
 "Acquisitions less Disposals of "   27 bytes   "lly Adjusted, Domestic Currency"
 "Acquisitions less Disposals of "   18 bytes   ", Unadjusted, Domestic Currency"
 "Acquisitions less Disposals of Valuables, Real, Domestic Currency"
 "Acquisitions less Disposals of "   24 bytes   "lly Adjusted, Domestic Currency"
 "Acquisitions less Disposals of Valuables, Real, Unadjusted, Domestic Currency"
 "International Investment Positi"   40 bytes   " (with Fund Record), US Dollars"
```

```
 "International Investment Positi"   29 bytes   "t, Debt instruments, US Dollars"
 "International Investment Positi"   47 bytes   "stment fund shares , US Dollars"
 "International Investment Positions, Assets, Direct investment, US Dollars"
 "International Investment Positi"   57 bytes   "oyee stock options , US Dollars"
 "International Investment Positi"   26 bytes   "ent,  Other equity , US Dollars"
 "International Investment Positions, Assets, Other investment , US Dollars"

 "Balance of Payments, Supplement"   18 bytes   "s and related items, US Dollars"
 "Balance of Payments, Supplement"   50 bytes   "eptional financing), US Dollars"
 "Balance of Payments, Supplement"   23 bytes   "t + Capital Account, US Dollars"
 "Total International Reserves, SDRs (gold at 35 SDRs per ounce)"
 "Total International Reserves, US Dollars (gold at 35 SDRs per ounce)"
 "Total Reserves, US Dollars (Gold at Market Price)"
 "Tourism, Number of Visitors, Persons, Number of"
 "Unemployment, Persons, Number of"
 "Unemployment, Persons, Percenta"   23 bytes   "g period previous year, Percent"
 "Unemployment, Persons, Percentage change, previous period, Percent"
 "All Indicators"
 "Monetary and Financial Statistics (MFS)"
```

```
CSV.write(
    wd * "/IFS_Indicators.csv",
    delim = ";",
    ifs_indicators)
```

```
"C:\\Directorio_Trabajo\\Julia\\IMFData/IFS_Indicators.csv"
```

Units: "CL_UNIT_MULT"

```
ifs_units = ifs_structure["Parameter Values"]["CL_UNIT_MULT"]
```

	parameter_value	description
	String	String
1	0	Units
2	2	Hundreds
3	3	Thousands
4	6	Millions
5	9	Billions
6	12	Trillions
7	N15	Quadrillionths
8	N14	Hundred Trillionths
9	N13	Ten Trillionths
10	N12	Trillionths
11	N11	Hundred Billionths
12	N10	Ten Billionths
13	N9	Billionths
14	N8	Hundred Millionths
15	N7	Ten Millionths
16	N6	Millionths
17	N5	Hundred Thousandths
18	N4	Ten Thousandths
19	N3	Thousandths
20	N2	Hundredths
21	N1	Tenths
22	1	Tens
23	4	Ten Thousands
24	5	Hundred Thousands
25	7	Ten Millions
26	8	Hundred Millions
27	10	Ten Billions
28	11	Hundred Billions
29	13	Ten Trillions
30	14	Hundred Trillions
...	...	...

```
ifs_units.description
```

```
31-element Vector{String}:
 "Units"
 "Hundreds"
 "Thousands"
 "Millions"
 "Billions"
```

```
 "Trillions"
 "Quadrillionths"
 "Hundred Trillionths"
 "Ten Trillionths"
 "Trillionths"
 "Hundred Billionths"
 "Ten Billionths"
 "Billionths"

 "Hundredths"
 "Tenths"
 "Tens"
 "Ten Thousands"
 "Hundred Thousands"
 "Ten Millions"
 "Hundred Millions"
 "Ten Billions"
 "Hundred Billions"
 "Ten Trillions"
 "Hundred Trillions"
 "Quadrillions"
```

Time Format: "CL_TIME_FORMAT"

```
ifs_time = ifs_structure["Parameter Values"]["CL_TIME_FORMAT"]
```

	parameter_value	description
	String	String
1	P1Y	Annual
2	P6M	Bi-annual
3	P3M	Quarterly
4	P1M	Monthly
5	P7D	Weekly
6	P1D	Daily

```
ifs_time.description
```

```
6-element Vector{String}:
 "Annual"
 "Bi-annual"
 "Quarterly"
```

```
"Monthly"
"Weekly"
"Daily"
```

Countries: "CL_AREA_IFS"

```
ifs_countries = ifs_structure["Parameter Values"]["CL_AREA_IFS"]
```

	parameter_value	description
	String	String
1	AF	Afghanistan
2	AL	Albania
3	DZ	Algeria
4	AD	Andorra
5	AO	Angola
6	AI	Anguilla
7	AG	Antigua and Barbuda
8	5M	AMF (Arab Monetary Fund)
9	AR	Argentina
10	AM	Armenia
11	AW	Aruba
12	AU	Australia
13	AT	Austria
14	AZ	Azerbaijan
15	BS	Bahamas
16	BH	Bahrain
17	BD	Bangladesh
18	5W	BEAC (Banque des Etats de l'Afrique Centrale)
19	5B	BIS (Bank for International Settlements)
20	BB	Barbados
21	BY	Belarus
22	BE	Belgium
23	R1	Belgo-Luxembourg Economic Union
24	BZ	Belize
25	BJ	Benin
26	BM	Bermuda
27	BT	Bhutan
28	BO	Bolivia
29	BA	Bosnia and Herzegovina
30	BW	Botswana
...	...	...

```
ifs_countries.description
```

```
278-element Vector{String}:
 "Afghanistan"
 "Albania"
 "Algeria"
 "Andorra"
 "Angola"
 "Anguilla"
 "Antigua and Barbuda"
 "AMF (Arab Monetary Fund)"
 "Argentina"
 "Armenia"
 "Aruba"
 "Australia"
 "Austria"

 "Non-oil Developing Countries (IMF)"
 "Non SRF countries"
 "Oil-exporting Countries (IMF)"
 "Other Central America not allocated (IMF)"
 "Other Holders"
 "SRF (excluding EA)"
 "SRF (including EA)"
 "SRF Countries"
 "Sub-Saharan Africa"
 "WAEMU (West African Economic and Monetary Union)"
 "Western Hemisphere"
 "All Countries, excluding the IO"
```

frequency: "CL_FREQ"

```
ifs_frequency = ifs_structure["Parameter Values"]["CL_FREQ"]
```

	parameter_value	description
	String	String
1	A	Annual
2	B	Bi-annual
3	Q	Quarterly
4	M	Monthly
5	D	Daily
6	W	Weekly

```
ifs_frequency.description
```

```
6-element Vector{String}:
 "Annual"
 "Bi-annual"
 "Quarterly"
 "Monthly"
 "Daily"
 "Weekly"
```

One Country, One Indicator

```
indicators = "NGDP_SA_XDC"
countries  = "US"
us_gdp = get_ifs_data(countries, indicators, "Q", 1900, 2100)
```

```
IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1950 to 2023
Data: 293 x 2 DataFrame
```

```
us_gdp.series
```

	date	value
	Date	Float64
1	1950-03-01	70207.0
2	1950-06-01	72595.8
3	1950-09-01	77038.3
4	1950-12-01	79986.3
5	1951-03-01	84000.0
6	1951-06-01	86022.5
7	1951-09-01	87846.3
8	1951-12-01	89044.5
9	1952-03-01	89955.0
10	1952-06-01	90257.5
11	1952-09-01	91925.3
12	1952-12-01	95203.0
13	1953-03-01	96995.0
14	1953-06-01	97937.3
15	1953-09-01	97792.8
16	1953-12-01	96492.5
17	1954-03-01	96336.3
18	1954-06-01	96530.3
19	1954-09-01	97749.0
20	1954-12-01	99933.5
21	1955-03-01	1.03268e5
22	1955-06-01	105383.0
23	1955-09-01	1.07555e5
24	1955-12-01	109273.0
25	1956-03-01	1.09936e5
26	1956-06-01	1.11502e5
27	1956-09-01	1.12798e5
28	1956-12-01	1.15116e5
29	1957-03-01	1.17445e5
30	1957-06-01	1.18006e5
...	...	...

```
us_gdp.indicator
```

```
"NGDP_SA_XDC"
```

```
us_gdp.frequency
```

```
"Q"
```

Availability of Data

```
indic = "NGDP_SA_XDC"
area  = "US"
data_available = get_ifs_data(area, indic, "Q", 1900, 2100)
```

```
IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1950 to 2023
Data: 293 x 2 DataFrame
```

```
data_not_available = get_ifs_data(area, indic, "M", 1900, 2100)
```

```
IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period
```

```
data_not_defined = get_ifs_data(area, "NGDP_SA", "Q", 1900, 2100)
```

```
IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period
```

Filter Indicators

```
ifs_indicators = ifs_structure["Parameter Values"]["CL_INDICATOR_IFS"]
#gdp_indicators = @where(
gdp_indicators = @subset(
    ifs_indicators,
    occursin.("Gross Domestic Product", :description),
    occursin.("Domestic Currency", :description))
```

	parameter_value	description
	String	String
1	NGDP_XDC	Gross Domestic Product, Nominal, Domestic Currency
2	NGDP_SA_XDC	Gross Domestic Product, Nominal, Seasonally Adjusted, Domestic Currency
3	NGDP_NSA_XDC	Gross Domestic Product, Nominal, Unadjusted, Domestic Currency
4	NGDP_R_XDC	Gross Domestic Product, Real, Domestic Currency
5	NGDP_R_SA_XDC	Gross Domestic Product, Real, Seasonally Adjusted, Domestic Currency
6	NGDP_R_NSA_XDC	Gross Domestic Product, Real, Unadjusted, Domestic Currency

```
indicators = gdp_indicators[2,1]
countries  = "US"
us_gdp = get_ifs_data(countries, indicators, "Q", 1900, 2100)
```

```
IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1950 to 2023
Data: 293 x 2 DataFrame
```

```
us_gdp.series
```

	date	value
	Date	Float64
1	1950-03-01	70207.0
2	1950-06-01	72595.8
3	1950-09-01	77038.3
4	1950-12-01	79986.3
5	1951-03-01	84000.0
6	1951-06-01	86022.5
7	1951-09-01	87846.3
8	1951-12-01	89044.5
9	1952-03-01	89955.0
10	1952-06-01	90257.5
11	1952-09-01	91925.3
12	1952-12-01	95203.0
13	1953-03-01	96995.0
14	1953-06-01	97937.3
15	1953-09-01	97792.8
16	1953-12-01	96492.5
17	1954-03-01	96336.3
18	1954-06-01	96530.3
19	1954-09-01	97749.0
20	1954-12-01	99933.5
21	1955-03-01	1.03268e5
22	1955-06-01	105383.0
23	1955-09-01	1.07555e5
24	1955-12-01	109273.0
25	1956-03-01	1.09936e5
26	1956-06-01	1.11502e5
27	1956-09-01	1.12798e5
28	1956-12-01	1.15116e5
29	1957-03-01	1.17445e5
30	1957-06-01	1.18006e5
...	...	...

Many Countries, One Indicator

```
function get_df(x)
    try
        df = countries_indicators[x].series
        df.country .= countries_indicators[x].area
        df.indicator .= countries_indicators[x].indicator
```

```
        df.frequency .= countries_indicators[x].frequency
        return df
    catch
        return DataFrames.DataFrame()
    end
end
```

```
get_df (generic function with 1 method)
```

```
indicators = "NGDP_SA_XDC"
countries  = ["US","CA","MX"]
countries_indicators = get_ifs_data(countries, indicators, "Q", 1900, 2100)
```

```
3-element Vector{IMFSeries}:
 IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1950 to 2023
Data: 293 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: CA
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1961 to 2023
Data: 249 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: MX
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1993 to 2023
Data: 121 x 2 DataFrame
```

```
size(countries_indicators)[1]
```

3

```
get_df(3)
```

	date	value	country	indicator	frequency
	Date	Float64	String	String	String
1	1993-03-01	3.76565e5	MX	NGDP_SA_XDC	Q
2	1993-06-01	3.82836e5	MX	NGDP_SA_XDC	Q
3	1993-09-01	3.94873e5	MX	NGDP_SA_XDC	Q
4	1993-12-01	4.06532e5	MX	NGDP_SA_XDC	Q
5	1994-03-01	4.20049e5	MX	NGDP_SA_XDC	Q
6	1994-06-01	4.36909e5	MX	NGDP_SA_XDC	Q
7	1994-09-01	4.52857e5	MX	NGDP_SA_XDC	Q
8	1994-12-01	4.72066e5	MX	NGDP_SA_XDC	Q
9	1995-03-01	5.02764e5	MX	NGDP_SA_XDC	Q
10	1995-06-01	5.54636e5	MX	NGDP_SA_XDC	Q
11	1995-09-01	600787.0	MX	NGDP_SA_XDC	Q
12	1995-12-01	6.53868e5	MX	NGDP_SA_XDC	Q
13	1996-03-01	7.09547e5	MX	NGDP_SA_XDC	Q
14	1996-06-01	749738.0	MX	NGDP_SA_XDC	Q
15	1996-09-01	7.99082e5	MX	NGDP_SA_XDC	Q
16	1996-12-01	8.61168e5	MX	NGDP_SA_XDC	Q
17	1997-03-01	9.08696e5	MX	NGDP_SA_XDC	Q
18	1997-06-01	9.58654e5	MX	NGDP_SA_XDC	Q
19	1997-09-01	1.01563e6	MX	NGDP_SA_XDC	Q
20	1997-12-01	1.07978e6	MX	NGDP_SA_XDC	Q
21	1998-03-01	1.12847e6	MX	NGDP_SA_XDC	Q
22	1998-06-01	1.1771e6	MX	NGDP_SA_XDC	Q
23	1998-09-01	1.22955e6	MX	NGDP_SA_XDC	Q
24	1998-12-01	1.27708e6	MX	NGDP_SA_XDC	Q
25	1999-03-01	1.34268e6	MX	NGDP_SA_XDC	Q
26	1999-06-01	1.40737e6	MX	NGDP_SA_XDC	Q
27	1999-09-01	1.46883e6	MX	NGDP_SA_XDC	Q
28	1999-12-01	1.5222e6	MX	NGDP_SA_XDC	Q
29	2000-03-01	1.59901e6	MX	NGDP_SA_XDC	Q
30	2000-06-01	1.6559e6	MX	NGDP_SA_XDC	Q
...	...	...	...	...	...

```
ndf = size(countries_indicators)[1]
df = map(_ -> DataFrames.DataFrame(), 1:ndf)
vcat([df[i] = get_df(i) for i in 1:ndf]...)
```

	date	value	country	indicator	frequency
	Date	Float64	String	String	String
1	1950-03-01	70207.0	US	NGDP_SA_XDC	Q
2	1950-06-01	72595.8	US	NGDP_SA_XDC	Q
3	1950-09-01	77038.3	US	NGDP_SA_XDC	Q
4	1950-12-01	79986.3	US	NGDP_SA_XDC	Q
5	1951-03-01	84000.0	US	NGDP_SA_XDC	Q
6	1951-06-01	86022.5	US	NGDP_SA_XDC	Q
7	1951-09-01	87846.3	US	NGDP_SA_XDC	Q
8	1951-12-01	89044.5	US	NGDP_SA_XDC	Q
9	1952-03-01	89955.0	US	NGDP_SA_XDC	Q
10	1952-06-01	90257.5	US	NGDP_SA_XDC	Q
11	1952-09-01	91925.3	US	NGDP_SA_XDC	Q
12	1952-12-01	95203.0	US	NGDP_SA_XDC	Q
13	1953-03-01	96995.0	US	NGDP_SA_XDC	Q
14	1953-06-01	97937.3	US	NGDP_SA_XDC	Q
15	1953-09-01	97792.8	US	NGDP_SA_XDC	Q
16	1953-12-01	96492.5	US	NGDP_SA_XDC	Q
17	1954-03-01	96336.3	US	NGDP_SA_XDC	Q
18	1954-06-01	96530.3	US	NGDP_SA_XDC	Q
19	1954-09-01	97749.0	US	NGDP_SA_XDC	Q
20	1954-12-01	99933.5	US	NGDP_SA_XDC	Q
21	1955-03-01	1.03268e5	US	NGDP_SA_XDC	Q
22	1955-06-01	105383.0	US	NGDP_SA_XDC	Q
23	1955-09-01	1.07555e5	US	NGDP_SA_XDC	Q
24	1955-12-01	109273.0	US	NGDP_SA_XDC	Q
25	1956-03-01	1.09936e5	US	NGDP_SA_XDC	Q
26	1956-06-01	1.11502e5	US	NGDP_SA_XDC	Q
27	1956-09-01	1.12798e5	US	NGDP_SA_XDC	Q
28	1956-12-01	1.15116e5	US	NGDP_SA_XDC	Q
29	1957-03-01	1.17445e5	US	NGDP_SA_XDC	Q
30	1957-06-01	1.18006e5	US	NGDP_SA_XDC	Q
...	...	...	...	...	...

Many Countries, Many Indicators

```
ifs_indicators = ifs_structure["Parameter Values"]["CL_INDICATOR_IFS"]
#gdp_indicators = @where(
gdp_indicators = @subset(
    ifs_indicators,
    occursin.("Gross Domestic Product", :description),
    occursin.("Domestic Currency", :description))
indicators = gdp_indicators[!,1]
countries  = ["GT","HN","SV","NI","CR","PA"]
countries_indicators = get_ifs_data(countries, indicators, "Q", 1900, 2100)
```

```
36-element Vector{IMFSeries}:
 IMF Data Series
Database: IFS
Area: GT
Indicator: NGDP_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: GT
Indicator: NGDP_SA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: GT
Indicator: NGDP_NSA_XDC
Description:
Frequency: Q
Time Period: 2013 to 2022
Data: 40 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: GT
Indicator: NGDP_R_XDC
Time Period: 1900 to 2100
```

```
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: GT
Indicator: NGDP_R_SA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: GT
Indicator: NGDP_R_NSA_XDC
Description:
Frequency: Q
Time Period: 2013 to 2022
Data: 40 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: HN
Indicator: NGDP_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: HN
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 2000 to 2023
Data: 93 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: HN
Indicator: NGDP_NSA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
```

```
Area: HN
Indicator: NGDP_R_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: HN
Indicator: NGDP_R_SA_XDC
Description:
Frequency: Q
Time Period: 2000 to 2023
Data: 93 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: HN
Indicator: NGDP_R_NSA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: SV
Indicator: NGDP_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period


 IMF Data Series
Database: IFS
Area: CR
Indicator: NGDP_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: CR
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1991 to 2023
```

```
Data: 129 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: CR
Indicator: NGDP_NSA_XDC
Description:
Frequency: Q
Time Period: 1991 to 2023
Data: 129 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: CR
Indicator: NGDP_R_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: CR
Indicator: NGDP_R_SA_XDC
Description:
Frequency: Q
Time Period: 1991 to 2023
Data: 129 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: CR
Indicator: NGDP_R_NSA_XDC
Description:
Frequency: Q
Time Period: 1991 to 2023
Data: 129 x 2 DataFrame

 IMF Data Series
Database: IFS
Area: PA
Indicator: NGDP_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period
```

```
 IMF Data Series
Database: IFS
Area: PA
Indicator: NGDP_SA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: PA
Indicator: NGDP_NSA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: PA
Indicator: NGDP_R_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: PA
Indicator: NGDP_R_SA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period

 IMF Data Series
Database: IFS
Area: PA
Indicator: NGDP_R_NSA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period
```

```
countries_indicators[3].series
```

	date	value
	Date	Float64
1	2013-03-01	1.03834e5
2	2013-06-01	102177.0
3	2013-09-01	1.02217e5
4	2013-12-01	1.08155e5
5	2014-03-01	1.09525e5
6	2014-06-01	1.09404e5
7	2014-09-01	111334.0
8	2014-12-01	1.17063e5
9	2015-03-01	1.179e5
10	2015-06-01	1.16185e5
11	2015-09-01	118716.0
12	2015-12-01	1.23222e5
13	2016-03-01	1.23842e5
14	2016-06-01	122800.0
15	2016-09-01	1.24517e5
16	2016-12-01	1.30842e5
17	2017-03-01	1.31681e5
18	2017-06-01	1.27649e5
19	2017-09-01	1.30292e5
20	2017-12-01	1.36885e5
21	2018-03-01	1.37006e5
22	2018-06-01	1.34676e5
23	2018-09-01	1.36448e5
24	2018-12-01	1.43238e5
25	2019-03-01	1.45002e5
26	2019-06-01	1.45013e5
27	2019-09-01	1.47695e5
28	2019-12-01	1.56262e5
29	2020-03-01	1.5166e5
30	2020-06-01	1.36646e5
...	...	...

```
ndf = size(countries_indicators)[1]
df = map(_ -> DataFrames.DataFrame(), 1:ndf)
[df[i] = get_df(i) for i in 1:ndf]
df = vcat(df...)
CSV.write(
    wd * "/IMFData_query.csv",
    delim = ";",
    df)
```

```
df
```

	date	value	country	indicator	frequency
	Date	Float64	String	String	String
1	2013-03-01	1.03834e5	GT	NGDP_NSA_XDC	Q
2	2013-06-01	102177.0	GT	NGDP_NSA_XDC	Q
3	2013-09-01	1.02217e5	GT	NGDP_NSA_XDC	Q
4	2013-12-01	1.08155e5	GT	NGDP_NSA_XDC	Q
5	2014-03-01	1.09525e5	GT	NGDP_NSA_XDC	Q
6	2014-06-01	1.09404e5	GT	NGDP_NSA_XDC	Q
7	2014-09-01	111334.0	GT	NGDP_NSA_XDC	Q
8	2014-12-01	1.17063e5	GT	NGDP_NSA_XDC	Q
9	2015-03-01	1.179e5	GT	NGDP_NSA_XDC	Q
10	2015-06-01	1.16185e5	GT	NGDP_NSA_XDC	Q
11	2015-09-01	118716.0	GT	NGDP_NSA_XDC	Q
12	2015-12-01	1.23222e5	GT	NGDP_NSA_XDC	Q
13	2016-03-01	1.23842e5	GT	NGDP_NSA_XDC	Q
14	2016-06-01	122800.0	GT	NGDP_NSA_XDC	Q
15	2016-09-01	1.24517e5	GT	NGDP_NSA_XDC	Q
16	2016-12-01	1.30842e5	GT	NGDP_NSA_XDC	Q
17	2017-03-01	1.31681e5	GT	NGDP_NSA_XDC	Q
18	2017-06-01	1.27649e5	GT	NGDP_NSA_XDC	Q
19	2017-09-01	1.30292e5	GT	NGDP_NSA_XDC	Q
20	2017-12-01	1.36885e5	GT	NGDP_NSA_XDC	Q
21	2018-03-01	1.37006e5	GT	NGDP_NSA_XDC	Q
22	2018-06-01	1.34676e5	GT	NGDP_NSA_XDC	Q
23	2018-09-01	1.36448e5	GT	NGDP_NSA_XDC	Q
24	2018-12-01	1.43238e5	GT	NGDP_NSA_XDC	Q
25	2019-03-01	1.45002e5	GT	NGDP_NSA_XDC	Q
26	2019-06-01	1.45013e5	GT	NGDP_NSA_XDC	Q
27	2019-09-01	1.47695e5	GT	NGDP_NSA_XDC	Q
28	2019-12-01	1.56262e5	GT	NGDP_NSA_XDC	Q
29	2020-03-01	1.5166e5	GT	NGDP_NSA_XDC	Q
30	2020-06-01	1.36646e5	GT	NGDP_NSA_XDC	Q
...	...	...	...	...	...

Examples

```
datasets = IMFData.get_imf_datasets()
```

	dataset_id	dataset_name
	String	String
1	AFRREO	Sub-Saharan Africa Regional Economic Outlook (AFRREO)
2	AFRREO201410	Sub-Saharan Africa Regional Economic Outlook (AFRREO) October 2014
3	AFRREO201504	Sub-Saharan Africa Regional Economic Outlook (AFRREO) April 2015
4	AFRREO201510	Sub-Saharan Africa Regional Economic Outlook (AFRREO) October 2015
5	AFRREO201604	Sub-Saharan Africa Regional Economic Outlook (AFRREO) April 2016
6	AFRREO201610	Sub-Saharan Africa Regional Economic Outlook (AFRREO) October 2016
7	AFRREO201704	Sub-Saharan Africa Regional Economic Outlook (AFRREO) April 2017
8	AFRREO201710	Sub-Saharan Africa Regional Economic Outlook (AFRREO) October 2017
9	AFRREO201804	Sub-Saharan Africa Regional Economic Outlook (AFRREO) April 2018
10	AFRREO201810	Sub-Saharan Africa Regional Economic Outlook (AFRREO) October 2018
11	AFRREO201904	Sub-Saharan Africa Regional Economic Outlook (AFRREO) April 2019
12	AFRREO201910	Sub-Saharan Africa Regional Economic Outlook (AFRREO) October 2019
13	AFRREO202004	Sub-Saharan Africa Regional Economic Outlook (AFRREO) April 2020
14	AFRREO202010	Sub-Saharan Africa Regional Economic Outlook (AFRREO) October 2020
15	APDREO	Asia and Pacific Regional Economic Outlook (APDREO)
16	APDREO201410	Asia and Pacific Regional Economic Outlook (APDREO) October 2014
17	APDREO201504	Asia and Pacific Regional Economic Outlook (APDREO) April 2015
18	APDREO201510	Asia and Pacific Regional Economic Outlook (APDREO) October 2015
19	APDREO201604	Asia and Pacific Regional Economic Outlook (APDREO) April 2016
20	APDREO201610	Asia and Pacific Regional Economic Outlook (APDREO) October 2016
21	APDREO201710	Asia and Pacific Regional Economic Outlook (APDREO) October 2017
22	APDREO2017M04	Asia and Pacific Regional Economic Outlook (APDREO) April 2017
23	APDREO201804	Asia and Pacific Regional Economic Outlook (APDREO) April 2018
24	APDREO201810	Asia and Pacific Regional Economic Outlook (APDREO) October 2018
25	APDREO201904	Asia and Pacific Regional Economic Outlook (APDREO) April 2019
26	APDREO201910	Asia and Pacific Regional Economic Outlook (APDREO) October 2019
27	APDREO202010	Asia and Pacific Regional Economic Outlook (APDREO) October 2020
28	BOP	Balance of Payments (BOP)
29	BOPAGG	Balance of Payments (BOP), World and Regional Aggregates
30	BOPAGG_2016	Balance of Payments (BOP), World and Regional Aggregates, 2016
...	...	...

```
# Use @where macro from DataFramesMeta to filter based on dataset name
ds_ifs = DataFramesMeta.@subset(datasets, DataFrames.occursin.("IFS", :dataset_id))
```

	dataset_id	dataset_name
	String	String
1	IFS	International Financial Statistics (IFS)
2	IFS_2017M08	International Financial Statistics (IFS), 2017 M08
3	IFS_2017M09	International Financial Statistics (IFS), 2017 M09
4	IFS_2017M10	International Financial Statistics (IFS), 2017 M10
5	IFS_2017M11	International Financial Statistics (IFS), 2017 M11
6	IFS_2017M12	International Financial Statistics (IFS), 2017 M12
7	IFS_2018M01	International Financial Statistics (IFS), 2018 M01
8	IFS_2018M02	International Financial Statistics (IFS), 2018 M02
9	IFS_2018M03	International Financial Statistics (IFS), 2018 M03
10	IFS_2018M04	International Financial Statistics (IFS), 2018 M04
11	IFS_2018M05	International Financial Statistics (IFS), 2018 M05
12	IFS_2018M06	International Financial Statistics (IFS), 2018 M06
13	IFS_2018M07	International Financial Statistics (IFS), 2018 M07
14	IFS_2018M08	International Financial Statistics (IFS), 2018 M08
15	IFS_2018M09	International Financial Statistics (IFS), 2018 M09
16	IFS_2018M10	International Financial Statistics (IFS), 2018 M10
17	IFS_2018M11	International Financial Statistics (IFS), 2018 M11
18	IFS_2018M12	International Financial Statistics (IFS), 2018 M12
19	IFS_2019M01	International Financial Statistics (IFS), 2019 M01
20	IFS_2019M02	International Financial Statistics (IFS), 2019 M02
21	IFS_2019M03	International Financial Statistics (IFS), 2019 M03
22	IFS_2019M04	International Financial Statistics (IFS), 2019 M04
23	IFS_2019M05	International Financial Statistics (IFS), 2019 M05
24	IFS_2019M06	International Financial Statistics (IFS), 2019 M06
25	IFS_2019M07	International Financial Statistics (IFS), 2019 M07
26	IFS_2019M08	International Financial Statistics (IFS), 2019 M08
27	IFS_2019M09	International Financial Statistics (IFS), 2019 M09
28	IFS_2019M10	International Financial Statistics (IFS), 2019 M10
29	IFS_2019M11	International Financial Statistics (IFS), 2019 M11
30	IFS_2019M12	International Financial Statistics (IFS), 2019 M12
...	...	...

```
ds_dot = DataFramesMeta.@subset(datasets, DataFrames.occursin.("DOT", :dataset_id))
```

	dataset_id	dataset_name
	String	String
1	DOT	Direction of Trade Statistics (DOTS)
2	DOT_2017Q1	Direction of Trade Statistics (DOTS), 2017 Q1
3	DOT_2017Q2	Direction of Trade Statistics (DOTS), 2017 Q2
4	DOT_2017Q3	Direction of Trade Statistics (DOTS), 2017 Q3
5	DOT_2017Q4	Direction of Trade Statistics (DOTS), 2017 Q4
6	DOT_2018Q1	Direction of Trade Statistics (DOTS), 2018 Q1
7	DOT_2018Q2	Direction of Trade Statistics (DOTS), 2018 Q2
8	DOT_2018Q3	Direction of Trade Statistics (DOTS), 2018 Q3
9	DOT_2018Q4	Direction of Trade Statistics (DOTS), 2018 Q4
10	DOT_2019Q1	Direction of Trade Statistics (DOTS), 2019 Q1
11	DOT_2019Q2	Direction of Trade Statistics (DOTS), 2019 Q2
12	DOT_2019Q3	Direction of Trade Statistics (DOTS), 2019 Q3
13	DOT_2019Q4	Direction of Trade Statistics (DOTS), 2019 Q4
14	DOT_2020Q1	Direction of Trade Statistics (DOTS), 2020 Q1
15	DOT_2020Q2	Direction of Trade Statistics (DOTS), 2020 Q2
16	DOT_2020Q3	Direction of Trade Statistics (DOTS), 2020 Q3
17	DOT_2020Q4	Direction of Trade Statistics (DOTS), 2020 Q4
18	DOT_2021Q1	Direction of Trade Statistics (DOTS), 2021 Q1
19	DOT_2021Q2	Direction of Trade Statistics (DOTS), 2021 Q2
20	DOT_2021Q3	Direction of Trade Statistics (DOTS), 2021 Q3
21	DOT_2021Q4	Direction of Trade Statistics (DOTS), 2021 Q4
22	DOT_2022Q1	Direction of Trade Statistics (DOTS), 2022 Q1
23	DOT_2022Q2	Direction of Trade Statistics (DOTS), 2022 Q2
24	DOT_2022Q3	Direction of Trade Statistics (DOTS), 2022 Q3
25	DOT_2022Q4	Direction of Trade Statistics (DOTS), 2022 Q4
26	DOT_2023Q1	Direction of Trade Statistics (DOTS), 2023 Q1

```
ifs_structure  = IMFData.get_imf_datastructure("IFS")
```

```
Dict{String, Any} with 2 entries:
  "Parameter Names"  => 5×2 DataFrame…
  "Parameter Values" => Dict{Any, Any}("CL_INDICATOR_IFS"=>1679×2 DataFrame…
```

Search for GDP indicators

```
ifs_indicators = ifs_structure["Parameter Values"]["CL_INDICATOR_IFS"]
gdp_indicators = DataFramesMeta.@subset(
    ifs_indicators,
```

```
    DataFrames.occursin.("Gross Domestic Product", :description),
    DataFrames.occursin.("Domestic Currency", :description))
CSV.write(wd * "/ifs_gdp_indicators.csv", gdp_indicators; delim='\t')
```

```
"C:\\Directorio_Trabajo\\Julia\\IMFData/ifs_gdp_indicators.csv"
```

Availability of Data

```
indic = "NGDP_SA_XDC"
area  = "US"
data_available = IMFData.get_ifs_data(area, indic, "Q", 1900, 2100)
```

```
IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA_XDC
Description:
Frequency: Q
Time Period: 1950 to 2023
Data: 293 x 2 DataFrame
```

```
data_not_available = get_ifs_data(area, indic, "M", 1900, 2100)
```

```
IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA_XDC
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period
```

```
data_not_defined = get_ifs_data(area, "NGDP_SA", "Q", 1900, 2100)
```

```
IMF Data Series
Database: IFS
Area: US
Indicator: NGDP_SA
Time Period: 1900 to 2100
Note: Indicator not defined for the given area or time period
```