

# **Компьютерная математика**

---

## **Лекция 3. Использование графики в пакете Mathematica**

# Построение графиков

File Edit Insert Format Cell Graphics Evaluation Palettes Window Help

In[1]:= ?\*Plot\*

System`

AbsArgPlot

AnatomyPlot3D

ArrayPlot

AudioPlot

BodePlot

ChromaticityPlot

ChromaticityPlot3D

CommunityGraphPlot

ComplexContourPlot

ComplexListPlot

ComplexPlot

ComplexPlot3D

ComplexRegionPlot

ComplexStreamPlot

ComplexVectorPlot

LogLinearPlot

LogLogPlot

LogPlot

MandelbrotSetPlot

MatrixPlot

MaxPlotPoints

MoleculePlot

MoleculePlot3D

NicholsPlot

NumberLinePlot

NyquistPlot

ParametricPlot

ParametricPlot3D

Plot

Plot3D

# Построение графиков

Смысл или значение каждой опции можно найти в оперативной справке. Для этого достаточно установить курсор в любом месте уточняемого символа и нажать клавишу **F1**.

BUILT-IN SYMBOL See Also Related Guides Tutorials URL

## Plot3D (график функции 2-х переменных) UPDATED IN 12

`Plot3D[f, {x, xmin, xmax}, {y, ymin, ymax}]`  
generates a three-dimensional plot of  $f$  as a function of  $x$  and  $y$ .

`Plot3D[{f1, f2, ...}, {x, xmin, xmax}, {y, ymin, ymax}]`  
plots several functions.

`Plot3D[{...}, w[f], ...]`  
plots  $f_i$  with features defined by the symbolic wrapper  $w$ .

`Plot3D[...], {x, y} ∈ reg`  
takes variables  $\{x, y\}$  to be in the geometric region  $reg$ .

### Details and Options

<b>Options</b>	<b>Options</b>	<b>Options</b>	<b>Options</b>
<code>PlotStyle</code>	<code>PlotRange</code>	<code>PlotRange</code>	<code>PlotRange</code>
<code>PlotStyle</code>	<code>PlotRange</code>	<code>PlotRange</code>	<code>PlotRange</code>
<code>PlotStyle</code>	<code>PlotRange</code>	<code>PlotRange</code>	<code>PlotRange</code>
<code>PlotStyle</code>	<code>PlotRange</code>	<code>PlotRange</code>	<code>PlotRange</code>

### Examples (158)

#### Basic Examples (4)

Plot a function:

```
In[1]:= Plot3D[Sin[x + y^2], {x, -3, 3}, {y, -2, 2}]
```

Out[1]=

# Базовые функции двумерной графики

---

**Plot**[{ $f_1[x]$ , ...,  $f_k[x]$ }, { $x$ ,  $x_{\min}$ ,  $x_{\max}$ }]

Функция строит графики функций, заданных уравнениями  $y = f_1[x]$ , ...,  $y = f_k[x]$  в декартовой системе координат.

**ListPlot**[{ $y_1$ , ...,  $y_k$ }]

Функция визуализирует числовую последовательность, то есть вещественнозначную функцию, заданную на множестве натуральных чисел. Результат выполнения функции ListPlot – множество точек плоскости с координатами  $\{i, y_i\}$ ,  $i = 1, \dots, k$ .

---

# Базовые функции двумерной графики

---

**ListPlot**[ $\{\{x_1, y_1\}, \dots, \{x_k, y_k\}\}$ ]

Функция отображает список чисел в виде точек с указанными координатами  $\{x_i, y_i\}$ .

**ParametricPlot**[ $\{x[t], y[t]\}, \{t, t_{\min}, t_{\max}\}$ ].

Функция используется для изображения параметрических кривых.

Чтобы построить графики нескольких функций, заданных параметрически, в качестве первого аргумента следует указать список соответствующих законов, каждый из которых – список уравнений  $\{\{x_1[t], y_1[t]\}, \dots, \{x_k[t], y_k[t]\}\}$ .

---

# Базовые функции двумерной графики

---

**ContourPlot[f[x, y], {x, x<sub>min</sub>, x<sub>max</sub>}, {y, y<sub>min</sub>, y<sub>max</sub>}]** представляет заданную поверхность в виде линий уровня  $f[x, y] = \text{const}$  на плоскости.

Поверхность  $z = f[x, y]$  может быть задана матрицей своих значений  $z_{i,j} = f[x_i, y_j]$  в узлах  $(x_i, y_j)$ . Тогда линии уровня этой поверхности  $f[x_i, y_j] = \text{const}$  изображаются при помощи функции

**ListContourPlot[array]**

---

# Базовые функции двумерной графики

---

**DensityPlot [f [x, y], {x, x<sub>min</sub>, x<sub>max</sub>}, {y, y<sub>min</sub>, y<sub>max</sub>}]**

Функция представляет трехмерную поверхность в виде графика плотности. Она отображает значения указанной функции с помощью окрашивания плоскости XOY в серый цвет различной интенсивности. Области, соответствующие большему значению  $z = f[x, y]$ , представлены более светлыми тонами.

**ListDensityPlot[array]**

представляет указанный массив в виде графика плотности. Элементы массива  $z_{i,j}$  являются значениями некоторой функции в узлах  $z_{i,j} = f[x_i, y_j]$ .

---

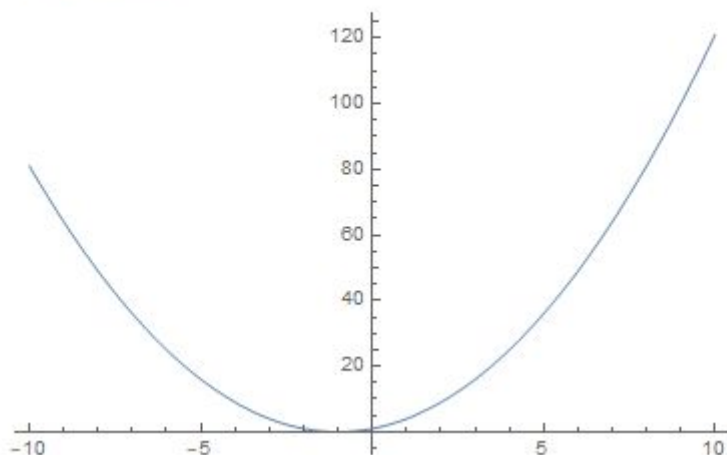
# Примеры

---

In[2]:= `Plot[x^2 + 2 * x + 1, {x, -10, 10}]`

график функции

Out[2]=



(Интервал  $\{x, \text{min}, \text{max}\}$  используется для задания области определения аргумента.)

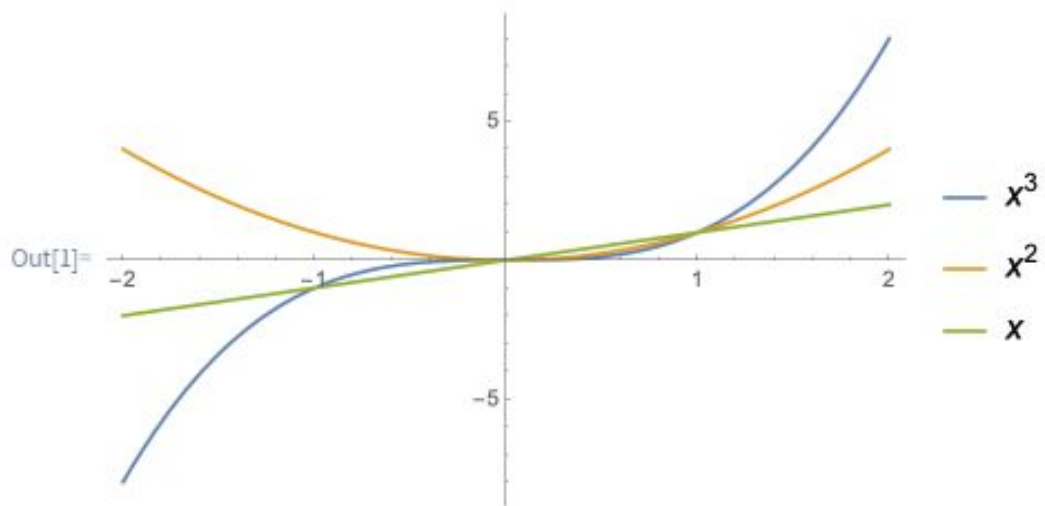
---



# Примеры

---

In[1]:= Plot[{ $x^3$ ,  $x^2$ ,  $x$ }, { $x$ , -2, 2}, PlotLegends → "Expressions"]



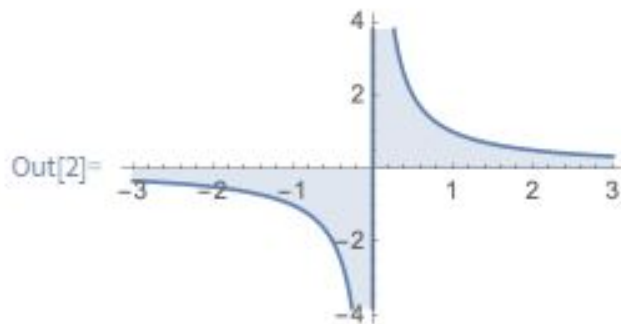
Добавление легенд

---

# Примеры

---

```
In[2]:= Plot[1/x, {x, -3, 3}, Filling -> Axis]
```



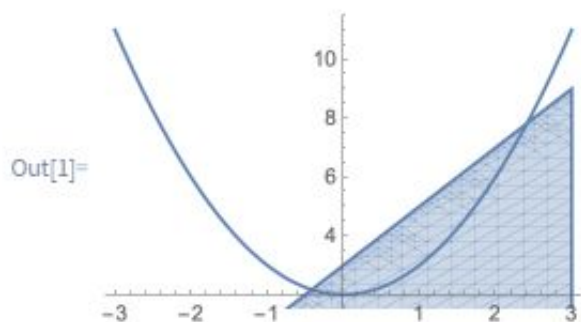
**Закрашивание области под графиком**

---

# Примеры

---

```
In[1]:= Show[{Plot[x2 + 2, {x, -3, 3}], RegionPlot[2 x > y - 3, {x, -3, 3}, {y, 0, 9}]]
```



Соединим несколько типов графиков с помощью функции Show

**КРАТКОЕ ВВЕДЕНИЕ ДЛЯ СТУДЕНТОВ МАТЕМАТИЧЕСКИХ СПЕЦИАЛЬНОСТЕЙ**

<https://www.wolfram.com/language/fast-introduction-for-math-students/ru///>

---

# Директивы двумерной графики

---

- ❑ **AbsoluteDashing [ {d1, d2,...} ]** — задает построение линий пунктиром со смежными (последовательными) сегментами, имеющими абсолютные длины d1, d2, ... (повторяемые циклически). Значения длины di задаются в пикселях;
  - ❑ **AbsolutePointSize [d]** — задает построение последующих точек графика в виде кружков с диаметром d (в пикселях);
  - ❑ **AbsoluteThickness [d]** — задает абсолютное значение толщины (в пикселях) для последующих рисуемых линий;
-

# Директивы двумерной графики

---

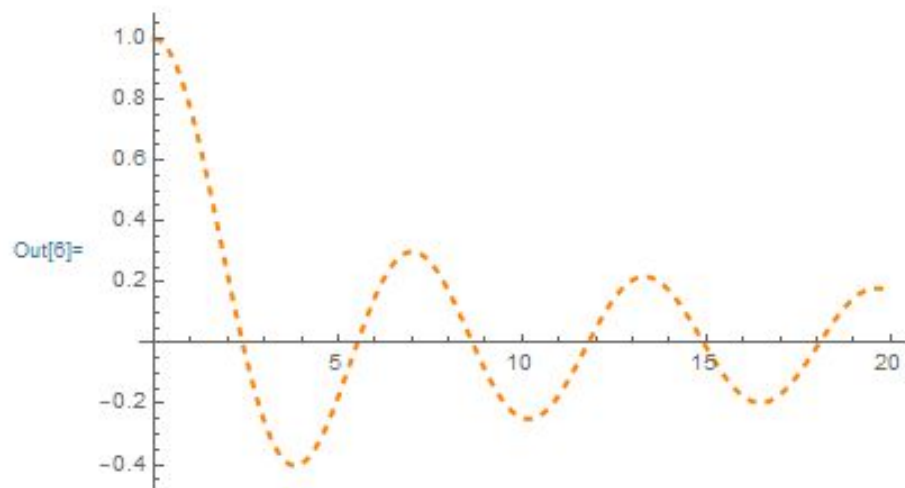
- ❑ **Dashing** [ $\{r_1, r_2, \dots\}$ ] — задает построение последующих линий пунктиром с последовательными сегментами длиной  $r_1, r_2, \dots$ , повторяемыми циклически, причем  $r_i$  задается как доля полной ширины графика;
  - ❑ **PointSize** [ $d$ ] — задает вывод последующих точек графика в виде кружков с относительным диаметром  $d$ , заданным как доля общей ширины графика;
  - ❑ **Thickness** [ $r$ ] — устанавливает для всех последующих линий толщину  $r$ , заданную как доля полной ширины графика.
-

# Примеры

---

```
In[6]:= Plot[BesselJ[0, x], {x, 0, 20}, PlotStyle -> {Orange, Dashed, Thick}]
```

гра... функция Бесселя J      стиль графика    оранже... штрих... жирный



+

# Примеры

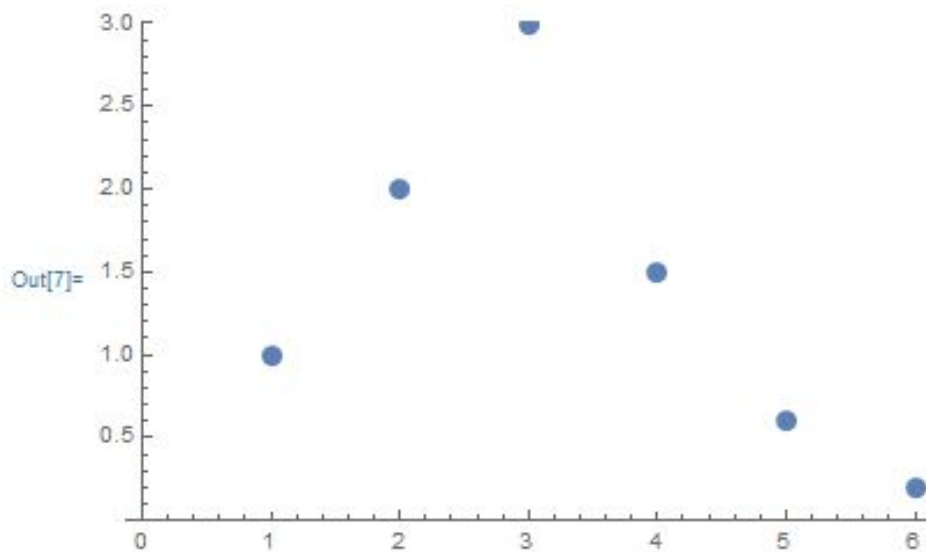
```
In[7]:= ListPlot[{1, 2, 3, 1.5, 0.6, 0.2}, PlotRange -> {0, 3}, PlotStyle -> PointSize[.026]]
```

диаграмма разброса данных

отображаемый диапа...

стиль графика

размер точки



# Примитивы двумерной графики

---

Примитивами двумерной графики называют дополнительные указания, вводимые в функцию

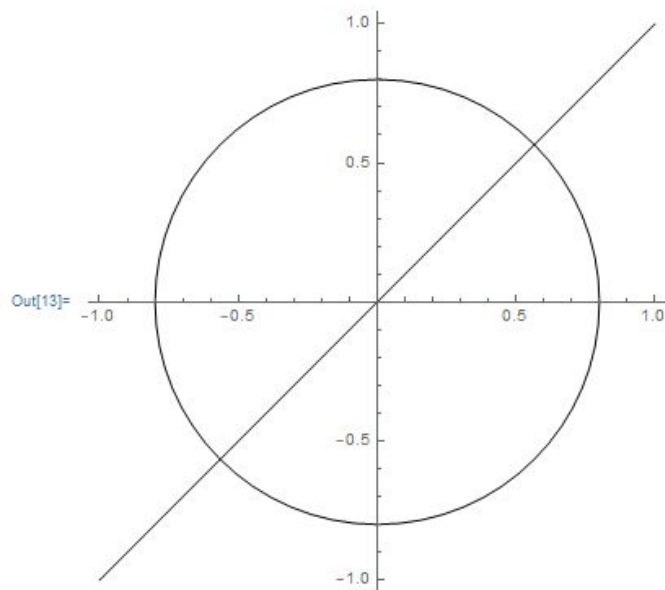
**Graphics [primitives, options]**, которая позволяет выводить различные примитивные фигуры без задания математических выражений, описывающих эти фигуры.

---



# Примитивы двумерной графики

```
In[11]:= g1 := Graphics[Line[{{-1, -1}, {1, 1}}]];
          |графика | (ломаная) линия
          g2 := Graphics[Circle[{0, 0}, 0.8]];
          |графика | круг
          Show[g1, g2, Graphics[PointSize[0.032], Point[{- .5, .2}], Axes → True]
          |показать | графика | размер точки | точка | оси | истина
```



Построение двух графических объектов с помощью примитивов двумерной графики

# Примитивы двумерной графики

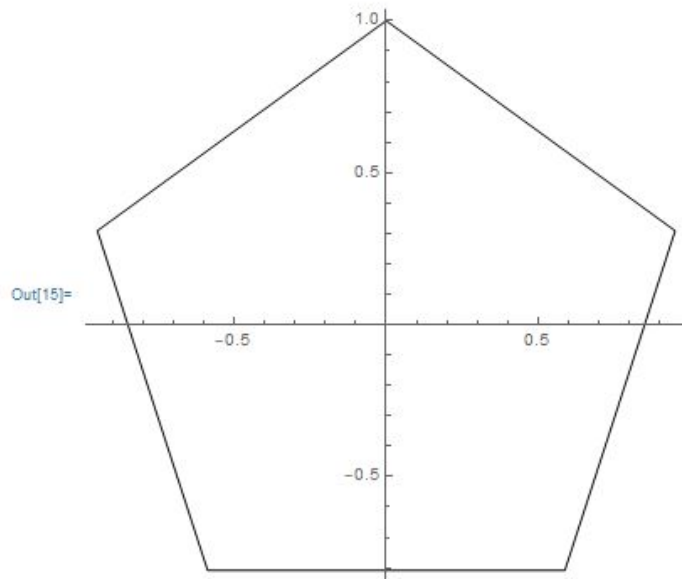
```
In[14]:= pentagon = Table[N[{Sin[2*Pi*n/5], Cos[2*Pi*n/5]}], {n, 6}]
```

[табл... [ч... [синус [число пи [косинус [число пи

```
Show[Graphics[Line[pentagon], Axes -> True]]
```

[пок... [графика [(ломаная) линия [оси [истина

```
Out[14]= {{0.951057, 0.309017}, {0.587785, -0.809017},  
          {-0.587785, -0.809017}, {-0.951057, 0.309017}, {0., 1.}, {0.951057, 0.309017}}
```



Построение пятиугольника

# Графические примитивы

---

Название примитива	Какой объект подготавливается к созданию
Point[ coords ]	Последовательность точек, указанных списком координат coords, в случае 2D вида $\{\{x_1, y_1\}, \dots, \{x_k, y_k\}\}$ , в случае 3D вида $\{\{x_1, y_1, z_1\}, \dots, \{x_k, y_k, z_k\}\}$ .
Line[ coords ]	Ломаная линия, соединяющая точки, указанные списком координат coords, в случае 2D вида $\{\{x_1, y_1\}, \dots, \{x_k, y_k\}\}$ , в случае 3D вида $\{\{x_1, y_1, z_1\}, \dots, \{x_k, y_k, z_k\}\}$ .
Rectangle[ {x <sub>min</sub> , y <sub>min</sub> }, {x <sub>max</sub> , y <sub>max</sub> } ]	Прямоугольник с заливкой определенного цвета, ориентированный параллельно осям координат, в аргументах указываются координаты левой нижней и правой верхней вершин.
Cuboid[ {x <sub>min</sub> , y <sub>min</sub> , z <sub>min</sub> }, {x <sub>max</sub> , y <sub>max</sub> , z <sub>max</sub> } ]	Прямоугольный параллелепипед, грани которого параллельны осям координат, в аргументах указываются координаты левой нижней и правой верхней вершин.

---

# Графические примитивы

---

Polygon[ coords ]	Многоугольник с заливкой определенного цвета, ограниченный ломаной, соединяющей указанные списком координат точки coords, в случае 2D вида $\{\{x_1, y_1\}, \dots, \{x_k, y_k\}\}$ , в случае 3D вида $\{\{x_1, y_1, z_1\}, \dots, \{x_k, y_k, z_k\}\}$ .
Circle[{x, y}, {r <sub>x</sub> , r <sub>y</sub> }]	Эллипс с центром в точке {x, y} и полуосями {r <sub>x</sub> , r <sub>y</sub> }. При указании третьего аргумента – дуга.
Disk [{x, y}, {r <sub>x</sub> , r <sub>y</sub> }]	Эллипс с заливкой определенного цвета с центром в точке {x, y} и полуосями {r <sub>x</sub> , r <sub>y</sub> }. При указании третьего аргумента – сектор.
Text[expression, coords]	Печатная форма выражения expression, центрированного относительно координат coords.

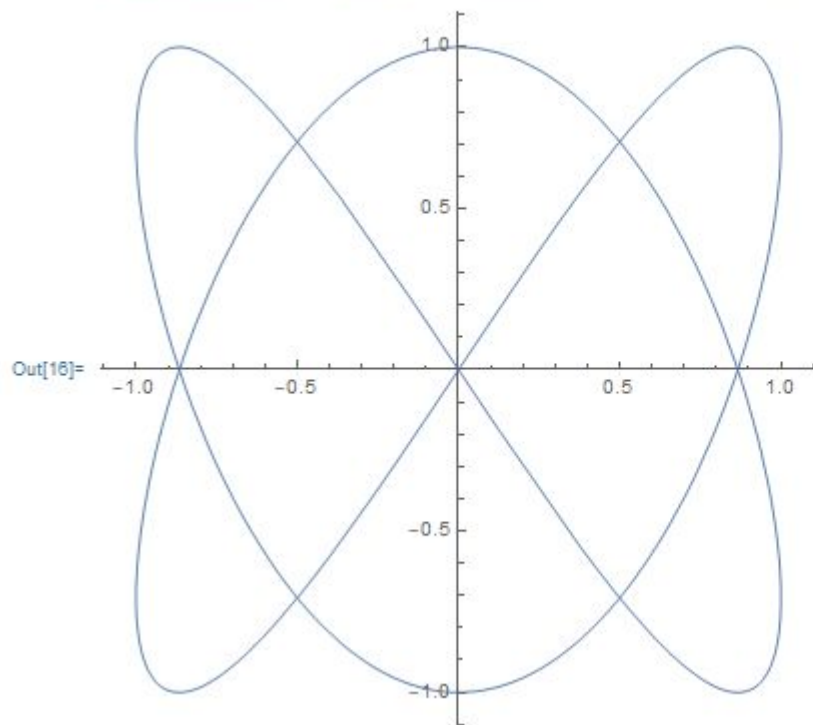
---

# Графики функций, заданных в параметрической форме

---

```
In[16]:= ParametricPlot[{Sin[2*t], Sin[3*t]}, {t, 0, 2*Pi}, AspectRatio -> Automatic]
```

график параметри... | синус | синус | чи... | аспектное отн... | автоматически



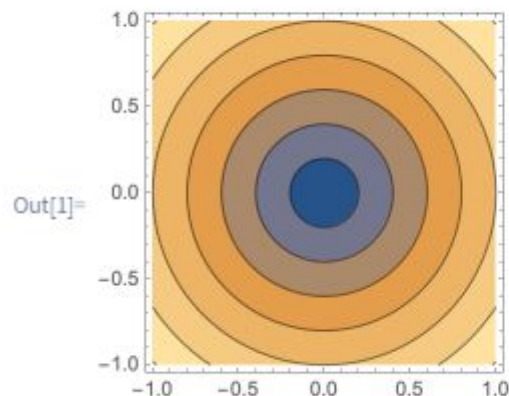
Построение фигуры Лиссажу

---

# Примеры

---

```
In[1]:= ContourPlot[ $\sqrt{x^2 + y^2}$ , {x, -1, 1}, {y, -1, 1}]
```



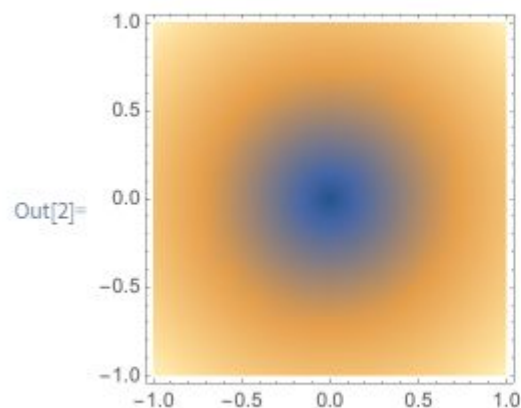
Функция ContourPlot позволяет построить график линий уровня

---

# Примеры

---

```
In[2]:= DensityPlot[ $\sqrt{x^2 + y^2}$ , {x, -1, 1}, {y, -1, 1}]
```



Функция DensityPlot строит непрерывный график - диаграмму плотности функции

---

# Базовые функции трехмерной графики

---

**`Plot3D[f[x, y], {x, xmin, xmax}, {y, ymin, ymax}].`**

График поверхности

**`ParametricPlot3D[{x[t], y[t], z[t]}, {t, tmin, tmax}]`**

позволяет строить  
пространственные  
параметризованные кривые

**`ParametricPlot3D [{x[t], y[t], z[t]}, {t, tmin, tmax}, {u, umin, umax}]`**

Функция отражает параметризованную поверхность

---



# Базовые функции трехмерной графики

---

Дискретным аналогом функции `Plot3D` является функция  
**`ListPlot3D[array]`**.

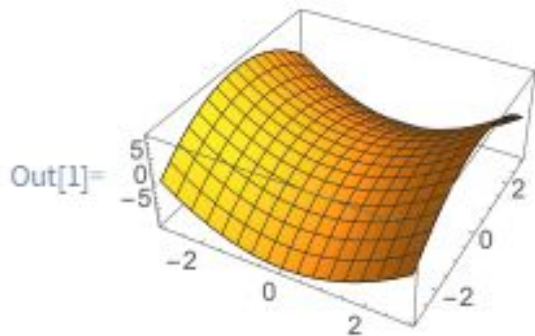
Она строит поверхность, для которой элементы массива являются значениями некоторой функции  $z_{i,j} = f[x_i, y_j]$ .

---

# Примеры

---

```
In[1]:= Plot3D[x2 - y2, {x, -3, 3}, {y, -3, 3}]
```



**Гиперболический параболоид**

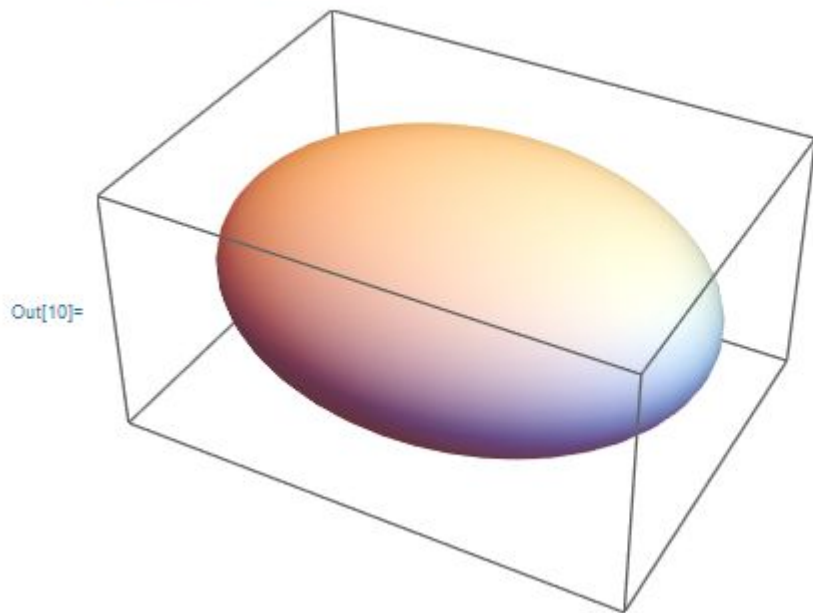
---

# Примеры

---

```
In[10]:= Graphics3D[Ellipsoid[{0, 0, 0}, {4, 3, 2}]]
```

[3-мерная г... эллипсоид



**Эллипсоид**

---

# Примеры

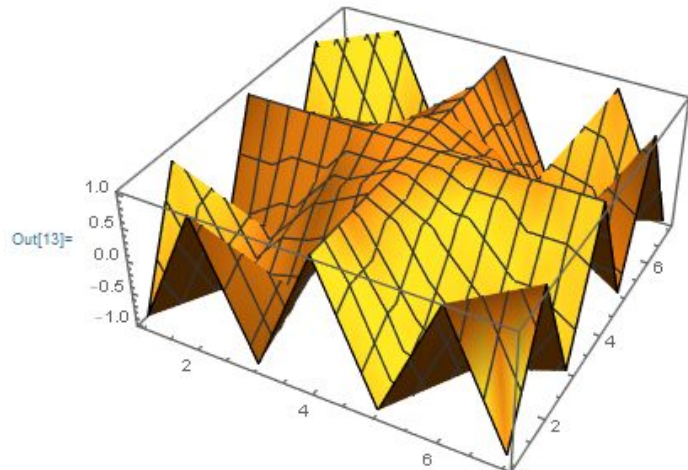
```
In[12]:= t3 = Table[Cos[x*y], {y, -3, 3}, {x, -3, 3}]
```

табл... косинус

```
ListPlot3D[t3]
```

3-мерная диаграмма разброса данных

```
Out[12]= {{Cos[9], Cos[6], Cos[3], 1, Cos[3], Cos[6], Cos[9]},  
          {Cos[6], Cos[4], Cos[2], 1, Cos[2], Cos[4], Cos[6]},  
          {Cos[3], Cos[2], Cos[1], 1, Cos[1], Cos[2], Cos[3]},  
          {1, 1, 1, 1, 1, 1, 1}, {Cos[3], Cos[2], Cos[1], 1, Cos[1], Cos[2], Cos[3]},  
          {Cos[6], Cos[4], Cos[2], 1, Cos[2], Cos[4], Cos[6]},  
          {Cos[9], Cos[6], Cos[3], 1, Cos[3], Cos[6], Cos[9]}}
```



Поверхность задается массивом своих высот