

SPSS Scripting and Automation

MARTIN SEBERA

FACULTY OF SPORTS
STUDIES, MASARYK
UNIVERSITY, BRNO

SEBERA@FSPS.MUNI.CZ

JANUARY 2024

Hello, it is me 😊

Orcid: 0000-0003-3750-1549

Scopus Author ID: 36165218300

ResearcherID: M-9818-2018

**Without syntax,
SPSS would be
just another
beautiful GUI
without a soul**

What is scripting and automation?

1. A script is a set of predefined instructions or commands that are executed by the software to perform specific actions.
2. These commands can be saved in a syntax file and executed to perform data manipulation, analysis, and graphical representation tasks automatically.

What is scripting and automation?

It allows users to:

- a) Apply the same set of commands to multiple data files.
- b) Automate complex data transformations and analyses.
- c) Schedule tasks to run at specific times.
- d) Reduce the potential for human error in repeated steps.
- e) Document the analysis process for replication or review.

Scripting and Motivation BITCOIN

Bitcoin

40 386 EUR

-254.438

-0.63%

04.01.2024

04.01.2022

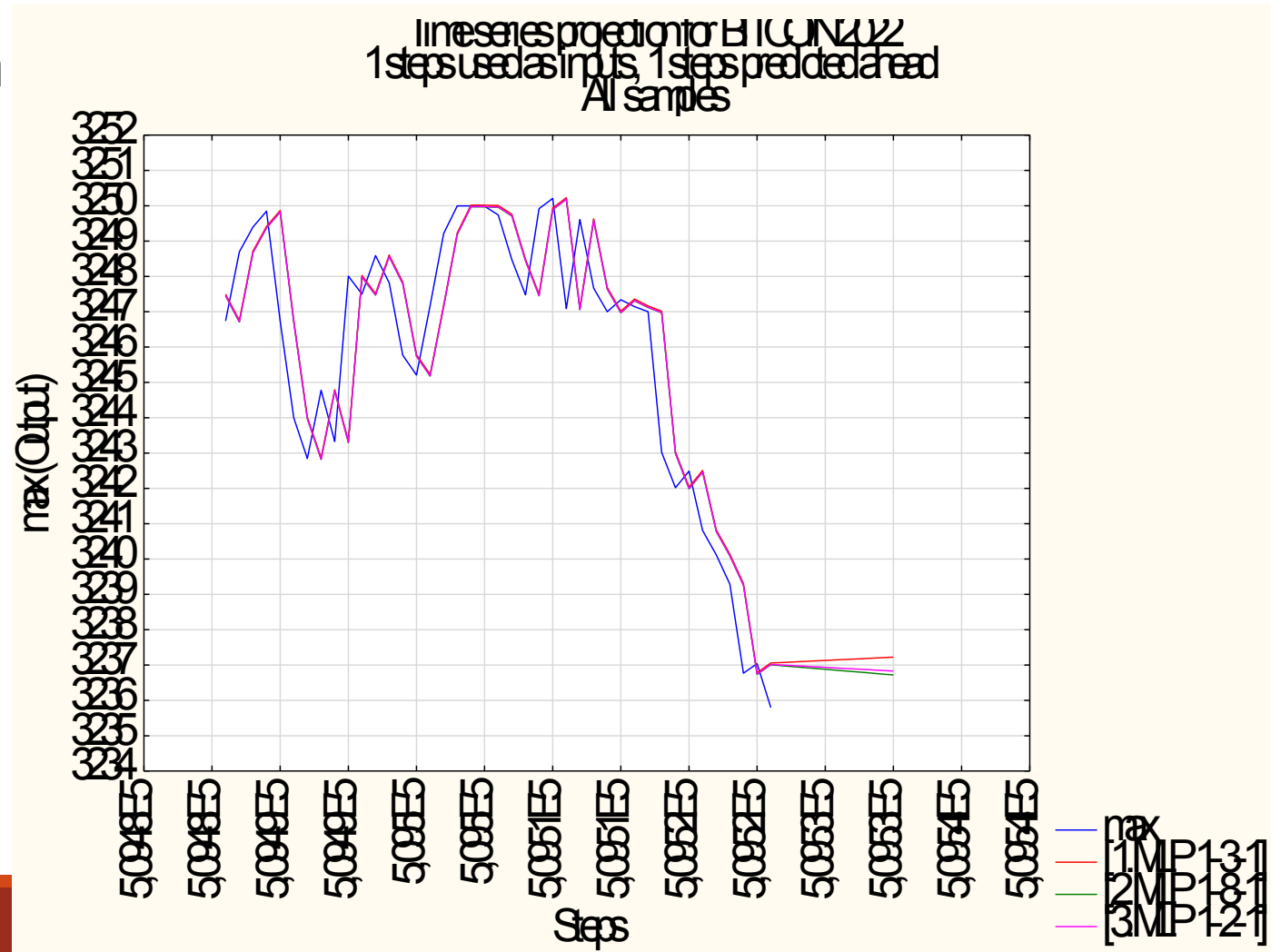


KURZYCZ

Bitcoin

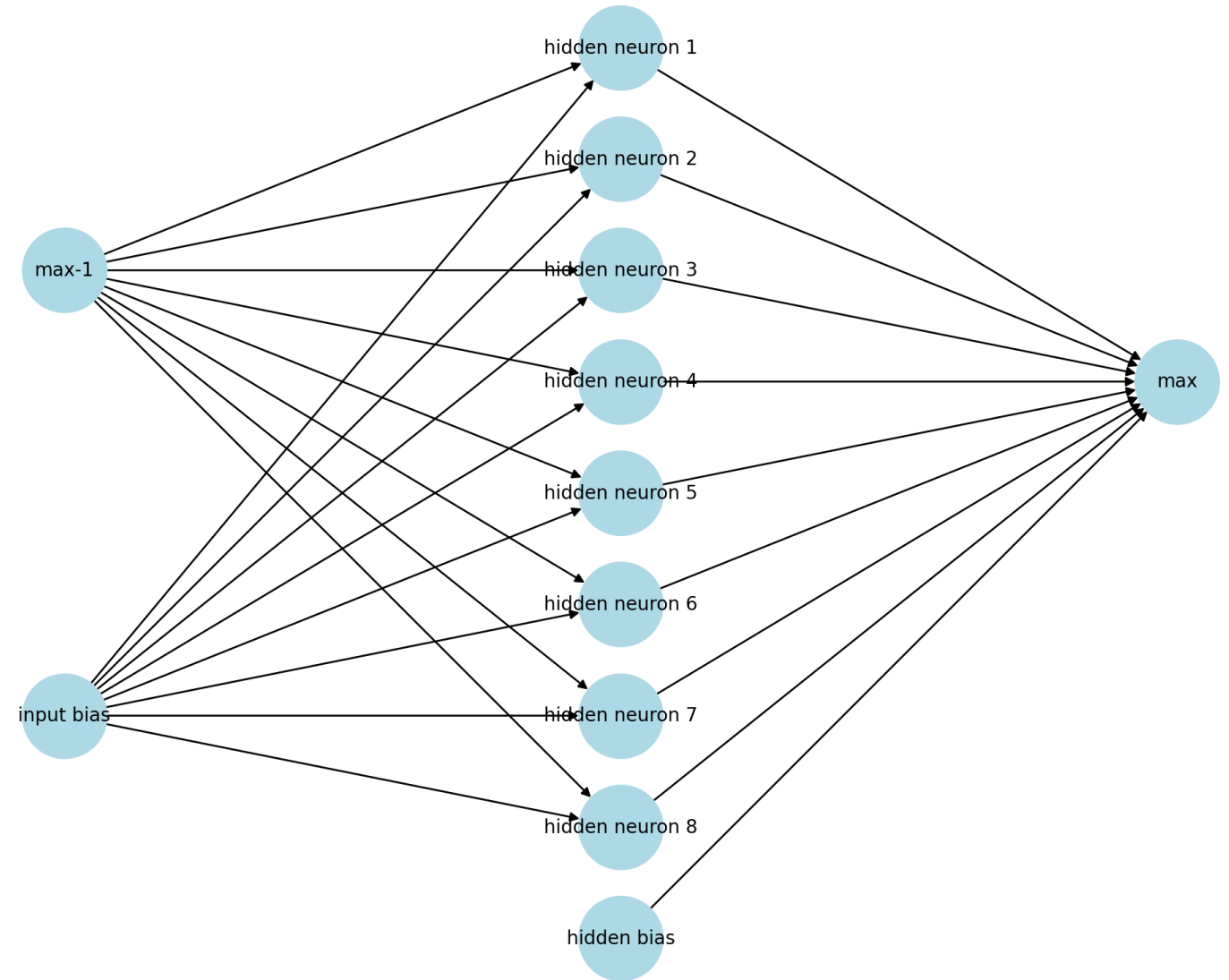
Scripting and motivation - BITCOIN

- a) the Bitcoin exchange rate in 2022 and
- b) **neural network model** that predicts the values of this exchange rate very well



Neural Network Graph Representation

- i. Neural network is simple.
- ii. 1 input neuron, 8 hidden neurons, 1 hidden bias and 1 "max" output neuron. The inputs are connected to all hidden neurons. All hidden neurons and "hidden bias" are connected to the output neuron "max".




```

void NN,ScaleInputs(double* input, double minimum, double maximum, int nContTargets, int steps)
{
    for(i=0; i<nContTargets; i++)
    {
        delta = (maximum-minimum)/(NN,max_input[i]-NN,min_input[i]);
        for(j=0; j<steps; j++)
            input[i+j*nContTargets] = minimum - delta*NN,min_input[i]+ delta*input[i+j*nContTargets];
    }
}

```

```

void NN,UnscaleTargets(double* output, double minimum, double maximum, int size)
{
    for(i=0; i<size; i++)
    {
        delta = (maximum-minimum)/(NN,max_target[i]-NN,min_target[i]);
        output[i] = (output[i] - minimum + delta*NN,min_target[i])/delta;
    }
}

```

```

void NN,ComputeFeedForwardSignals(double* MAT_INOUT,double* V_IN,double* V_OUT, double* V_BIAS,int size1,int size2,int layer)
{
    for(row=0;row < size2; row++)
    {
        V_OUT[row]=0.0;
        for(col=0;col<size1;col++)V_OUT[row]+=(*(MAT_INOUT+(row*size1)+col)*V_IN[col]);
        V_OUT[row]+=V_BIAS[row];
        if(layer==0) V_OUT[row] = tanh(V_OUT[row]);
    }
}

```

```

void NN,RunNeuralNet_TS_Reg ()
{
    NN,ComputeFeedForwardSignals((double*)NN,input_hidden_weights,NN,input,NN,hidden,NN,hidden_bias,1, 2,0);
    NN,ComputeFeedForwardSignals((double*)NN,hidden_output_wts,NN,hidden,NN,output,NN,output_bias,2, 1,1);
}

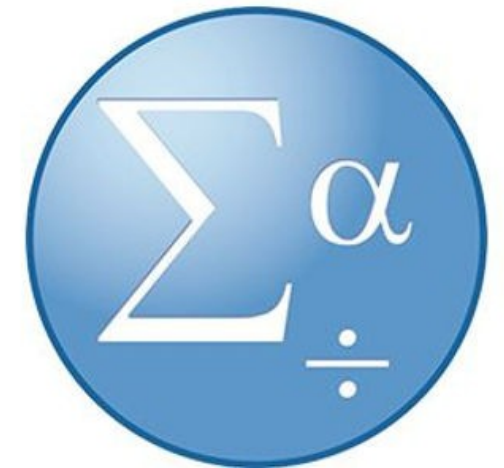
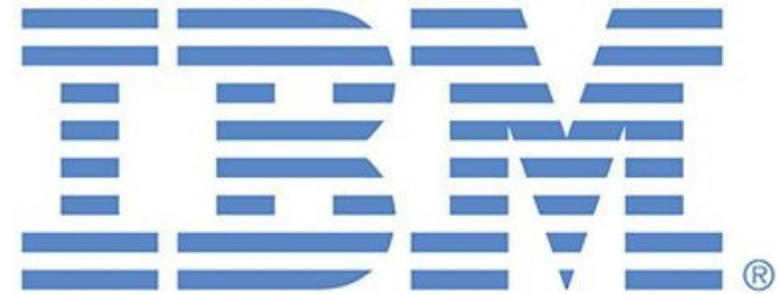
```

Scripting and motivation - BITCOIN

- a) Forecasting the price of Bitcoin, like any other cryptocurrency or financial asset, is very **complex and involves a number of challenges and uncertainties, external influences and unpredictable moments.**
- b) In retrospect, I could model the development of the course, but the prediction is actually poorly modeled. Well, because I don't have the other input information.

Is SPSS a programming language?

1. SPSS Syntax is a command language that is unique to SPSS Statistics.
2. Effectively, it's the SPSS operating system.
3. The language was developed in the 1960's and so pre-dates Windows/Mac versions of SPSS by some margin



Why use syntax?

1. Automate your entire analytical process
2. Complete repetitive tasks more quickly
3. Change/fix something to update all your work
4. Share your data preparation and analysis with others
5. A huge range of tasks in IBM SPSS Statistics can be completed using SPSS Syntax
6. SPSS Syntax is a command language that is much easier to understand than many analytical programming languages such as Python or R

What is Syntax?

1. SPSS syntax is a programming language that is unique to SPSS. It allows you to write commands that run SPSS procedures, rather than using the graphical user interface.
2. Syntax allows users to perform tasks that would be too tedious or difficult to do using the drop-down menus. This is the case when you are re-running the same analysis many times.
3. Note that the two methods of interacting with SPSS — **drop-down menus** and **syntax** — **are not mutually exclusive**. You can use both methods if you wish, or switch from one method to the other depending on the actions you want to perform.

Basic Syntax Rules

FORMATTING

Statements in SPSS end with a period.

SPSS syntax is not case-sensitive. You can use all lower case, all upper case, or a mixture of both when writing syntax.

COMMENTS

A comment is a line of text in a program that is not read by the computer as a command. In SPSS syntax, placing an asterisk (*) or a forward-slash followed by an asterisk (/*) at the start of a line

SPSS Syntax Color Coding

By default, SPSS uses color and bolding to indicate the roles of the words in the syntax.

Dark blue/purple Procedure names; execution statements

Green Statements associated with the given procedure

Dark red/orange Option keywords

Gray Comments

Black Variable names; other text

When should I use syntax?

The biggest benefits of using SPSS syntax are reproducibility and communication.

REPRODUCIBILITY

1. It is far easier to "retrace your steps" with syntax if you need to recall the modifications you've made to your dataset. It's also far easier to find and correct mistakes in your analysis if you have the syntax, as opposed to if you had used the menus.
2. Syntax can easily be modified and re-used on future projects, saving you time and effort in the long run.

When should I use syntax? - Communication

- a) It is much easier to communicate to others what actions you performed in SPSS by **showing someone your syntax** than it is to describe how you used the menus.
- b) The content of the menus and dialog windows has changed over time with each version of SPSS, while the syntax has remained relatively unchanged.
- c) In general, if you are working on a major project (like a thesis, dissertation, or research for publication), or if you are collaborating with others on data analysis, we strongly recommend using SPSS syntax.

EXAMPLE

sample SPSS
script for basic
calculations

*basic statistical characteristics

```
DESCRIPTIVES VARIABLES=Points v100_m Long_jump Shot_put High_jump v400_m v110_m_hurdles  
Discus_Throw Pole_vault Javelin v1500_m  
/STATISTICS=MEAN SUM STDDEV RANGE MIN MAX SEMEAN.
```

*

* tests of normality One-Sample Kolmogorov-Smirnov Test

```
NPART TESTS
```

```
/K-S(NORMAL)=Points v100_m Long_jump Shot_put High_jump v400_m v110_m_hurdles Discus_Throw  
Pole_vault Javelin v1500_m  
/MISSING ANALYSIS  
/KS_SIM CIN(99) SAMPLES(10000).
```

*

* calculation of correlation coefficients

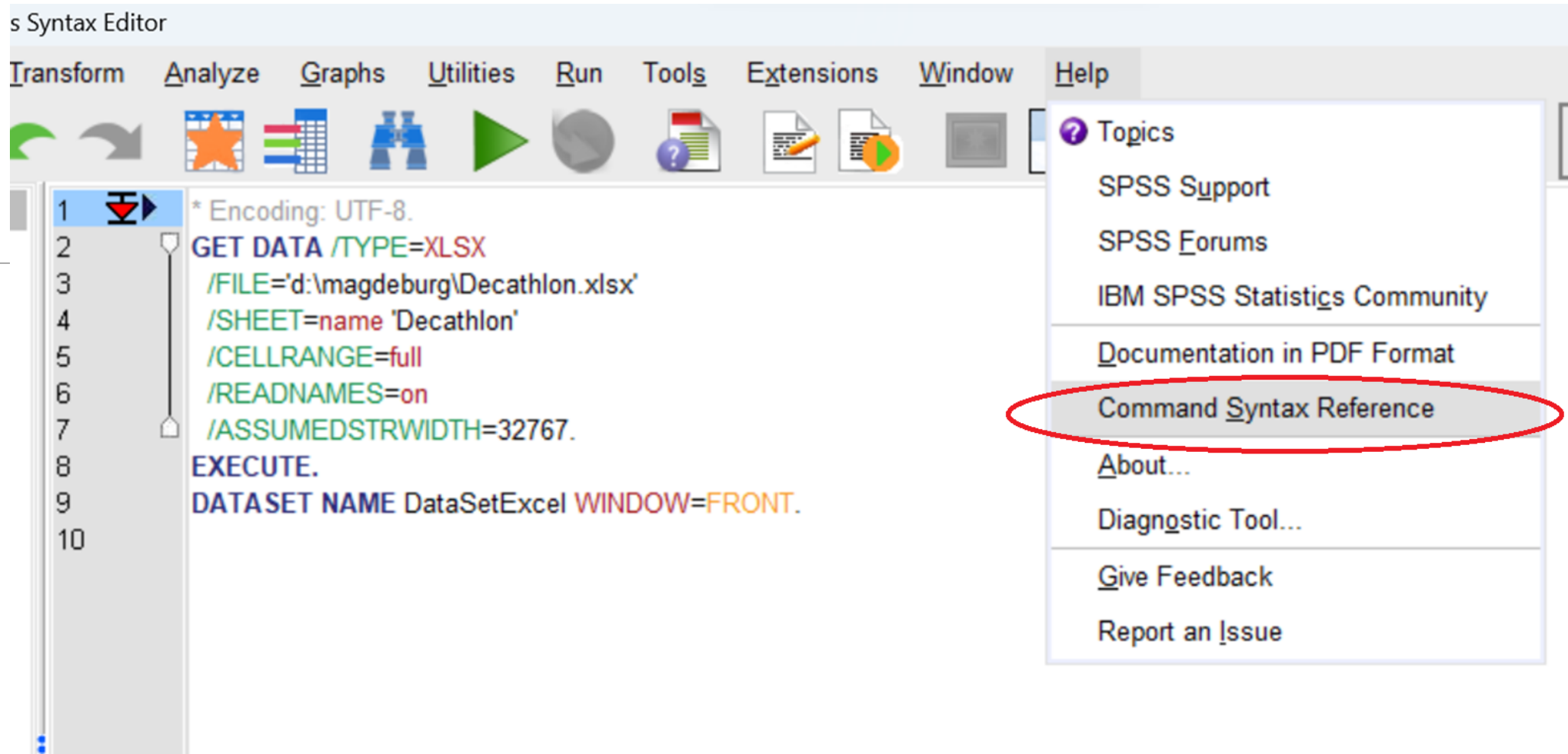
```
STATS CORRELATIONS VARIABLES=Points v100_m Long_jump Shot_put High_jump v400_m v110_m_hurdles  
Discus_Throw Pole_vault Javelin v1500_m  
/OPTIONS CONFLEVEL=95 METHOD=FISHER  
/MISSING EXCLUDE=YES PAIRWISE=YES.
```

*

* t-test between 2 groups, together with effect size

```
T-TEST GROUPS=Finished(1 2)  
/MISSING=ANALYSIS  
/VARIABLES=v100_m Long_jump  
/ES DISPLAY(TRUE)  
/CRITERIA=CI(.95).
```

HELP



Help > Command Syntax Reference. (This will open a PDF containing the reference manual.)

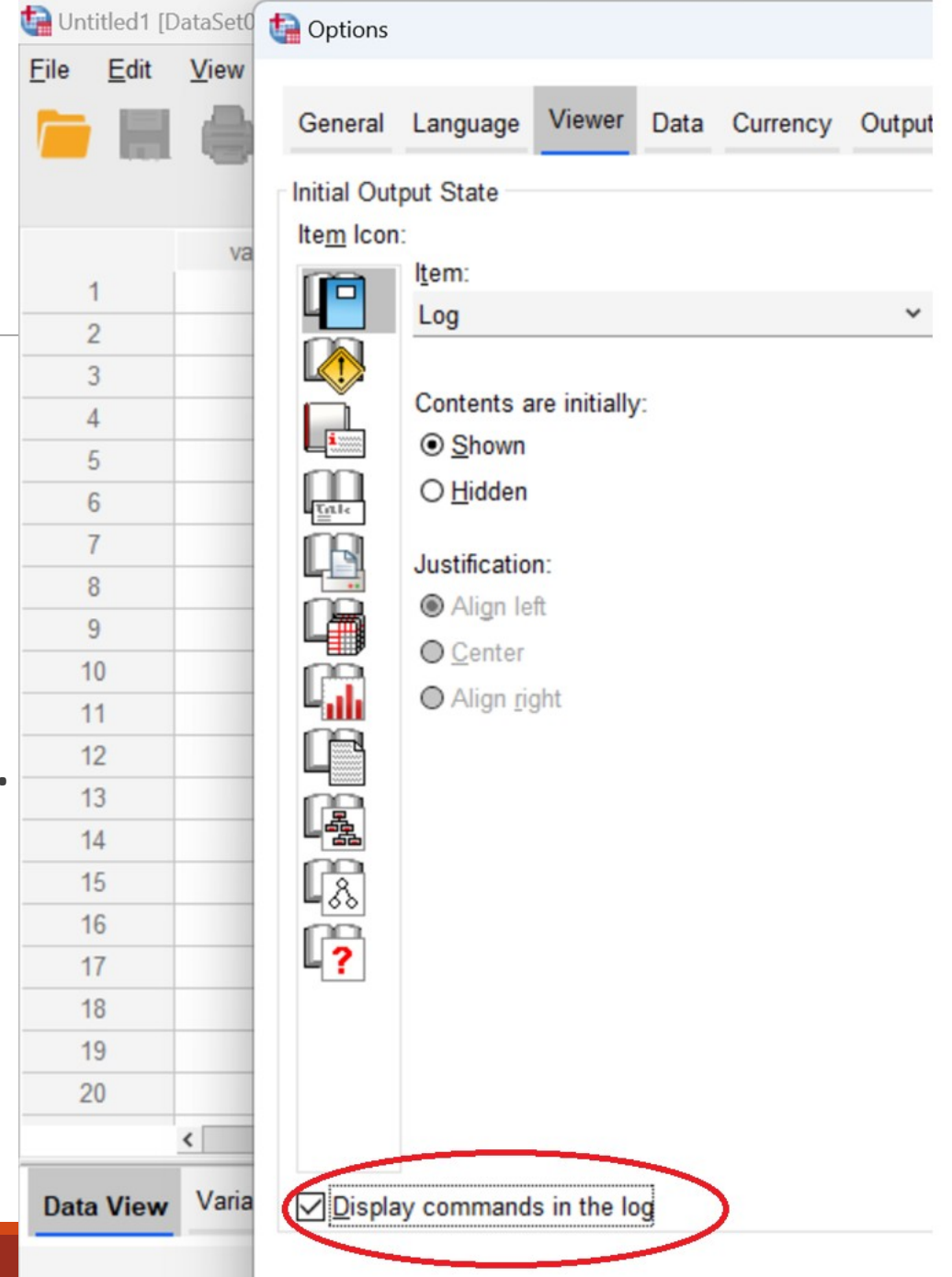
How to Make Syntax Print in the Output Window

Click **Edit > Options**.

Click the **Viewer** tab.

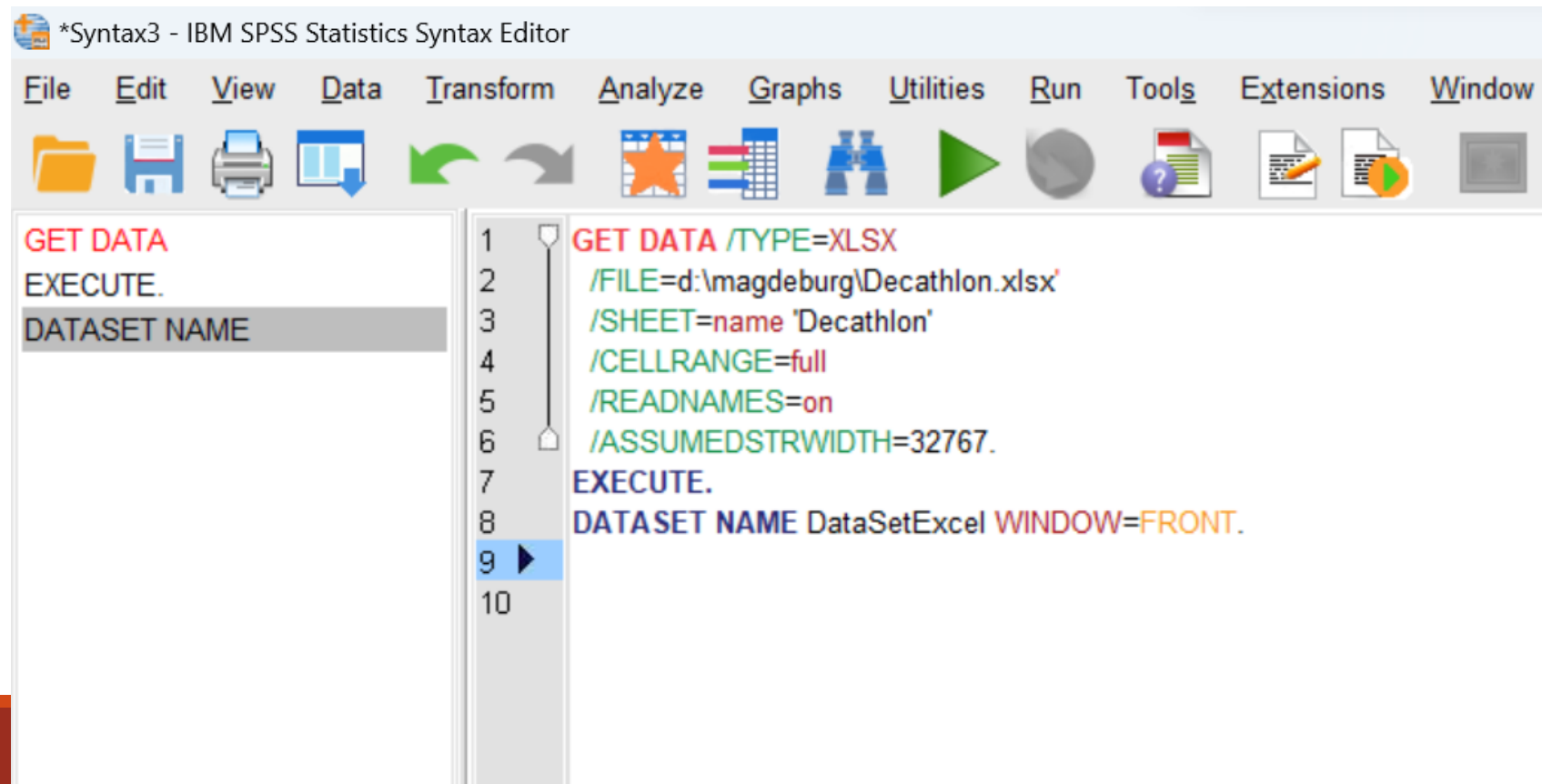
Check the box for **Display commands in log**.

Click **OK** when finished.



OPENING THE SYNTAX EDITOR

To open a new Syntax Editor window, click **File > New > Syntax**.



Files in SPSS

What are the 3 types of files in SPSS?

There are three types of SPSS files that we will use during this class:

1. data files, which end in . sav;
2. syntax files, which end in . sps; and
3. output files, which end in . spv.

Importing data

How the data should look for import?

1. In the first row of column are labels,
2. from second row you have a data

D	E	F	G	H
Points	100 m	Long jump	Shot put	High jump
5456	11,29		14,3	
746	11,53			
3056	11,05	7,05	12,84	1
5737	10,94	7,43	13,77	1
1631	11,05	6,86		
2480	11,05	7,12	14,98	
3936	10,79	7,04	13,32	1
6182	10,8	7,38	16,17	2
1668	10,77		14,45	
8021	10,55	7,34	14,44	1
8067	11,14	6,99	14,91	1
8225	10,69	7,48	14,8	2
7404	11,36	6,68	14,92	1
8820	10,44	7,96	15,23	2
6543	11,28	7,2	13,04	1
7926	10,87	7,38	13,07	1
7995	11,89	7,99	14,57	1

Importing Data from an Excel File

```
GET DATA /TYPE=XLSX  
/FILE='C:\path\to\file.xlsx'  
/SHEET=name 'Name-of-Sheet'  
/CELLRANGE=full  
/READNAMES=on  
/ASSUMEDSTRWIDTH=32767.  
EXECUTE.  
DATASET NAME DataSetExcel WINDOW=FRONT.
```

Note: If you are importing an *.xlsx file, use /TYPE=XLSX. If you are reading an *.xls file, use /TYPE=XLS.

The most important lines in this code are /FILE="" and /SHEET=name "".

```
GET DATA /TYPE=XLSX  
/FILE='C:\path\to\file.xlsx'  
/SHEET=name 'Name-of-Sheet'  
/CELLRANGE=full  
/READNAMES=on  
/ASSUMEDSTRWIDTH=32767.  
EXECUTE.  
DATASET NAME DataSetExcel WINDOW=FRONT.
```


Importing Data from an Text File(txt, csv)

```
GET DATA /TYPE=XLSX
```

```
/FILE='C:\path\to\file.xlsx'
```

```
/SHEET=name 'Name-of-Sheet'
```

```
/CELLRANGE=full
```

```
/READNAMES=on
```

```
/ASSUMEDSTRWIDTH=32767.
```

```
EXECUTE.
```

```
DATASET NAME DataSetExcel WINDOW=FRONT.
```

Note: If you are importing an *.xlsx file, use /TYPE=XLSX. If you are reading an *.xls file, use /TYPE=XLS.

The most important lines in this code are /FILE="" and /SHEET=name "".

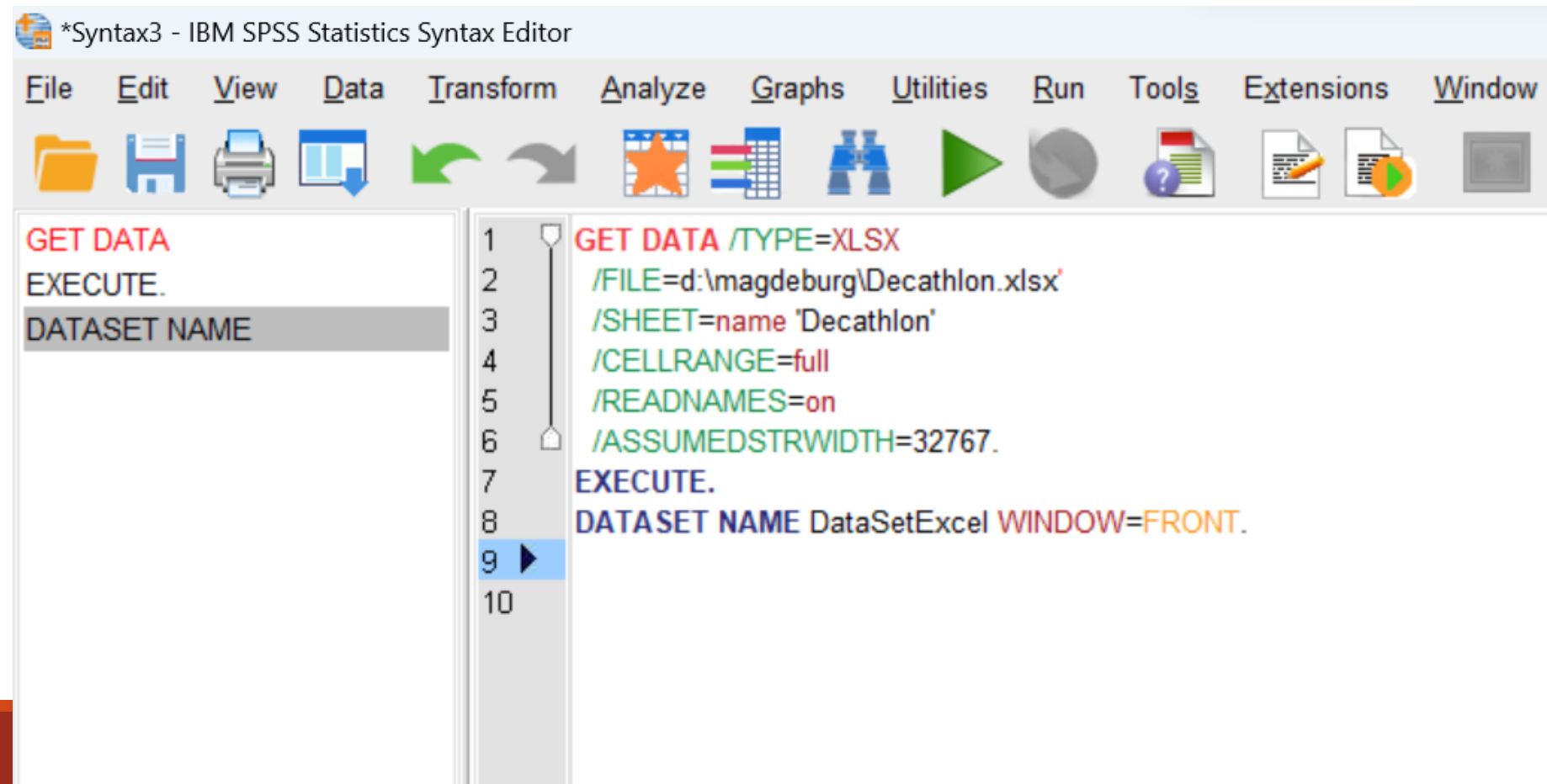
```
GET DATA /TYPE=XLSX
/FILE='C:\path\to\file.xlsx'
/SHEET=name 'Name-of-Sheet'
/CELLRANGE=full
/READNAMES=on
/ASSUMEDSTRWIDTH=32767.
EXECUTE.
DATASET NAME DataSetExcel WINDOW=FRONT.
```

Importing - my experience 😊

1. **Don't leave data import to automatic recognition.**
2. Of all statics, the most important thing is the correct data preparation, do it carefully (in SPSS, in Excel, ...) do a standard import.
3. Clear the data...
4.

Opening the syntax editor

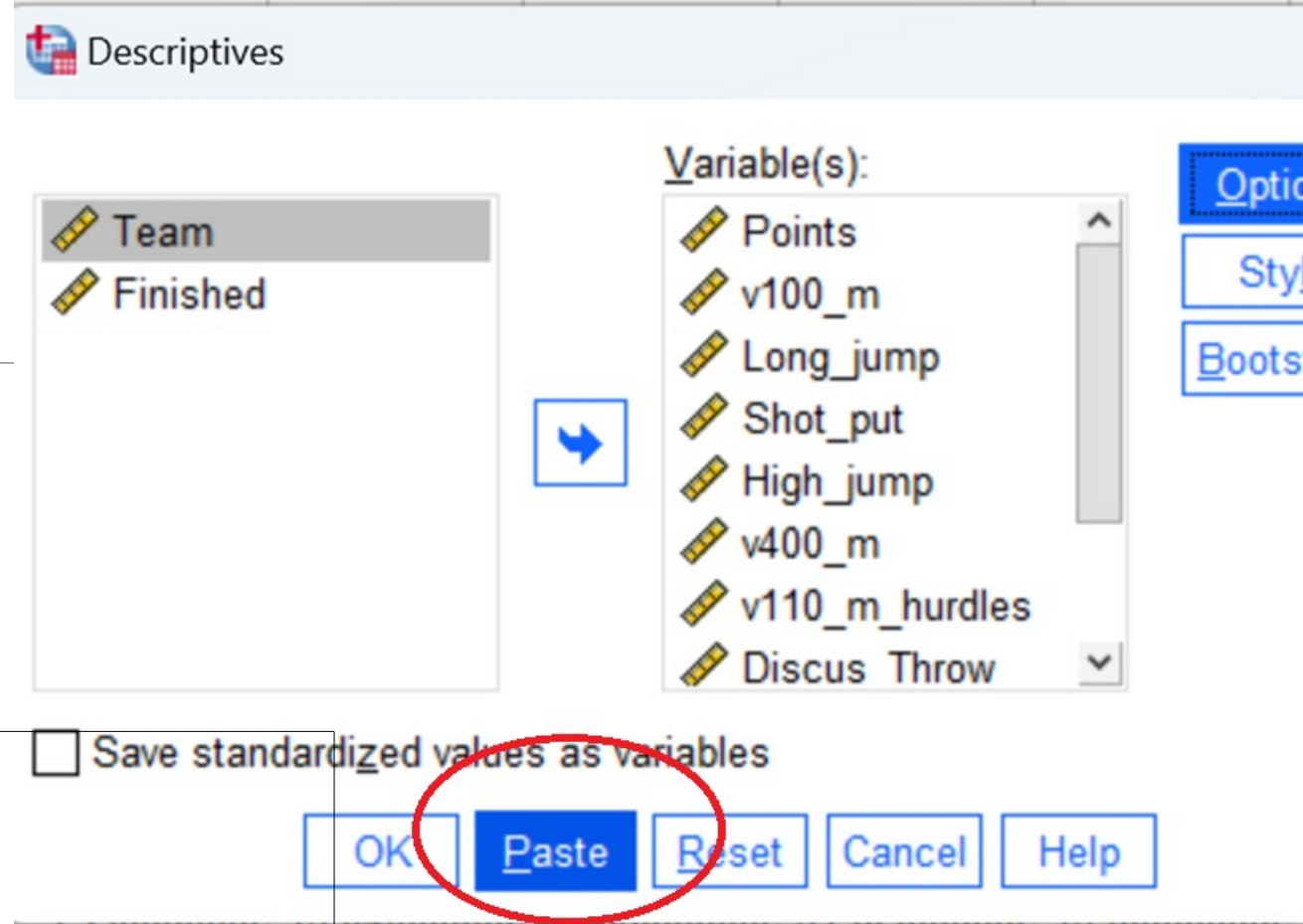
To open a new Syntax Editor window, click **File > New > Syntax**.



Syntax editor

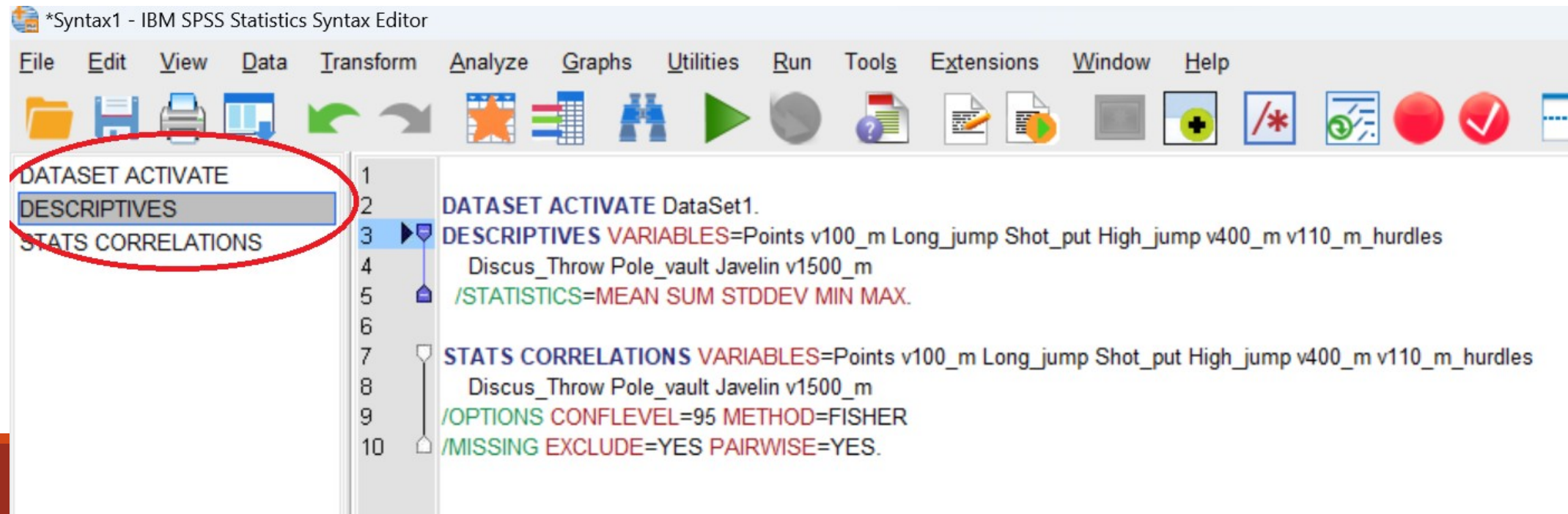
You can paste the syntax code of every analysis you perform into a syntax editor window by clicking on Paste at the bottom of an analysis dialogue box.

Syntax code will be pasted.



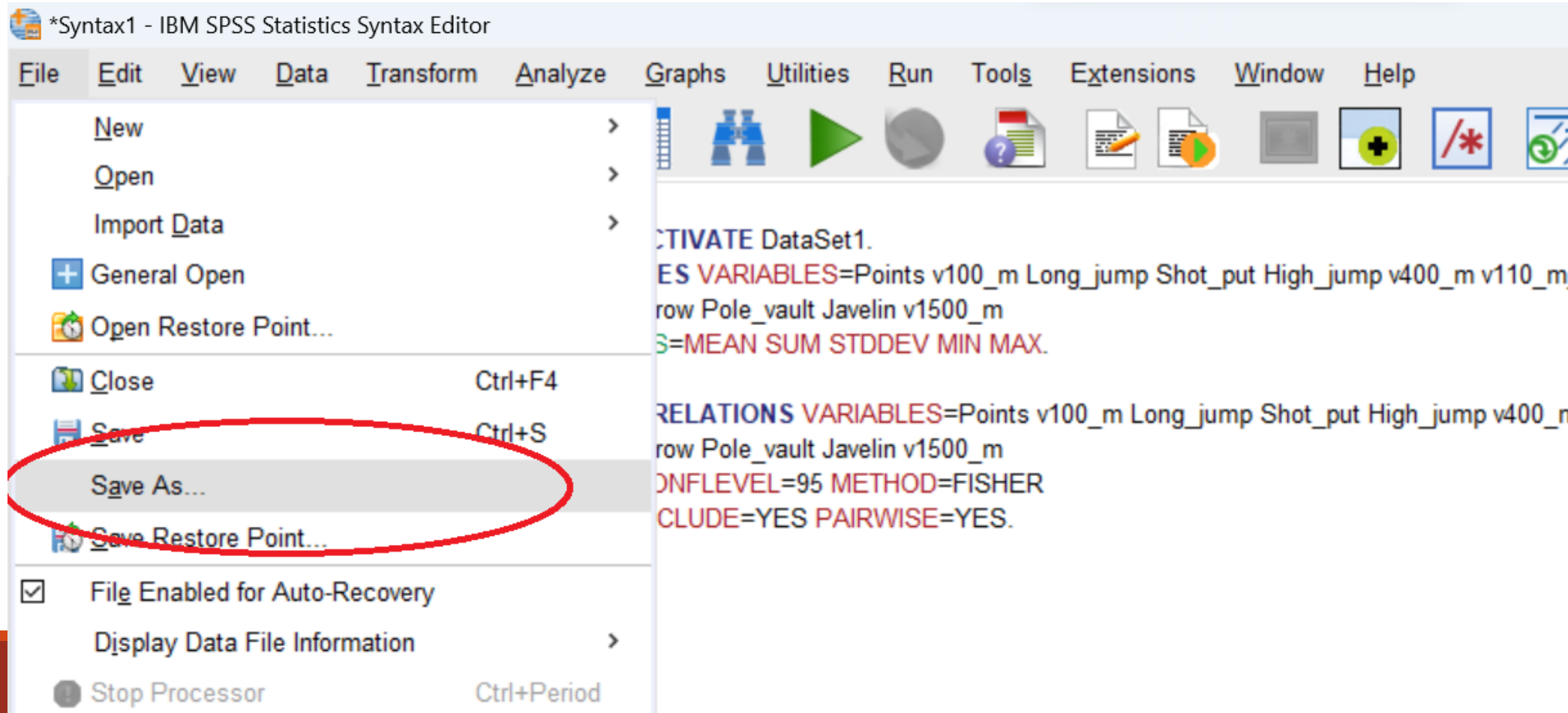
Syntax editor

The left panel of the Syntax Editor window shows an outline of the commands in your syntax, and can be used to navigate within your code. You can jump to a specific part of your code by clicking on the command in the left panel.



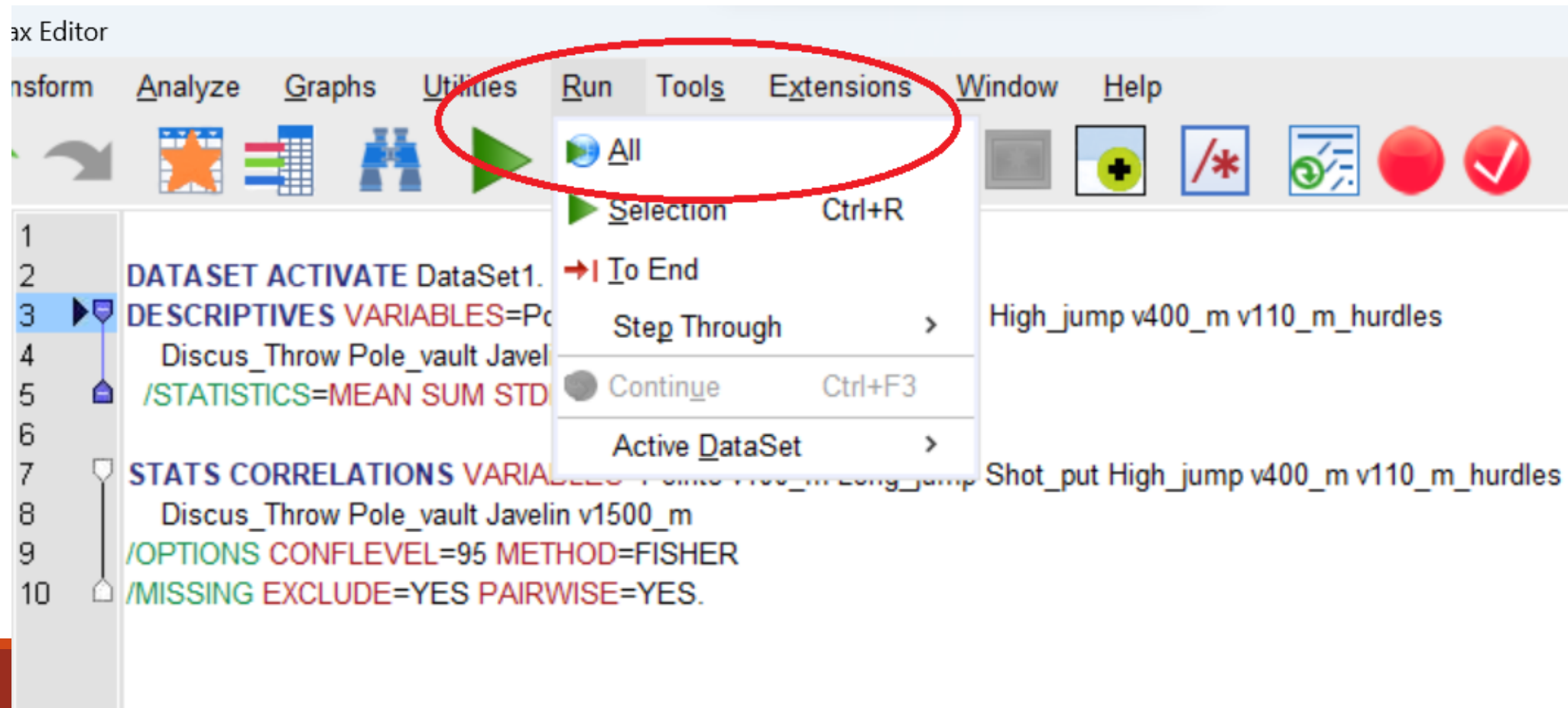
Syntax editor

Syntax can be saved as an *.sps file.



How to execute

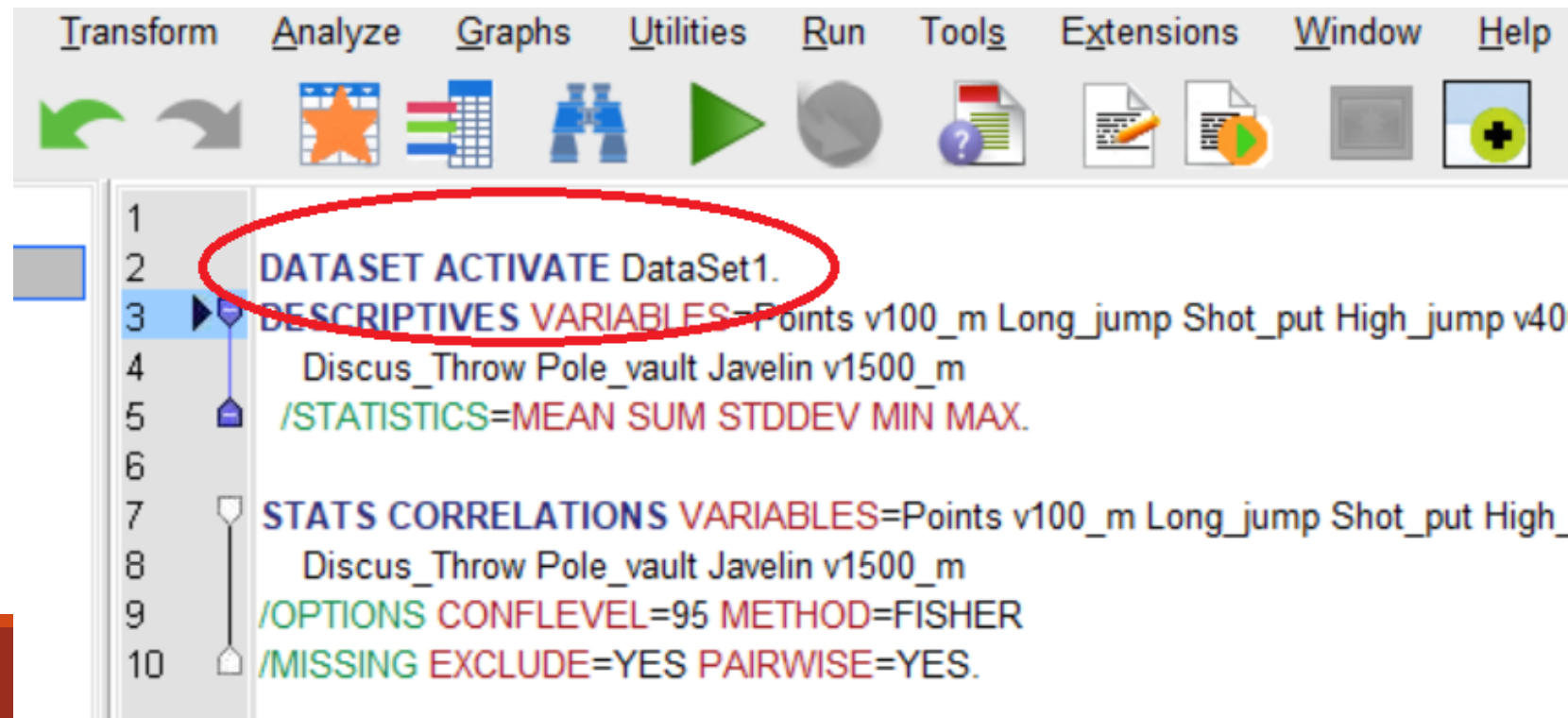
Clicking **Run > All** will execute the procedure



Making sense of SPSS syntax

The Dataset Activate command

By default, when SPSS Statistics opens a new data file it labels it as 'DataSet1'. When it opens a second data file, it labels it as 'DataSet2' and continues add a sequential number to each new data file that is opened



```
11  
12 FREQUENCIES VARIABLES=v100_m Long_jump Shot_put  
13 /ORDER=ANALYSIS.  
14 ▶
```

The screenshot shows the SPSS syntax editor with the following command: `FREQUENCIES VARIABLES=v100_m Long_jump Shot_put /ORDER=ANALYSIS.` Red lines and numbers 1 through 5 are used to highlight specific parts of the command: 1 points to the start of the command, 2 points to the `VARIABLES` keyword, 3 points to the list of variables, 4 points to the slash before the subcommand, and 5 points to the `ANALYSIS` keyword.

1. As with all SPSS syntax procedures it starts with a single Command: **FREQUENCIES**
2. This is followed by an optional Keyword: **VARIABLES**
3. It then specifies 3 variables: **v100m, Long_jump, Shot_put**
4. This is followed by a slash with a Subcommand and another Keyword: **/ORDER=ANALYSIS**
5. Finally the procedure is completed with a single period character:

A few things to note

SPSS Syntax is remarkably unfussy about how it's typed:

- a) It's generally not case sensitive
- b) You can use as many lines as you want to specify a single command.
- c) It normally doesn't require correct indentation to run.

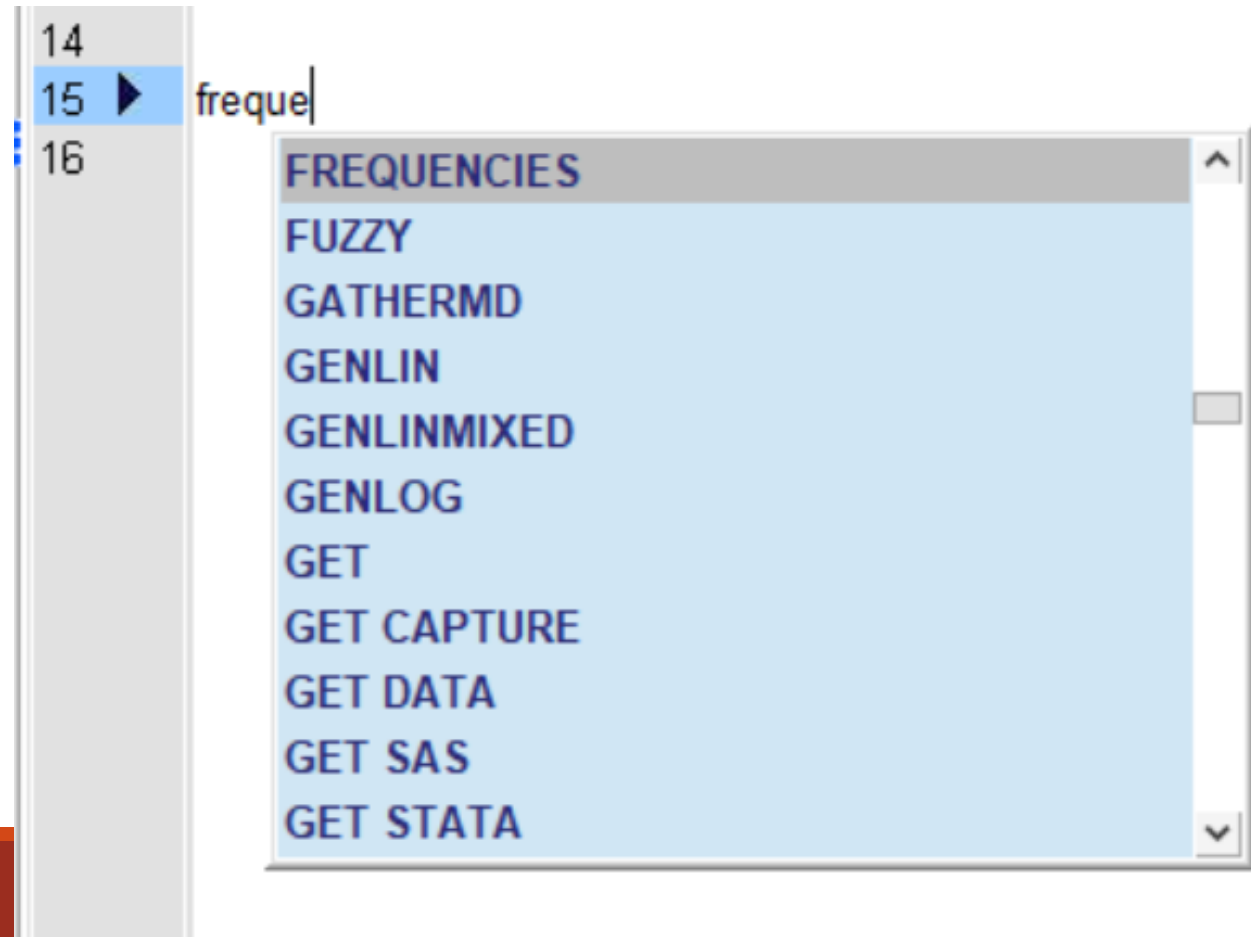
A few things to note

However there are some basic rules:

- a) Each new command (e.g. Frequencies) must start on a new line.
- b) Each procedure must end with a period as a 'command terminator'.
- c) Most subcommands are separated by slashes (/). The slash before the first subcommand on a command is usually optional

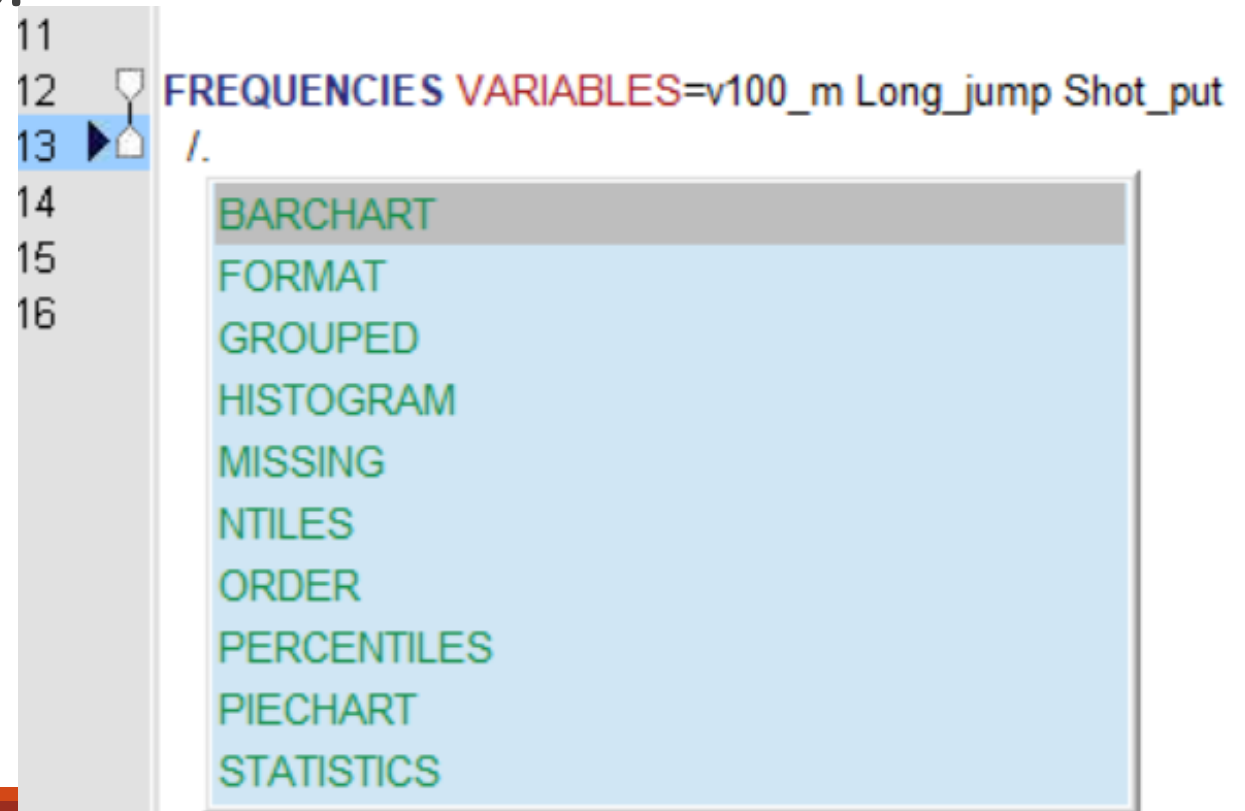
Getting help with syntax

Notice the useful autocomplete function works when manually typing syntax.



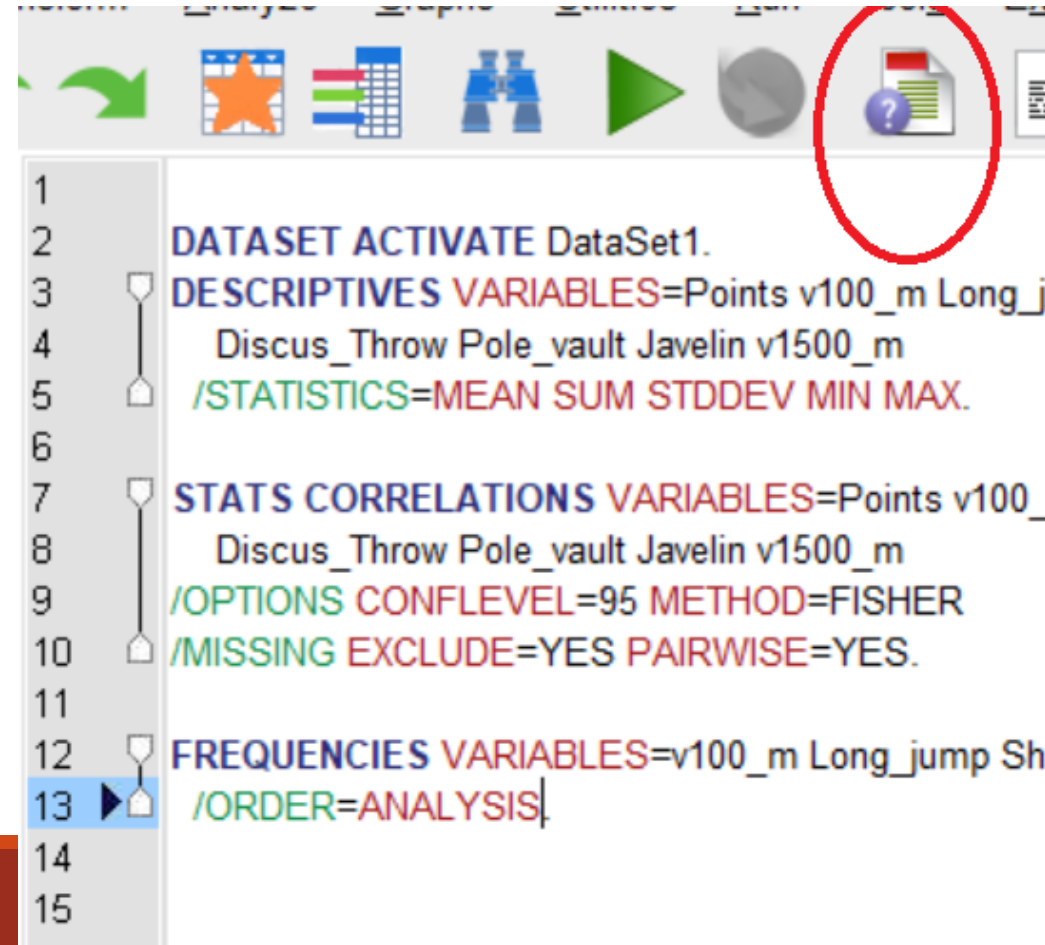
Getting help with syntax

For instance, if you add a forward slash after a command, a list of subcommands appears.



Getting help with syntax

A really useful feature is the ability to quickly look-up the syntax diagram for a procedure just by highlighting the command word and clicking the Syntax Help button on the toolbar of an open Syntax window.



Examples

Remember that you can always **create syntax** simply by clicking the **Paste button** in an SPSS dialog

- However, in the next few examples we will delve into aspects of syntax that illustrate how different procedures work or reveal functions that aren't obvious to the casual user
- To begin with let's look at opening data in IBM SPSS Statistics

Opening data files

The GET FILE command is used to import an SPSS data file (*.sav).

- The GET DATA command however is used to import non-native file formats such as Excel workbooks or comma separated (CSV) files.
- Note the use of the /TYPE subcommand that specifies the file type to be imported

```
GET DATA
```

```
/TYPE=XLSX
```

```
/FILE='D:\Magdeburg\Decathlon.xlsx '
```

Filtering data with the Select Cases dialog

Using the paste button in the Select Cases dialog tends to generate surprisingly complex syntax

* Filtr creating.

USE ALL.

```
COMPUTE filter_$=(Finished = 0).
```

```
VARIABLE LABELS filter_$ 'Finished = 1 (FILTER)'.|
```

```
VALUE LABELS filter_$ 0 'Not Selected' 1 'Selected'.
```

```
FORMATS filter_$ (f1.0).
```

```
FILTER BY filter_$.
```

```
EXECUTE.
```

* FREQUENCIES pro Long_jump with using filter.

```
FREQUENCIES VARIABLES=Long_jump
```

```
/ORDER=ANALYSIS.
```

Select Cases

Select

All cases

If condition is satisfied

Finished = 1

Random sample of cases

Based on time or case range

Use filter variable:

Output

Filter out unselected cases

Copy selected cases to a new dataset

Dataset name:

Delete unselected cases

Current Status: Do not filter cases

Useful Syntax Window Functions

- Breakpoints: Breakpoints are used to stop command syntax execution at positions so you can check the results at each stage before moving on.
- Bookmarks: Creating bookmarks allow you to more easily navigate large command syntax files.
- Step Through: You can step through command syntax one command at a time.
- Split view: From the main menu click Window > Split View. This splits the syntax editor window into two panes so that you can check one part of the syntax file without losing focus on another.

Useful Syntax Window functions

Break point

Bookmark

Line Numbers

Error Pane

Split window

The screenshot shows a syntax window with the following content:

```
1 * Encoding: windows-1252.  
2 *Open the data file.  
3 GET FILE='C:\Temp\Employee data with age.sav'.  
4 DATASET NAME $DataSet WINDOW=FRONT.  
5  
6 FREQUENCIES VARIABLES=gender jobcat minority  
7 /ORDER=ANALYSIS.  
8  
9 DESCRIPTIVES VARIABLES=age educ salary salbegin  
10 /STATISTICS=MEAN STDDEV MIN MAX.  
11  
12 CROSSTABS  
13 /TABLES=gender minority BY jobcat  
14 /FORMAT=AVALUE TABLES  
15 /CELLS=COUNT ROW  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39 OUTPUT MODIFY  
40 /SELECT ALL EXCEPT (TABLES CHARTS)  
41 /DELETEOBJECT DELETE = YES.  
42  
43 OUTPUT MODIFY  
44 /SELECT TABLES  
45 /IF SUBTYPES=["Frequencies"]  
46 /TABLECELLS SELECT=["Total"]  
47 SELECTDIMENSION=ROWS  
48 STYLE=BOLD APPLYTO=ROW.  
49  
50  
51
```

The Error Pane at the bottom contains the following information:

Line	Command	Information
1	GET FILE	The document is already in use by another user or process. If you make changes to the document they may overwrite changes made by others or your changes may be overwritten by others.
3	Dataset Name	The active dataset will replace the existing dataset named \$DataSet.

**Without syntax,
SPSS would be
just another
beautiful GUI
without a soul**

25 examples 😊

1. Add_99AtEndOfAllVariableNames.sps
2. AreAllWordsPresent.sps
3. BarChartsForSchoolTypesBySexWherePercentagesOfEachSexAddUpTo100Percent.sps
4. CalculateAge.sps
5. CalculateChi-squareSigGiven_q_and_df.sps
6. CalculateMcNemarChi-SquareTest.sps
7. CombineStringAndNumber.sps
8. CountOutliers.sps
9. decathlon factorial.sps
10. ExampleOfSurfacePlot.sps
11. ExampleWorkingWithMatrices.sps
12. ExcludeOutliersDefinedAsMeanPlusMinus2SD.sps
13. FAWithSpearmanCorrelation.sps
14. FindCubicRoot.sps
15. FindDuplicates.sps
16. FindInverseOfAMatrix.sps
17. FlagCasesWhereSalaryIsInTop95percentile.sps
18. HowToSpeedUpIGRAPH.sps
19. ChangeAllStringsToLowerCase.sps
20. LogisticRegressionByMacro.sps
21. NewVarEqualsMeanOfAnotherVar.sps
22. ParseStringIntoManyVariables.sps
23. PopulationPyramid.sps
24. PrintMeanPlusMinusStddevInTable.sps
25. ReverseDigitsOfAnInteger.sps