

CHAPTER 8

US FEDERAL RESERVE BANK

```
> library(fImport)
```

The FRED2 database of the Feder Reserve Bank in St. Louis contains more than 20'000 time series for download. The time series are divided into categories. Most of the time series are economic time series, but we also find series of historical foreign exchange rates, and interest rates and bond prices on the server. The spectrum of the historical times series is categorised into the following topics

LISTING 8.1: FRED2 TIME SERIES CATEGORIES

- Banking
- Business/Fiscal
- Consumer Price Indexes (CPI)
- Employment & Population
- Exchange Rates
- Foreign Exchange Intervention
- Gross Domestic Product (GDP) and Components
- Interest Rates
- Monetary Aggregates
- Producer Price Indexes (PPI)
- Reserves and Monetary Base
- U.S. Trade & International Transactions
- U.S. Financial Data
- Regional Data

To access the FRED2 database follow this link in your browser:

<http://research.stlouisfed.org/fred2>

In this chapter we present how to manage the download of daily time series from the category "Interest Rates".

8.1 THE DOWNLOAD URL

When we know the symbol name of a time series, e.g. `textttDAAA` for *Moody's Seasoned Aaa Corporate Bond Yield* then we can download the data as a CSV file

```
http://research.stlouisfed.org/fred2/series/DAAA/downloaddata/DAAA.csv
```

For other time series we have just to modify the symbol name in the URL. How we can find out the symbol names on the server? The Federal Reserve divides the time series into categories and list the members on individual web pages. In the following we will show how we can access the categories and how we can create listings for an easy search of the symbol names.

8.2 DOWNLOADING A TIME SERIES

To download a historical time series from the FRED data base we have just to know the symbol name. The following example shows how to download the daily time series records for *Moody's Seasoned Aaa Corporate Bond Yield*.

Example: Download Moody's Seasoned Aaa Corporate Bond Yield

First set the symbol name, which is "DAAA"

```
> NAME <- "DAAA"
```

Note, all daily time series names start with a "D", weekly with a "W", and monthly with a "M". Then compose the download URL

```
> URL <- composeURL(
  "research.stlouisfed.org/fred2/series/",
  NAME,
  "/downloaddata/",
  NAME,
  ".csv")
```

and get the datafile using the function `read.csv()`. We use the function `read.csv()` since the FRED data base of the Federal Reserve keeps the data records in CSV formatted files.

```
> Download <- read.csv(URL, stringsAsFactors = FALSE)
```

The downloaded file is a data frame. Have a look on dimension, the start date and the last few records

```
> class(Download)
```

```
[1] "data.frame"
> tail(Download)

      DATE VALUE
7109 2010-04-01 5.33
7110 2010-04-02 5.40
7111 2010-04-05 5.44
7112 2010-04-06 5.45
7113 2010-04-07 5.34
7114 2010-04-08 5.35
```

Finally we convert the data frame into an object of class `timeSeries`, omitting NA's which were created from missing data records on weekends and holidays.

```
> DAAA <- timeSeries(
  data = as.numeric(Download[, 2]),
  charvec = format(Download[, 1]),
  units = NAME)
> DAAA <- na.omit(DAAA)
> start(DAAA)

GMT
[1] [1983-01-03]

> tail(DAAA)

GMT
      DAAA
2010-04-01 5.33
2010-04-02 5.40
2010-04-05 5.44
2010-04-06 5.45
2010-04-07 5.34
2010-04-08 5.35
```

8.3 THE FUNCTION `fredDownload()`

The sequence of the above code snippets can be used to write a download function

```
> fredDownload <- function(name) {
  # Compose Download URL:
  URL <- composeURL(
    "research.stlouisfed.org/fred2/series/", name,
    "/downloaddata/", name, ".csv")

  # Download Data:
  Download <- read.csv(URL, stringsAsFactors = FALSE)

  # Convert to timeSeries:
  data <- as.numeric(Download[, 2])
  charvec <- as.character(Download[, 1])
  units <- name
```

```

    ts <- na.omit(timeSeries(data, charvec, units))

    # Return Value:
    ts
}

```

The returned value of the function `fredDownload()` is an S4 object of class `timeSeries`.

Example: Download the Daily Effective Federal Funds Rate

As an example we download the historical time series records for the daily effective Federal Funds rate

```

> DFF <- fredDownload("DFF")
> start(DFF)
GMT
[1] [1954-07-01]

> tail(DFF)
GMT
          DFF
2010-04-03 0.20
2010-04-04 0.20
2010-04-05 0.20
2010-04-06 0.20
2010-04-07 0.19
2010-04-08 0.19

```

Note, we have always to download the whole time series since inception until the last record, even we like to download the data for the last 4 weeks.

8.4 TIME SERIES LISTINGS

The historical data sets on the FRED2 data base are stored by categories. In the following sections we show how to create listings from the categories.

8.5 TIME SERIES CATEGORIES

The categories with their internal code numbers can be extracted from the web site.

Example: Downloading Time Series Categories

For this we proceed in the following way. First download the listing of categories.

```

> URL <- "http://research.stlouisfed.org/fred2"
> Download <- read.lynx(URL)

```

Then extract categories and their descriptions

```
> CATEGORIES <- substring(indexGrep("/categories/", Download, perl = TRUE), 55)[-1]
> START <- grep("  Categories", Download, perl = TRUE) + 1
> END <- START + length(CATEGORIES) - 1
> DESCRIPTIONS <- substring(Download[START:END], 8)
> fredMainListing <- data.frame(
  Category = CATEGORIES,
  Description = DESCRIPTIONS,
  stringsAsFactors = FALSE)
```

and print the main listing

```
> fredMainListing
  Category      Description
1        23      Banking
2         1 Business/Fiscal
3         9 Consumer Price Indexes (CPI)
4        10 Employment & Population
5        15 Exchange Rates
6    32145 Foreign Exchange Intervention
7        18 Gross Domestic Product (GDP) and Components
8        22 Interest Rates
9        24 Monetary Aggregates
10       31 Producer Price Indexes (PPI)
11       45 Reserves and Monetary Base
12       13 U.S. Trade & International Transactions
13       46 U.S. Financial Data
14     3008 Regional Data
```

Example: Downloading Sub Category 22

Among this list we find Category No. 22 for "Interest Rates". Have a look on the web site

<http://research.stlouisfed.org/fred2/categories/22>

we find a list of sub categories. Now we download from this site the sub categories for category 22

```
> CATEGORY <- 22
> URL <- composeURL("research.stlouisfed.org/fred2/categories/", CATEGORY)
> Download <- read.lynx(URL)
```

and again we extract the category numbers and their descriptions:

```
> CATEGORIES <- substring(indexGrep("/categories/[0-9]*$", Download, perl = TRUE), 55)
> START <- grep("  Categories:", Download, perl = TRUE) + 1
> END <- START + length(CATEGORIES) - 1
> DESCRIPTIONS <- gsub("^ *\\[.*\\]", "", Download[START:END], perl = TRUE)
> fredCategory22 <- data.frame(
  Category = CATEGORIES,
  Description = DESCRIPTIONS,
  stringsAsFactors = FALSE)
```

Print the listing for category 22

```
> fredCategory22
  Category                               Description
1      121          Certificates of Deposit (9)
2      120          Commercial Paper (25)
3      119          Corporate Aaa & Baa (6)
4      118 FRB Rates - discount, fed funds, primary credit (20)
5      117          Prime Bank Loan Rate (4)
6      116          Treasury Bills (21)
7      115          Treasury Constant Maturity (48)
8       82          Treasury Inflation-Indexed Securities (112)
9      114          30yr Mortgage (4)
10     113          Other (15)
```

Example: Write a Function to Download the Sub Categories

When we use these code snippets to write a function `fredCategoryListing()` for the listing of categories it becomes simple to create the listings for further categories

```
> fredCategoryListing <- function(category) {
  # Compose URL and Download Data:
  URL <- composeURL("research.stlouisfed.org/fred2/categories/", category)
  Download <- read.lynx(URL)

  # Extract Categories and Descriptions:
  CATEGORIES <- substring(indexGrep("/categories/[0-9]*$", Download, perl = TRUE), 55)
  START <- grep("  Categories:", Download, perl = TRUE) + 1
  END <- START + length(CATEGORIES) - 1
  DESCRIPTIONS <- gsub("^ *\\[.*\\]", "", Download[START:END], perl = TRUE)
  fredCategory <- data.frame(
    Category = CATEGORIES,
    Description = DESCRIPTIONS,
    stringsAsFactors = FALSE)

  # Return Value:
  fredCategory
}
```

Example: Generate a Complete Listing of Sub Categories

Now let us generate a complete listing. First we create a list of the major categories. Note the loop over all categories will create a long listing.

```
> Categories <- MainCategories <- Descriptions <- NULL
> CATEGORIES <- fredMainListing[, 1]
> for (i in 1:length(CATEGORIES)) {
  CATEGORY <- as.numeric(CATEGORIES[i])
  if (CATEGORY < 100) {
    cat("\nCategory: ", CATEGORY, " ", fredMainListing[i, 2], "\n\n")
    if (CATEGORY == 9) {
```

```

    newCategories <- as.vector(fredMainListing[i, 1])
    newDescriptions <- as.vector(fredMainListing[i, 2])
    newDescriptions <- gsub("CPI", "28", newDescriptions, perl = TRUE)
  } else if (CATEGORY == 31) {
    newCategories <- as.vector(fredMainListing[i, 1])
    newDescriptions <- as.vector(fredMainListing[i, 2])
    newDescriptions <- gsub("PPI", "18", newDescriptions, perl = TRUE)
  } else {
    y <- fredCategoryListing(CATEGORY)
    newCategories <- as.vector(y[, 1])
    newDescriptions <- as.vector(y[, 2])
  }
  ans <- cbind(
    Category = newCategories,
    Description = newDescriptions)
  rownames(ans) = 1:nrow(ans)
  colnames(ans) = c("CATEGORY", "DESCRIPTION")
  print(as.data.frame(ans), quote = FALSE)

  Categories <- c(Categories, newCategories)
  MainCategories <- c(MainCategories, rep(CATEGORY, length(newCategories)))
  Descriptions <- c(Descriptions, newDescriptions)
}
}

```

Category: 23 Banking

	CATEGORY	DESCRIPTION
1	83	Condition of Banks (267)
2	64	8th District Banking Performance (105)
3	101	Commercial Credit (31)
4	100	Loans (8)
5	99	Securities & Investments (3)

Category: 1 Business/Fiscal

	CATEGORY	DESCRIPTION
1	4	Employment Cost Index (17)
2	5	Federal Government Debt (18)
3	97	Household Sector (34)
4	3	Industrial Production (20)
5	2	Productivity & Cost (24)
6	6	Retail Sales (10)
7	98	Other Economic Indicators (25)
8	32216	Health Insurance (5)
9	32217	Gas Prices (60)

Category: 9 Consumer Price Indexes (CPI)

	CATEGORY	DESCRIPTION
1	9	Consumer Price Indexes (28)

Category: 10 Employment & Population

	CATEGORY	DESCRIPTION
1	11	Establishment Survey Data (27)

2 12 Household Survey Data (14)
 3 104 Population (3)

Category: 15 Exchange Rates

CATEGORY	DESCRIPTION
1 94	Daily Rates (26)
2 95	Monthly Rates (39)
3 32219	Annual Rates (26)
4 105	Trade-Weighted Indexes (14)
5 158	By Country (81)

Category: 18 Gross Domestic Product (GDP) and Components

CATEGORY	DESCRIPTION
1 106	GDP/GNP (23)
2 107	Gov't Receipts, Expenditures & Investment (76)
3 108	Imports & Exports (22)
4 109	Industry (15)
5 110	Personal Income & Outlays (34)
6 21	Price Indexes & Deflators (9)
7 112	Saving & Investment (29)

Category: 22 Interest Rates

CATEGORY	DESCRIPTION
1 121	Certificates of Deposit (9)
2 120	Commercial Paper (25)
3 119	Corporate Aaa & Baa (6)
4 118	FRB Rates - discount, fed funds, primary credit (20)
5 117	Prime Bank Loan Rate (4)
6 116	Treasury Bills (21)
7 115	Treasury Constant Maturity (48)
8 82	Treasury Inflation-Indexed Securities (112)
9 114	30yr Mortgage (4)
10 113	Other (15)

Category: 24 Monetary Aggregates

CATEGORY	DESCRIPTION
1 25	M1 and Components (26)
2 29	M2 and Components (33)
3 96	M2 Minus Small Time Deposits (3)
4 28	M3 and Components (23)
5 30	MZM (5)
6 26	Memorandum Items (7)
7 52	Other (6)

Category: 31 Producer Price Indexes (PPI)

CATEGORY	DESCRIPTION
1 31	Producer Price Indexes (18)

Category: 45 Reserves and Monetary Base

CATEGORY	DESCRIPTION
1 122	Borrowings (8)
2 32215	Factors Affecting Reserve Balances (39)
3 32218	Maturity Distribution of Term Auction Credit, Other
4 124	Loans, and Securities (54)
5 123	Monetary Base (19)
6 342	Reserves (22)

Category: 13 U.S. Trade & International Transactions

CATEGORY	DESCRIPTION
1 16	Exports (42)
2 17	Imports (42)
3 3000	Income Payments & Receipts (45)
4 125	Trade Balance (24)
5 127	U.S. International Finance (96)
6 32220	Trade Indexes (832)

Category: 46 U.S. Financial Data

CATEGORY	DESCRIPTION
1 49	Commercial Banking (22)
2 32141	Exchange Rates (9)
3 47	Interest Rates (139)
4 48	Monetary (108)
5 50	Reserves (45)
6 51	Discontinued (19)

Then we create the listing

```
> fredCategories <- data.frame(
  CATEGORY = Categories,
  MAINCATEGORY = MainCategories,
  DESCRIPTION = Descriptions,
  stringsAsFactors = FALSE)
```

Note, here we restricted the generation of the list to the first 100 categories which are the most important ones. Category 9 and 31 have to be considered separately, since they list no sub categories and contribute thus directly to the listing.

8.6 TIME SERIES SELECTION

From the listings we can select a category, e.g. 116, which is the category for Treasury Bills. Have a look on the web site

<http://research.stlouisfed.org/fred2/categories/116>

On this site we select an instrument, e.g. "DTB3" the daily historical data series for 3-month Treasury Bills. If you select them you will be directed to the chart page. The link to this web page is

<http://research.stlouisfed.org/fred2/series/DTB3?cid=116>

On this web page we also find the download "Link: Download Data". Follow this link

```
http://research.stlouisfed.org/fred2/series/DTB3/downloaddata?cid=116
```

and we are directed to the download page. On this page you can select the data range and the file format. For the file format we make the selection "Text, Comma separated". Then we press the download button. the returned file comes with the name "DTB3.csv". From the source Code of the HTML page we can extract the link

```
http://research.stlouisfed.org/fred2/series/DTB3/downloaddata/DTB3.csv
```

The construction of the name of the link can be generalised to other data sets.

Example: Download 1-Year Treasury Bill Rates

Select the value for the 1-Year Treasury Bill Secondary Market Rates and download the data records. First we search for the symbol name in category 116

```
> DTB1YR <- fredDownload("DTB1YR")
```

Have a look on the data set

```
> start(DTB1YR)
GMT
[1] [1959-07-15]

> tail(DTB1YR)
GMT
          DTB1YR
2010-04-01  0.39
2010-04-02  0.42
2010-04-05  0.44
2010-04-06  0.48
2010-04-07  0.46
2010-04-08  0.45
```

8.7 SUB CATEGORY LISTINGS

Starting from the categories we can generate listing for the historical time series. As an example we consider the category for the "Interest Rates". The sub categories are

LISTING 8.2: CATEGORY OF INTEREST RATES

Cerificates of Deposit
 Commercial Papers
 Corporate Aaa \& Baa
 FRB Rates, Fed Funds, Primary Credit
 Prime Bank Loan Rate
 Treasury Bills

beside some others.

Example: Create a Listing for Certificates of Deposit

The certificates of deposit are listed on FRED's web page

<http://research.stlouisfed.org/fred2/categories/121>

To make the listing comprehensive we write a small function `fredSymbolListing()` which will help us to generate listings for the different categories.

```
> fredSymbolListing <- function(category) {
  # Compose URL:
  URL <- composeURL(
    "research.stlouisfed.org/fred2/categories/", category)

  # Download and Clean Data:
  Download <- read lynx(URL)
  download <- substring(indexGrep("\\[_\\]", Download, perl = TRUE), 12)

  # Select Daily Series:
  download <- indexGrep("^D", download, perl = TRUE)

  # If there are discontinued series, remove them ...
  if (length(grep("DISCO", download, perl = TRUE)) > 0)
    download <- download[-grep("DISCO", download, perl = TRUE)]

  # Create Listing:
  SYMBOL <- gsub(".*$", "", download, perl = TRUE)
  DESCRIPTION <- substring(download, nchar(SYMBOL)+2)
  data.frame(SYMBOL, DESCRIPTION, stringsAsFactors = FALSE)
}
```

Note, to suppress discontinued series we added a line to the code to remove them. Furthermore, some line finish not properly, they are too long, ending with a date which has its origin from the next field of the table in the HTML File. We also have suppressed these unwanted parts at the end of the descriptions.

And now create a listing for category 120

```
> certificatesListing <- fredSymbolListing(120)
> certificatesListing
```

	SYMBOL	DESCRIPTION
1	DCPF1M	1-Month AA Financial Commercial Paper Rate 1997-01-02
2	DCPN30	1-Month AA Nonfinancial Commercial Paper Rate
3	DCPF2M	2-Month AA Financial Commercial Paper Rate 1997-01-02
4	DCPN2M	2-Month AA Nonfinancial Commercial Paper Rate
5	DCPF3M	3-Month AA Financial Commercial Paper Rate 1997-01-02
6	DCPN3M	3-Month AA Nonfinancial Commercial Paper Rate

The returned data frame has two columns SYMBOL and DESCRIPTION.

Exercise: Create a Listing for Commercial Papers

The commercial papers have category number 120. Create a listing with the name commercialPapersListing using the function fredSymbolListing().

Example: Create a Listing for Corporate Aaa & Baa

The number of the category is 119 and the listing becomes

```
> corporateListing <- fredSymbolListing(category = 119)
> corporateListing
```

	SYMBOL	DESCRIPTION
1	DAAA	Moody's Seasoned Aaa Corporate Bond Yield 1983-01-03
2	DBAA	Moody's Seasoned Baa Corporate Bond Yield 1986-01-02

Example: Create a Listing for FRB Rates, Fed Funds, Primary Credit

The number of the category is 118 and the listing becomes

```
> fedFundsListing <- fredSymbolListing(118)
> fedFundsListing
```

	SYMBOL	DESCRIPTION
1	DFF	Effective Federal Funds Rate 1954-07-01 2010-04-08 D % NA
2	DFEDTARL	Federal Funds Target Range - Lower Limit 2008-12-16
3	DFEDTARU	Federal Funds Target Range - Upper Limit 2008-12-16
4	DISCNTD8	Federal Reserve Bank of St. Louis Basic Discount Rate
5	DPCREDIT	Primary Credit Rate 2003-01-09 2010-04-08 D % NA

Example: Create a Listing for Prime Bank Loan Rate

The number of the category is 117 and the listing becomes

```
> primeRateListing <- fredSymbolListing(117)
> primeRateListing
```

	SYMBOL	DESCRIPTION
1	DPRIME	Bank Prime Loan Rate 1955-08-04 2010-04-08 D % NA

Example: Create a Listing for Treasury Bills

The number of the category is 116 and the listing becomes

```
> tBillsListing <- fredSymbolListing(116)
> tBillsListing
```

	SYMBOL	DESCRIPTION
1	DTB1YR	1-Year Treasury Bill: Secondary Market Rate 1959-07-15
2	DTB3	3-Month Treasury Bill: Secondary Market Rate 1954-01-04
3	DTB4WK	4-Week Treasury Bill: Secondary Market Rate 2001-07-31
4	DTB6	6-Month Treasury Bill: Secondary Market Rate 1958-12-09

Example: Create a Listing for Treasury Constant Maturity

The number of the category is 115 and the listing becomes

```
> constMaturityListing <- fredSymbolListing(115)
> constMaturityListing
```

	SYMBOL	DESCRIPTION
1	DGS10	10-Year Treasury Constant Maturity Rate 1962-01-02
2	DFII10	10-Year Treasury Inflation-Indexed Security, Constant
3	DGS1M0	1-Month Treasury Constant Maturity Rate 2001-07-31
4	DGS1	1-Year Treasury Constant Maturity Rate 1962-01-02
5	DGS20	20-Year Treasury Constant Maturity Rate 1993-10-01
6	DFII20	20-Year Treasury Inflation-Indexed Security, Constant
7	DGS2	2-Year Treasury Constant Maturity Rate 1976-06-01
8	DGS30	30-Year Treasury Constant Maturity Rate 1977-02-15
9	DFII30	30-Year Treasury Inflation-Indexed Security, Constant
10	DGS3M0	3-Month Treasury Constant Maturity Rate 1982-01-04
11	DGS3	3-Year Treasury Constant Maturity Rate 1962-01-02
12	DGS5	5-Year Treasury Constant Maturity Rate 1962-01-02
13	DFII5	5-Year Treasury Inflation-Indexed Security, Constant
14	DGS6M0	6-Month Treasury Constant Maturity Rate 1982-01-04
15	DGS7	7-Year Treasury Constant Maturity Rate 1969-07-01
16	DFII7	7-Year Treasury Inflation-Indexed Security, Constant

Example: Create a Listing for Treasury Inflation Index Securities

The number of the category is 89 and the listing becomes

```
> inflationIndexedListing <- fredSymbolListing(89)
> inflationIndexedListing
```

[1]	SYMBOL	DESCRIPTION
<0 rows>		(or 0-length row.names)

Example: Create a Listing for 30 Year Mortgage

The number of the category is 114 and the listing becomes

```
> mortgageListing <- fredSymbolListing(114)
> mortgageListing
```

```
[1] SYMBOL      DESCRIPTION
<0 rows> (or 0-length row.names)
```

Example: Create a Listing for Treasury Inflation Index Securities

The number of the category is 113 and the listing becomes

```
> inflationIndexedListing <- fredSymbolListing(113)
> inflationIndexedListing
      SYMBOL      DESCRIPTION
1 DLTBOARD Long-Term Composite Rate of U.S. Treasury Securities
2      DLTA      Treasury Long-Term Average (25 years and above)
```

8.8 FUNCTION SUMMARY

In this chapter we have written functions to generate listings of categories and to download time series data from the Federal Reserve data base FRED2. The functions are

LISTING 8.3: FUNCTIONS TO DOWNLOAD DATA FROM FRED2

Functions:

fredCategoryListing	creates a category listing
fredSymbolListing	creates a listing of symbol names and descriptions
fredDownlod	downloads a time series from the FRED2 data base

Arguments:

category	an integer, the number of the (sub) category
name	a character string, the symbol name
