



Wolfram Mathematica

Интерактивные вычисления и визуализация

Таранчук Валерий Борисович

Учебные материалы, рекомендации пользователям СКА Mathematica, обучающие примеры и упражнения.

Оригинал документа создан и демонстрируется в NB, конспект предоставляется студентам в формате PDF, учебные материалы размещены на соответствующей странице курса в LMS Moodle.

Уважаемые читатели. В сгруппированных секциях ниже (и везде далее в подобных) размещён материал для демонстраций и обсуждения на лекциях. Такой материал каждый может подготовить сам (например, по записям на лекции) или, найдя аналогичный в других электронных ресурсах, книгах.

► Содержание учебных материалов по темам лекций, заданиям практических занятий, имена файлов

Основные конструкции и операции языка Wolfram Language.

Работа с файлами, прием и обработка данных и изображений; экспорт и импорт данных, формул, графики.

▼ План лекции

Повторение пройденного – ... манипулировать выражениями с интерактивностью в расчетах ...

Повторение пройденного – Части выражений

Импорт, экспорт данных, результатов, графиков

Примеры извлечения данных из удаленных серверов

О графических объектах системы Mathematica

Типы (категории) графических объектов Mathematica. КСР

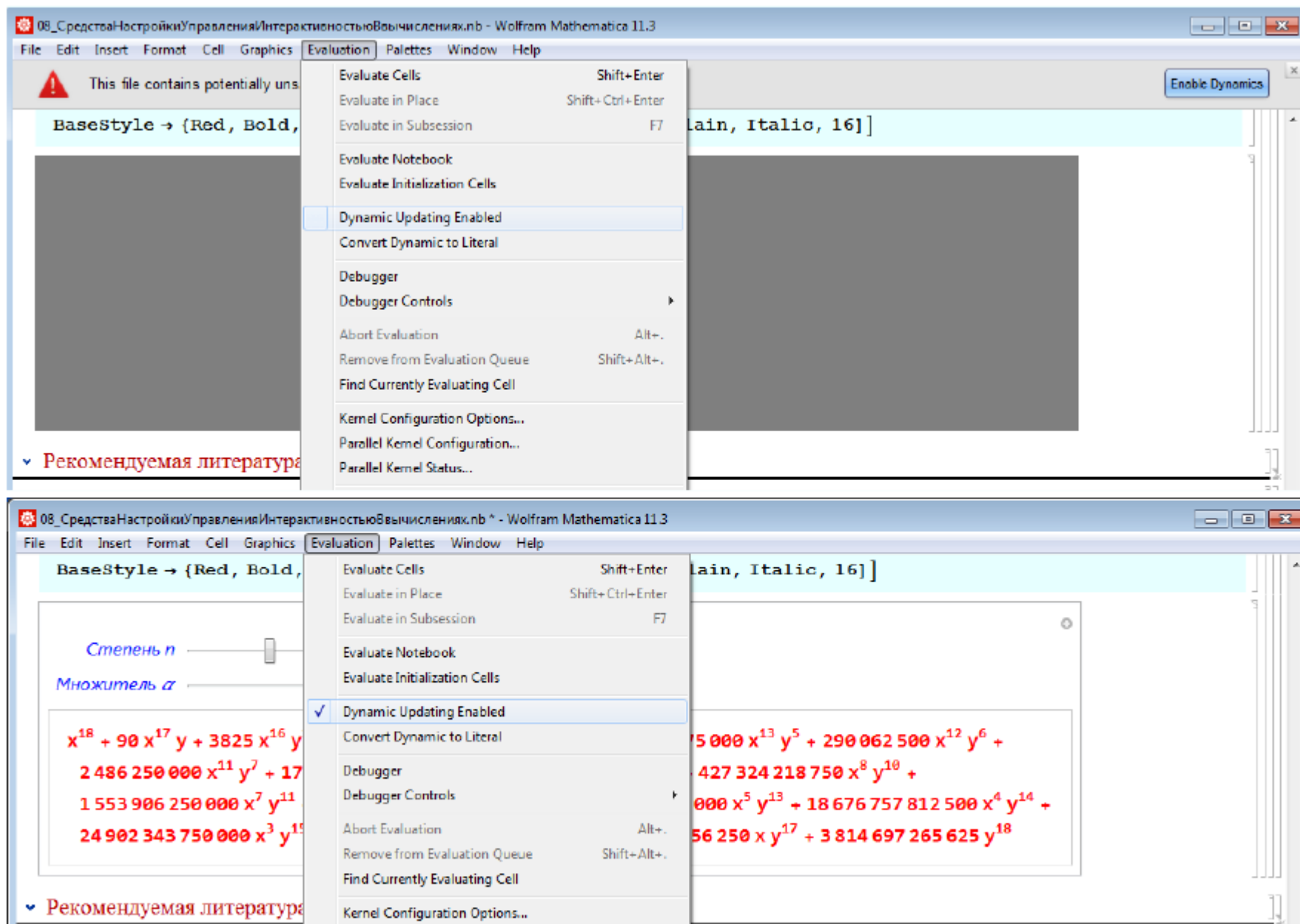
▼ Повторение пройденного – ... манипулировать выражениями с интерактивностью в расчетах ...

▼ Как манипулировать выражениями с интерактивностью в расчетах, уточнять интерфейс.

(* j) Предыдущий, но другие начальные значения *)

```
Manipulate[Expand[(x + α y)^n], {{n, 18, "Степень n"}, 0, 50, 1, Appearance → "Labeled"},
  {{α, 5, "Множитель α"}, -3, 5, 0.1, Appearance → "Labeled"},
  BaseStyle → {Red, Bold, 18}, LabelStyle → Directive[Blue, Plain, Italic, 16]]
```

Чтобы у Вас “работала” интерактивность ... Рекомендации на иллюстрациях ниже:



Задание. Проверьте установки собственного блокнота. Убедитесь, что анимация подключена. Для контроля/упражнений

скачайте, например, на сайте **WOLFRAM Demonstrations Project** исходный текст презентации **Cutoff Parallelepipeds ...** Освойте инструменты и ответьте на вопросы типа перечисленных ниже:

- ✓ кто автор/разработчик презентации,
- ✓ где, в каком университете/центре/компании реализован проект,
- ✓ найдите в исходном коде, как реализованы варианты оформления – например, в чем различия при выводе полей `translate` и `width`.

WDP_Gauss..., WDP_Making..., код доступа 04251113

▼ Повторение пройденного – Части выражений

▼ Части выражений. Основные функции

Сложные выражения состоят из частей. Для выделения любой заданной части выражений используются функция *Part* или двойные квадратные скобки, 0-ая часть – тип, то что возвращает `Head`. Более подробно об этой функции будет изложение при изучении правил работы со списками, здесь отметим только основные возможности.

- ✓ `Head[expr]` – возвращает тип, нулевая часть (0-ая часть);
- ✓ `Part[expr,m]` выделяет *m*-ую часть выражения (или списка) с начала; если *m* отрицательно, позиция отсчитывается с конца, эквивалентные записи: `expr[[m]]` (две подряд квадратные скобки), `expr[...]` (двойные квадратные скобки);
- ✓ `Part[expr,n1;;n2]` – части выражения (или списка) от *n*₁ до *n*₂;
- ✓ `First[expr]` – возвращает первый элемент в выражении (или в списке);
- ✓ `Last[expr]` – возвращает последний элемент в выражении (или в списке);
- ✓ `Numerator[expr]` – возвращает числитель выражения или указанной части;
- ✓ `Denominator[expr]` – возвращает знаменатель выражения или указанной части.

Задание. Прочитайте, составьте и выполните собственные упражнения аналогичные рассмотренным в:

- ✓ <https://mathematica.stackexchange.com/questions/96610/change-some-elements-at-the-second-level-of-a-list-based-on-some-conditions>,
- ✓ <https://mathematica.stackexchange.com/questions/84908/how-to-apply-replaceall-to-denominator>,
- ✓ <https://mathematica.stackexchange.com/questions/214605/how-to-specify-part-in-at-mixed-levels>

▼ Списки. Примеры использования `Count` с `levelspec`

Задание. Составьте и выполните собственные упражнения аналогичные рассмотренным в:

✓ Count[list,pattern,levelspec] – на указанном уровне levelspec – конкретно на levelspec

Акценты: примеры на различие Count[{a, ..., x_..., 3], Count[{a, ..., {3}]

Part, ..mera..., ..omin..., код доступа 10232504

▼ Импорт, экспорт данных, результатов, графиков

▼ Поддерживаемые форматы для экспорта и импорта данных

Система *Mathematica* поддерживает все широко распространенные форматы и обеспечивает экспорт и импорт данных, программных модулей. Полный список форматов обмена можно посмотреть следующим образом (в списке более 170 расширений имен):

`$ImportFormats`

```
{3DS, ACO, Affymetrix, AgilentMicroarray, AIFF, ApacheLog, ArcGRID, AU, AVI, Base64, BDF, Binary, Bit, BMP, BSON,
Byte, BYU, BZIP2, CDED, CDF, Character16, Character8, CIF, Complex128, Complex256, Complex64, CSV, CUR,
DAE, DBF, DICOM, DIF, DIMACS, Directory, DOT, DXF, EDF, EML, EPS, ExpressionJSON, ExpressionML, FASTA,
FASTQ, FCS, FITS, FLAC, GenBank, GeoJSON, GeoTIFF, GIF, GPX, Graph6, Graphlet, GraphML, GRIB, GTOP030,
GXL, GZIP, HarwellBoeing, HDF, HDF5, HIN, HTML, HTTPRequest, HTTPResponse, ICC, ICNS, ICO, ICS, Ini,
Integer128, Integer16, Integer24, Integer32, Integer64, Integer8, JavaProperties, JavaScriptExpression,
JCAMP-DX, JPEG, JPEG2000, JSON, J VX, KML, LaTeX, LEDA, List, LWO, M4A, MAT, MathML, MBOX, MCTT, MDB, MESH,
MGF, MIDI, MMCIF, MO, MOL, MOL2, MP3, MPS, MTP, MTX, MX, MXNet, NASACDF, NB, NDK, NetCDF, NEXUS, NOFF, OBJ,
ODS, OFF, OGG, OpenEXR, Package, Pajek, PBM, PCAP, PCX, PDB, PDF, PGM, PHPIni, PLY, PNG, PNM, PPM, PXR,
PythonExpression, QuickTime, Raw, RawBitmap, RawJSON, Real128, Real32, Real64, RIB, RLE, RSS, RTF, SCT,
SDF, SDTS, SDTSDM, SFF, SHP, SMA, SME, SMILES, SND, SP3, Sparse6, STL, String, SurferGrid, SXC, Table, TAR,
TerminatedString, TeX, Text, TGA, TGF, TIFF, TIGER, TLE, TSV, UBJSON, UnsignedInteger128, UnsignedInteger16,
UnsignedInteger24, UnsignedInteger32, UnsignedInteger64, UnsignedInteger8, USGSDEM, UUE, VCF, VCS, VTK,
WARC, WAV, Wave64, WDX, WebP, WNet, WMLF, WXF, XBM, XHTML, XHTMLMathML, XLS, XLSX, XML, XPORT, XYZ, ZIP}
```


\$ExportFormats

```
{3DS, ACO, AIFF, AU, AVI, Base64, Binary, Bit, BMP, BSON, Byte, BYU, BZIP2, C, CDF, Character16, Character8,
Complex128, Complex256, Complex64, CSV, CUR, DAE, DICOM, DIF, DIMACS, DOT, DXF, EMF, EPS, ExpressionJSON,
ExpressionML, FASTA, FASTQ, FCS, FITS, FLAC, FLV, FMU, GeoJSON, GIF, Graph6, Graphlet, GraphML, GXL, GZIP,
HarwellBoeing, HDF, HDF5, HTML, HTMLFragment, HTTPRequest, HTTPResponse, ICNS, ICO, Ini, Integer128,
Integer16, Integer24, Integer32, Integer64, Integer8, JavaProperties, JavaScriptExpression, JPEG,
JPEG2000, JSON, J VX, KML, LEDA, List, LWO, M4A, MAT, MathML, Maya, MCTT, MGF, MIDI, MO, MOL, MOL2, MP3,
MTX, MX, MXNet, NASACDF, NB, NetCDF, NEXUS, NOFF, OBJ, OFF, OGG, Package, Pajek, PBM, PCX, PDB, PDF,
PGM, PHPIni, PLY, PNG, PNM, POV, PPM, PXR, PythonExpression, QuickTime, RawBitmap, RawJSON, Real128,
Real32, Real64, RIB, RTF, SCT, SDF, SMA, SND, Sparse6, STL, String, SurferGrid, SVG, SWF, Table, TAR,
TerminatedString, TeX, TeXFragment, Text, TGA, TGF, TIFF, TSV, UBJSON, UnsignedInteger128, UnsignedInteger16,
UnsignedInteger24, UnsignedInteger32, UnsignedInteger64, UnsignedInteger8, UAE, VideoFrames, VRML, VTK,
WAV, Wave64, WDX, WebP, WNet, WMF, WMLF, WXF, X3D, XBM, XHTML, XHTMLMathML, XLS, XLSX, XML, XYZ, ZIP, ZPR}
```

▼ Импорт файлов Excel

Для импорта данных в блокнот используется функция `Import[]`, при этом необходимо указать каталог. Чаще всего импортируются файлы формата `.xlsx` и `.xls`. По умолчанию импортируется вся книга; можно указать (имя или номер) конкретного листа, вывести формулы, используемые на указанном листе, вывести указанные числа из ячеек указанного столбца. Примеры (можно вписывать номер или имя листа):

- ✓ Откроем файл `IVT_12_xlsx.xlsx` (2 варианта записи)

```
SystemOpen["T://PedProc/IVT/IVT_12_xlsx.xlsx"]
```

Указываем каталог, принимаемый файл:

```
SetDirectory[NotebookDirectory[]];
Import["IVT_12_xlsx.xlsx"]
```

Указываем каталог, принимаемый файл и лист:

```
Import["IVT_12_xlsx.xlsx", {"Data", 2}]  
Import["IVT_12_xlsx.xlsx", {"Data", "Second Sheet"}]
```

Вывод формул с указанного листа обеспечивает функция Import с конкретными уточнениями ({“Formulas”,3}). Также на рисунке 12.1 даны 2 фрагмента скриншотов листа Excel, которые иллюстрируют идентичность источника и результата импорта:

```
SetDirectory[NotebookDirectory[]];  
Import["IVT_12_xlsx.xlsx", {"Formulas", 3}]
```

	A	B	C
1	x-data	y-data	
2	1	8	
3	2	24	
4	3	50	
5	4	86	
6	5	132	
7	6	188	
8	7	254	
9	8	330	
10	9	416	
11	10	512	
12	11	618	
13	12	734	
14	13	860	
15	14	996	
16	15	1142	

Рис. 12.1. Фрагменты листа Excel с формулами

Конкретные столбцы и строки могут быть указаны отдельно – `Import["file.xlsx",{representation,sheet,row,column}]`. Примеры ниже, в последней секции извлекаются из второго листа, элементы из строк 3, 5, 7 второго столбца:

```
Import["IVT_12.xlsx", {"Data", "Second Sheet", All, 1}]
```

```
Import["IVT_12_xlsx.xlsx", {"Data", "Second Sheet", 3, All}]
```

В секции ниже извлекаются из второго листа элементы из строк 3, 5, 7 второго столбца:

```
Import["IVT_12_xlsx.xlsx", {"Data", 2, {3, 5, 7}, 2}]
```

▼ Очистка данных

Excel файлы обычно содержат ненужные (в *Mathematica*) заголовки в первой строке каждого листа. Для того, чтобы избавиться от этих заголовков, можно использовать функцию `Rest` (`Rest[list]` – возвращает список с удаленным первым элементом):

```
Import["IVT_12_xlsx.xlsx", {"Data", "Second Sheet", All, 2}] // Rest
```

Замечание. При импорте данных *Mathematica* формирует список, с содержанием которого можно работать, используя любую из приведенных ранее функций управления и манипуляций с элементами списков. Примеры:

```
r = Import["IVT_12_xlsx.xlsx"]
```

```
r[[2]]
```

▼ Экспорт данных в файлы Excel

Функция `Export` позволяет экспортировать данные в любой из форматов. Для хранения вычислений часто используются форматы `.csv`, `.xlsx` и `.xls`. Пример экспорта ниже, формируется матрица, выводятся в файл `IVT_11_xls.xls` на лист `data` сама матрица, а на второй лист с именем `transposed data` – транспонированная матрица. На рисунках 12.2 и 12.3 скриншоты иллюстрируют результаты экспорта:

```
m = Table[10 i + j, {i, 9}, {j, 3}]
m // MatrixForm
SetDirectory[NotebookDirectory[]];
Export["IVT_12_xlsx.xls", {"data" → m, "transposedD" → Transpose[m]}]
```


	A	B	C	D	E	F	G	H
1	11	12	13					
2	21	22	23					
3	31	32	33					
4	41	42	43					
5								

Рис. 12.2. Фрагменты листа Excel data с результатом экспорта

	A	B	C	D	E	F	G	H
1	11	21	31	41				
2	12	22	32	42				
3	13	23	33	43				
4								

Рис. 12.3. Фрагменты листа Excel data с результатом экспорта

▼ Экспорт контента в txt файл

- ▶ ✓ Создаём список, отправляем его в txt файл
- ▶ ✓ Формируем и выводим сеточную функцию (таблицу значений)

▼ Экспорт всего блокнота в PDF

Колонтитулы вверху страницы PageHeaders . Колонтитулы внизу страницы PageFooters. Разбивать содержимое ячейки при разбиении на страницы PageBreakWithin.

Отдельный модуль, где собраны все настройки:

C:\Program Files\Wolfram Research\Mathematica\10.0\AddOns\Applications\ AuthorTools\ FrontEnd\TextResources\SetPrintingOptions.nb

C:\Program Files\Wolfram Research\Mathematica\10.2\AddOns\Applications\AuthorTools\ FrontEnd\TextResources\

<https://mathematica.stackexchange.com/questions/1877/efficiency-of-paralleldo-when-a-global-variable-matrix-is-being-updated?rq=1>

c:\Program Files\Wolfram Research\Mathematica\11.3\AddOns\Applications\AuthorTools\FrontEnd\TextResources\

```
sFileName = "T:\\PedProc\\IVT\\IVT_12_forPDF.pdf";
SetDirectory[NotebookDirectory[]];
DeleteFile[sFileName]
NotebookPrint[EvaluationNotebook[], sFileName]
(* Установки печати: File Name, Letter, 11, 11, 22, 22 *)
SystemOpen[sFileName]
```

▼ Экспорт графиков в растровый формат (например, PNG)

Пример экспорта получаемого (формируемого) графика

```
SetDirectory[NotebookDirectory[]];
Export[NotebookDirectory[] <> "IVT_12_Plot_191118.png",
Plot[{Sin[x], Cos[x]}, {x, 0, 2  $\pi$ }, PlotLegends → Automatic], "PNG"]
```

Пример получения и последующего экспорта сформированного графика

```
grf1 = Plot[ {Sin[x^2], x + Sin[2 * x^2.5], x * Sin[1.5 * x^3] - x}, {x, 0, 3},
  ImageSize → 540, PlotStyle → {{Blue, Thin}, {Green, Thick, Dashed}, {Red, Thick}},
  AxesStyle → {Directive[Black, Thick, 14], Directive[Blue, Thick, 22]},
  GridLines → {{1, 1.5, 2, 2.5, 3}, {-6, -3, 2, 4}},
  GridLinesStyle → Directive[Gray, Dashed],
  PlotLabel → Style[Framed["Демонстрационный график с оформлением", FrameStyle → Red, RoundingRadius → 5],
    16, Plain, Blue, Background → LightYellow], Filling → {1 → {2}}]
```

```
SetDirectory[NotebookDirectory[]];
Export[NotebookDirectory[] <> "IVT_12_Plot_grf1.png", grf1, "PNG"]
```

Замечание. В заданиях практикума даны упражнения по примерам использования функций системы для упорядочения и очистки данных, извлекаемых из веб-сайтов.

▼ Экспорт графиков в растровый формат JPG

Пример экспорта получаемого (формируемого) графика в JPG. 2 варианта

```
SetDirectory[NotebookDirectory[]];
Export[NotebookDirectory[] <> "IVT_12_Plot_191118.jpg",
  Image[Plot3D[Sin[x] + Cos[y], {x, -10, 10}, {y, -10, 10}]]]
```

```
SetDirectory[NotebookDirectory[]];
Export[NotebookDirectory[] <> "IVT_12_Plot_191118p.jpg",
  Plot3D[Sin[x] + Cos[y], {x, -10, 10}, {y, -10, 10}]]]
```

```
Export["IVT_12_SinCos.jpg",
  Image[Plot3D[Sin[x] + Cos[y], {x, -10, 10}, {y, -10, 10}]]]
```

▼ Защита блокнота паролем

Mathematica содержит разные средства защиты данных, их возможности достаточно широкие, и это может быть предметом отдельного рассмотрения. Ниже приведены секции с функциями простейшего кодирования с паролем и открытия в виде оригинала

для тестового блокнота. В кодах в следующих секциях, кроме рассмотренных ранее, используются функции:

- `Encode[“source”,“dest”,“key”]` – производит и сохраняет в файл `dest` закодированную версию файла `source`, который нужно считывать, используя `Get[“dest”,“key”]`;
- `Get[“filename”,“key”]` – читает файл `filename`, который закодирован функцией `Encode` с использованием ключа “key”;
- ▶ Результатом выполнения следующей секции будет создание закодированного указанным паролем файла, сохранение его с записанным именем в рабочем каталоге (`Encode...`):
- ▶ Результатом выполнения одной из следующих секций будет раскодирование и открытие в секции вывода полного кода файла, но его вид не соответствует оригиналу (`Get...`):
- ▶ А вот результатом выполнения следующей секции будет раскодирование и открытие в новом блокноте кода в виде оригинала до кодирования (`CreateDocument...`):

.....

▼ Примеры извлечения данных из удаленных серверов

`CountryData` – данные о странах

`CityData` – данные о городах

`AdministrativeDivisionData` – данные об административных единицах

`WeatherData` – метеорологические данные

`FinancialData` – информация о финансовых инструментах

...

▼ Примеры получения и визуализации данных по торгам акций компании EPAM Systems

▼ Код (сокращенное название) компании

Ниже запрос и получение списка компаний Нью-Йоркской фондовой биржи, в названиях которых на первом месте EP (априори код компании в каталоге биржи неизвестен; коды регулярно исключают, добавляют):

```
FinancialData[ "NYSE:EP*", "Lookup" ]
```

```
{NYSE:EP, NYSE:EPAM, NYSE:EPB, NYSE:EPC, NYSE:EPD, NYSE:EPE, NYSE:EPL,
NYSE:EPM, NYSE:EP-PC, NYSE:EPPRC, NYSE:EPR, NYSE:EPRF, NYSE:EPR-PC, NYSE:EPR-PE,
NYSE:EPR-PF, NYSE:EPRPRBCL, NYSE:EPRPRC, NYSE:EPRPRDCL, NYSE:EPRPRE, NYSE:EPRPRF}
```

▼ Ответ на запрос выше (для демонстрации, если нет Интернета):

```
{"NYSE:EP", "NYSE:EPAM", "NYSE:EPB", "NYSE:EPC", "NYSE:EPD", "NYSE:EPE", "NYSE:EPL",
"NYSE:EPM", "NYSE:EP-PC", "NYSE:EPPRC", "NYSE:EPR", "NYSE:EPRF", "NYSE:EPR-PC", "NYSE:EPR-PE",
"NYSE:EPR-PF", "NYSE:EPRPRBCL", "NYSE:EPRPRC", "NYSE:EPRPRDCL", "NYSE:EPRPRE", "NYSE:EPRPRF"}
```

Запрос и получение полного названия компании EPAM:

```
FinancialData["EPAM", "Name"]
```

EPAM Systems

▼ Ответ на запрос выше (для демонстрации, если нет Интернета):

EPAM Systems Inc

▼ Запрос списка дат, стоимостей продаж

Запрос и получение списка дат, стоимостей продаж ("OHLC" – list of open, high, low, and close values for the day) в указанном временном интервале, запоминание списка с именем `finDataEPAM` (если есть Интернет):

Используя функцию `FinancialData`, сделан запрос на сервер Wolfram списка дат и стоимостей акций компании EPAM Systems на момент закрытия Нью-Йоркской фондовой биржи в временном интервале с 1/01/2019 по 17/11/2019, выведена диаграмма.

```
finDataEPAM = FinancialData["EPAM", {{2020, 1, 1}, {2020, 3, 31}}]
```

▶ Ответ на запрос выше (для демонстрации, если нет Интернета), определяем `finDataEPAM`:

▼ Примеры извлечения и визуализации финансовых данных (`CandlestickChart`, `TradingChart`)

Запрос и получение списка дат, стоимостей продаж ("OHLC" – list of open, high, low, and close values for the day) в указанном временном интервале, запоминание списка с именем `finDataEPAM` (если есть Интернет):

```
finDataEPAMohlc = FinancialData["EPAM", "OHLC", {{2020, 1, 1}, {2020, 3, 31}}]
```


► Ответ на запрос выше (для демонстрации, если нет Интернета, `finDataEPAMohlcl`):

CandlestickChart – диаграмма японские свечи, чаще всего используется для отображения биржевых, а именно значений высокой цены, низкой цены, цены открытия и цены закрытия

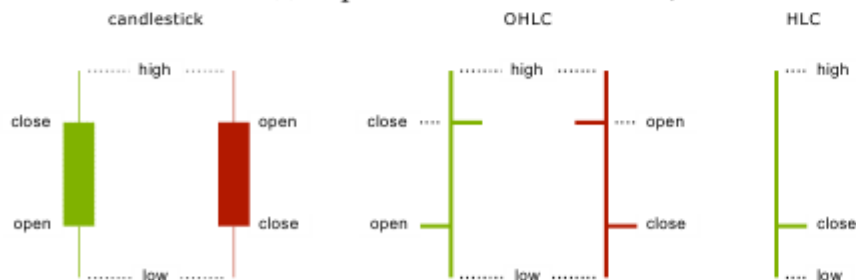
```
CandlestickChart[finDataEPAMohlcl, BaseStyle -> 14, AspectRatio -> 4 / 10, ImageSize -> 650]
```



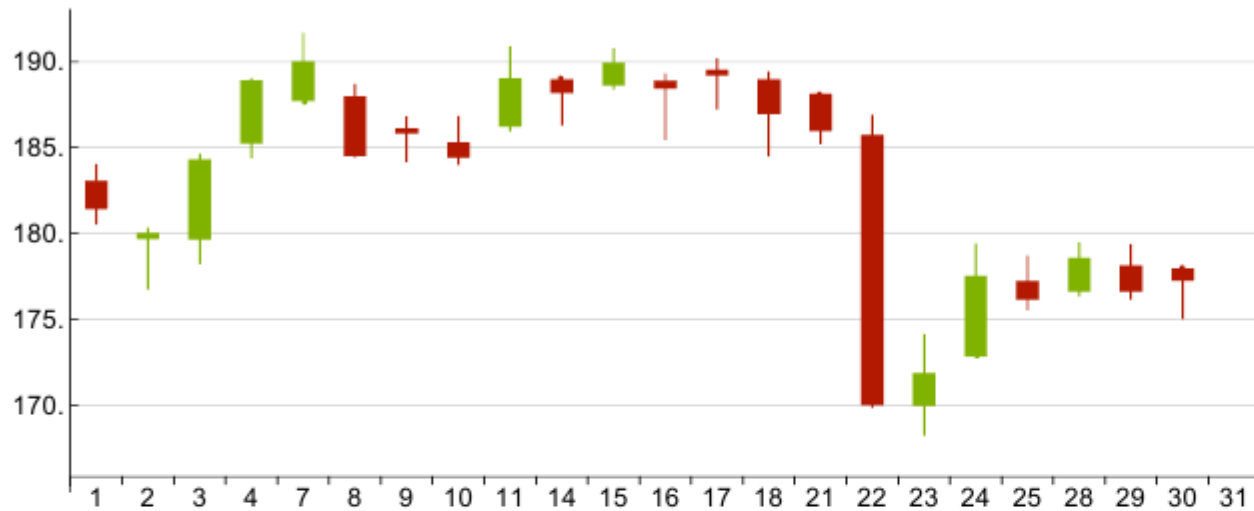
```
finDataEPAMohlcl2 = FinancialData["EPAM", "OHLC", {{2019, 10, 1}, {2019, 10, 30}}]
```

► Ответ на запрос выше (для демонстрации, если нет Интернета, `finDataEPAMohlcl2`):

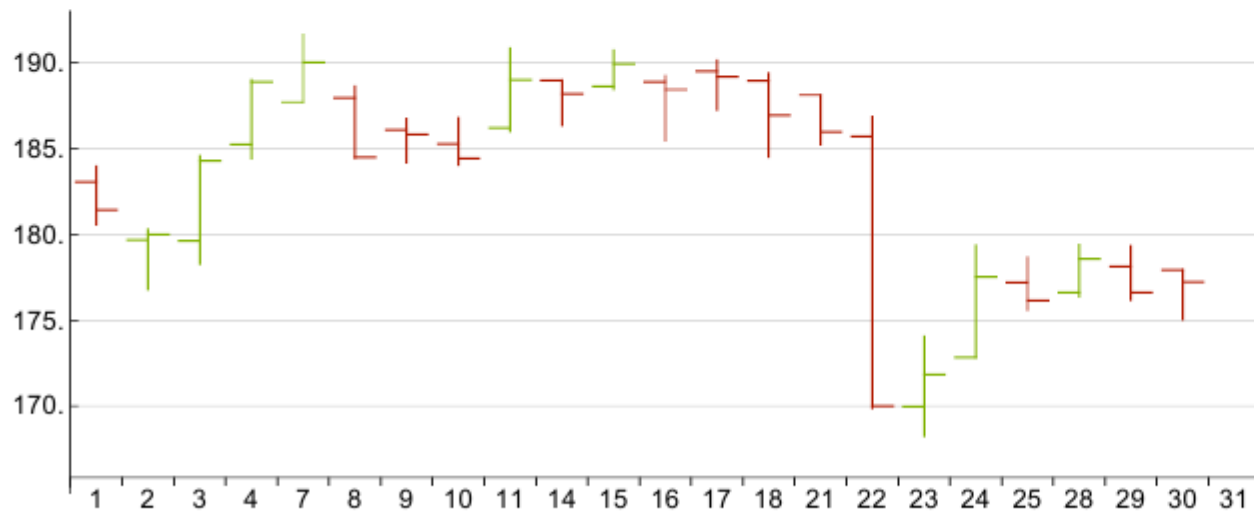
CandlestickChart – диаграмма японские свечи, ...



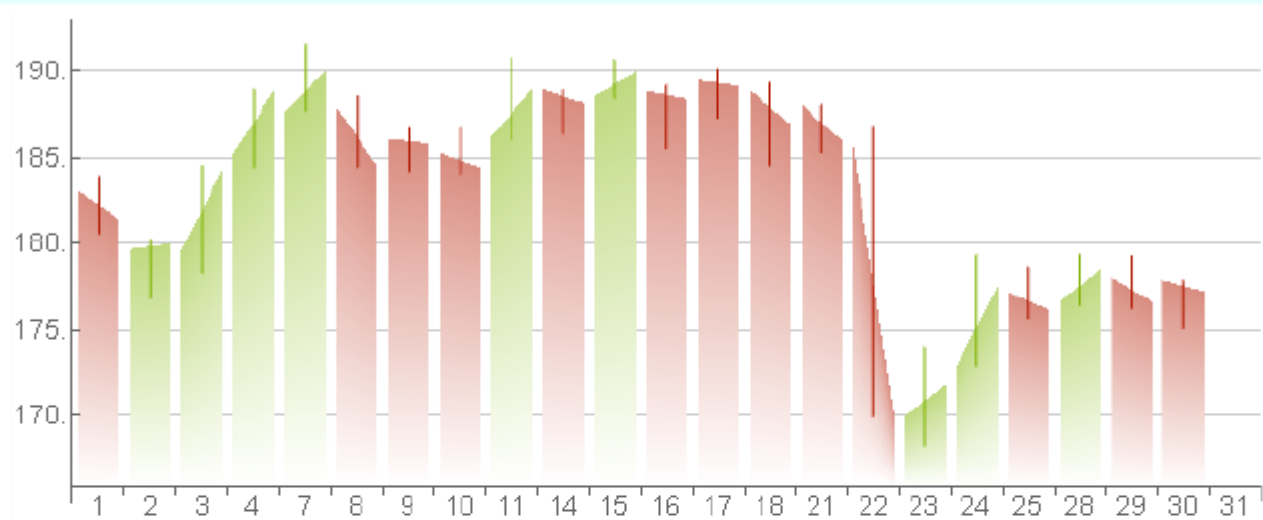
```
CandlestickChart[finDataEPAMohl2, BaseStyle → 14, AspectRatio → 4 / 10, ImageSize → 650]
```



```
CandlestickChart[finDataEPAMohl2, BaseStyle → 14,  
AspectRatio → 4 / 10, ImageSize → 650, ChartElementFunction → "OHLC"]
```



```
CandlestickChart[finDataEPAMohlcv2, BaseStyle → 14,
  AspectRatio → 4 / 10, ImageSize → 650, ChartElementFunction → "FadingOHLC"]
```



Запрос и получение списка дат, стоимостей, объемов продаж (“OHLCV” - list of open, high, low, close, and volume for the day) в указанном временном интервале, запоминание списка с именем `finDataEPAM` (если есть Интернет):

```
finDataEPAMohlcv = FinancialData["EPAM", "OHLCV", {{2019, 1, 1}, {2019, 10, 30}}]
```

► Ответ на запрос выше (для демонстрации, если нет Интернета. `finDataEPAMohlcv`):

```
finDataEPAMohlcv2 = FinancialData["EPAM", "OHLCV", {{2019, 10, 1}, {2019, 10, 30}}]
```

► Ответ на запрос выше (для демонстрации, если нет Интернета. `finDataEPAMohlcv2`):

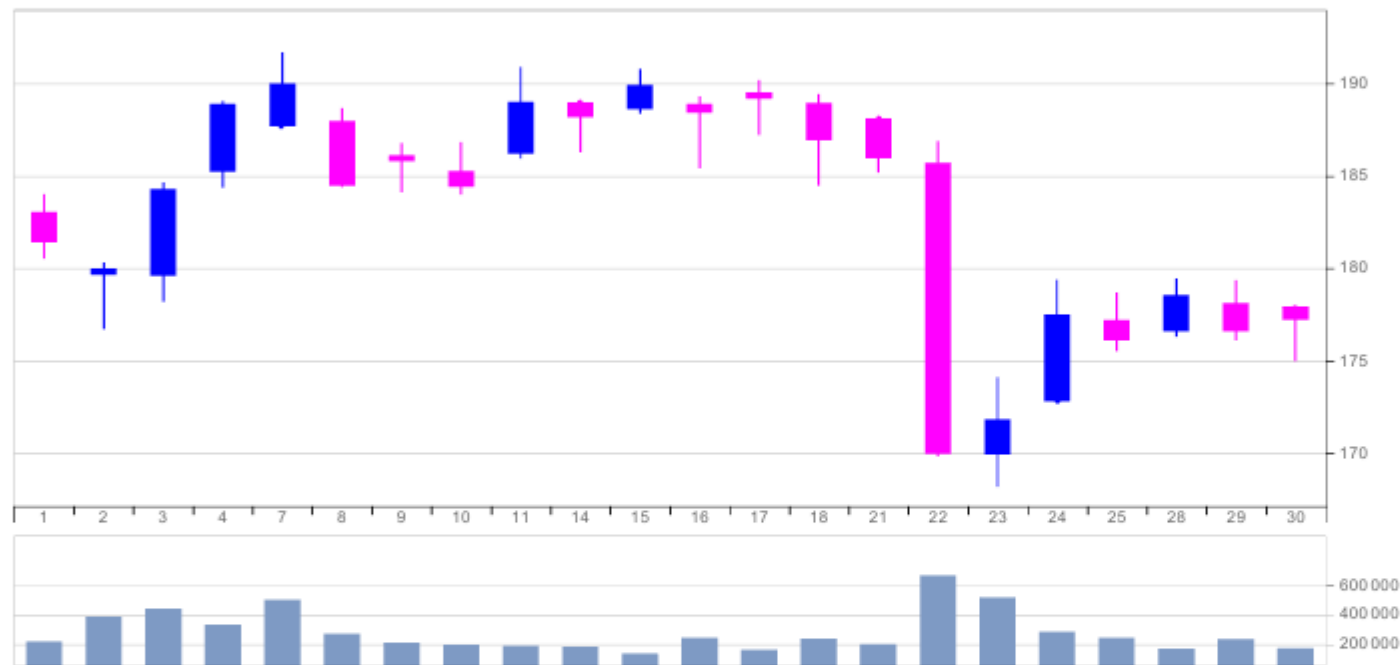
```
CandlestickChart[finDataEPAMohlcv, BaseStyle → 14, AspectRatio → 4 / 10, ImageSize → 650]
```

TradingChart – торговый график, генерирует диаграмму, показывающую цены (минимальную, максимальную, при открытии и закрытии) объемы продаж на каждую дату диапазона. Обратить внимание – строка сверху при интерактивном просмотре

► Ответ на запрос выше (для демонстрации, если нет Интернета. `finDataEPAMohlcv2`):

```
TradingChart[finDataEPAMohlcv, BaseStyle -> 14,
  AspectRatio -> 4 / 10, ImageSize -> 740, TrendStyle -> {Blue, Magenta}]
```

```
TradingChart[finDataEPAMohlcv2, BaseStyle -> 14,
  AspectRatio -> 4 / 10, ImageSize -> 740, TrendStyle -> {Blue, Magenta}]
```



▼ О графических объектах системы *Mathematica*

▼ Концептуально графики в системе *Mathematica* являются ..

Графика системы *Mathematica* всегда считалась для многих пакетов и систем эталоном и во многом способствовала ее высокой репутации мирового лидера среди подобных программных приложений. Графические возможности достигаются, как обилием встроенных функций графики, так и средствами их модификации с помощью директив, опций и примитивов, языка программирования. *Mathematica* позволяет строить практически любые виды математических графиков, причем обычного пользователя в большинстве случаев удовлетворяют графики, параметры которых система задает по умолчанию.

Концептуально графики в системе *Mathematica* являются объектами, которые создаются (возвращаются) соответствующими графическими функциями. Их немного, но они охватывают построение практически всех типов математических графиков.

Достигается это за счет применения функций, опций и директив. Графическим объектам можно присваивать имена, а затем оперировать как любыми объектами.

Программирование графики в *Mathematica* относится к функциональному типу и сводится, в основном, к выбору вложенных друг в друга или одиночных функций. Почти каждый ввод в системе является функцией (командой), каждая имеет определённый набор атрибутов и может иметь набор опций, директив. Атрибуты важны для эффективной работы системы и могут использоваться при программировании. Пользователи *Mathematica* могут применять опции для управления работой отдельных функций. Опции влияют на результат выполнения функции, различные варианты опций могут дать совершенно разные по виду результаты.

Mathematica содержит множество средств для визуализации функций и данных. Система автоматизирует процесс подбора и указания разных деталей построения графика. Применение соответствующих опций позволяет оформлять графические образы в любом виде, дизайне. Опция всегда имеет значение по умолчанию, которое определяет действие функции, если не указано другое. Несмотря на то, что используемые по умолчанию параметры визуализации тщательно подобраны, чтобы наилучшим образом (с точки зрения разработчиков) подходить для большинства типичных случаев, *Mathematica* обеспечивает пользователей возможностями индивидуальной настройки.

Более того, как уже отмечалось ранее, пользователь может в Out-секции блокнота *Mathematica* изменить размер графического изображения, 3D объекты можно вращать и перемещать. Отдельно будет обстоятельный разговор об интерактивных графических объектах, точнее – об инструментах, позволяющих “оживить”, сделать иллюстрации управляемыми, интерактивными с разными альтернативами не только оформления, но и изменения визуализируемых функций.

▼ Каждое изображение в системе называется графическим объектом ..

Используя оригинальные алгоритмы, разработанные в компании, на языке Wolfram отлажены и включены в систему мощные функции, которые автоматизируют процесс создания содержательных и эстетически оформляемых иллюстраций функций, структурированных и неструктурированных данных, причём не только для точек, линий и поверхностей, но и для графов и сетей.

Каждое графическое изображение в системе *Mathematica* называется графическим объектом. Условно (к настоящему моменту нет общепринятой терминологии) формируемые системой графические объекты можно разделить на следующие категории:

▼ Типы (категории) графических объектов *Mathematica* [Терминология В.Тар]. КСР

▼ Общие замечания

Mathematica содержит разные средства для графического отображения функций и данных. Система предоставляет пользователю множество возможностей настройки, оформления деталей выводимых графиков, в частности, таких как цветовой дизайн, эстетичность и масштаб, частота дискретизации. Несмотря на то, что используемые по умолчанию параметры тщательно подобраны, чтобы наилучшим образом подходить для подавляющего большинства случаев, *Mathematica* позволяет делать

пользовательские настройки визуализации, в соответствии с предпочтениями пользователя.

Условно (к настоящему моменту нет общепринятой терминологии) формируемые системой графические объекты можно разделить на следующие категории:

▼ О графических объектах системы *Mathematica*

Графика системы *Mathematica* всегда считалась для многих пакетов и систем эталоном и во многом способствовала ее высокой репутации мирового лидера среди подобных программных приложений. Графические возможности достигаются, как обилием встроенных функций, так и средствами их модификации с помощью директив, опций, примитивов, языка программирования. *Mathematica* позволяет строить любые виды математических графиков, причем обычного пользователя в большинстве случаев удовлетворяют графики, параметры которых задаются по умолчанию. В системе поддерживается работа практически со всеми форматами графики, их импорт и экспорт.

Концептуально графики в *Mathematica* являются объектами, которые создаются (возвращаются) соответствующими графическими функциями. Они охватывают построение всех типов математических графиков, иллюстраций деловой графики – гистограмм, двумерных и трехмерных секторных и столбчатых, пузырьковых диаграмм, специфических графиков для таких областей, как финансы и статистика, теория графов, управляющие системы. Достигается многообразие за счет применения функций, опций и директив. Графическим объектам можно присваивать имена, а затем оперировать как любыми объектами.

Программирование графики в *Mathematica* относится к функциональному типу. Почти каждый ввод в системе является функцией (командой), каждая имеет определённый набор атрибутов и может иметь набор опций, директив. Атрибуты важны для эффективной работы системы и могут использоваться при программировании. Пользователи *Mathematica* могут применять опции для управления работой отдельных функций. Опции влияют на результат выполнения функции, различные варианты опций могут дать совершенно разные по виду результаты.

Более того, как уже отмечалось ранее, пользователь может в Out-секции блокнота *Mathematica* изменить размер графического изображения, 3D объекты можно вращать и перемещать. Отдельно будет обстоятельный разговор об интерактивных графических объектах, точнее – об инструментах, позволяющих “оживить”, сделать иллюстрации управляемыми, интерактивными с разными альтернативами не только оформления, но и изменения параметров визуализации.

▼ Типы (категории) графических объектов *Mathematica*

Используя оригинальные алгоритмы, разработанные в компании, отлажены и включены в систему многочисленные функции, которые автоматизируют процесс создания содержательных и эстетически оформляемых иллюстраций функций, данных (структурированных и неструктурированных), причём не только для точек, линий и поверхностей, но и для графов и сетей. Каждое графическое изображение в системе *Mathematica* называется графическим объектом. Условно (к настоящему моменту нет общепринятой терминологии) формируемые системой графические объекты можно разделить на следующие категории:

- ▶ Графики аналитически задаваемых функций одной переменной
 - ▶ Визуализация массивов одномерных данных
 - ▶ Гистограммы, столбиковые, круговые, секторные диаграммы
 - ▶ Финансовые диаграммы
 - ▶ Специальные функции графики пакетов-приложений
 - ▶ Средства 2D графики аналитически задаваемых функций
 - ▶ Функции интерполяции и визуализации 2D графики массивов данных
 - ▶ Функции формирования и вывода объектов 2D графики
 - ▶ Средства 3D графики аналитически задаваемых функций
 - ▶ Функции интерполяции и визуализации 3D графики массивов данных
 - ▶ Географические визуализации и вычисления
 - ▶ Функции сбора, формирования и вывода объектов 3D графики
-

▼ Рекомендуемая литература

3. Таранчук, В.Б. Введение в язык Wolfram : учеб. материалы для студентов фак. прикладной математики и информатики спец. 1-31 03 04 «Информатика» / В.Б. Таранчук. – Минск : БГУ, 2015. – 51 с. <http://www.elib.bsu.by/handle/123456789/120745>
4. Таранчук, В.Б. Основы программирования на языке Wolfram : учеб. материалы для студентов фак. прикладной математики и информатики спец. 1-31 03 04 «Информатика» / В.Б. Таранчук. – Минск : БГУ, 2015. – 49 с. <http://www.elib.bsu.by/handle/123456789/120748>
5. Таранчук, В.Б. Введение в графику системы Mathematica : учеб. материалы для студентов фак. прикладной математики и информатики / В.Б. Таранчук. – Минск : БГУ, 2017. – 53 с. <http://elib.bsu.by/handle/123456789/186348>