Experiment 2: Plotting and data visulization

Name: Soniya Eknath Girhe.

Sec_Batch_RollNo.: A7_B3_59.

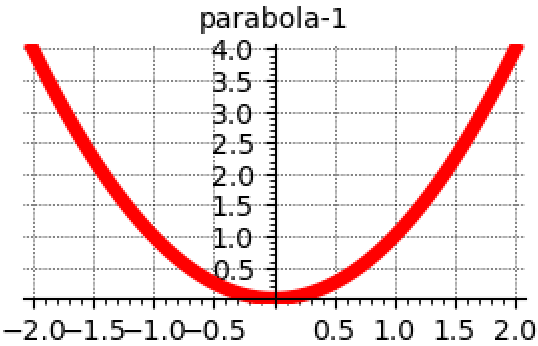
Date: 28/02/2026.

In [16]:
```
var('x')
f(x)=x^2
show(f)
```

$x \mapsto x^2$

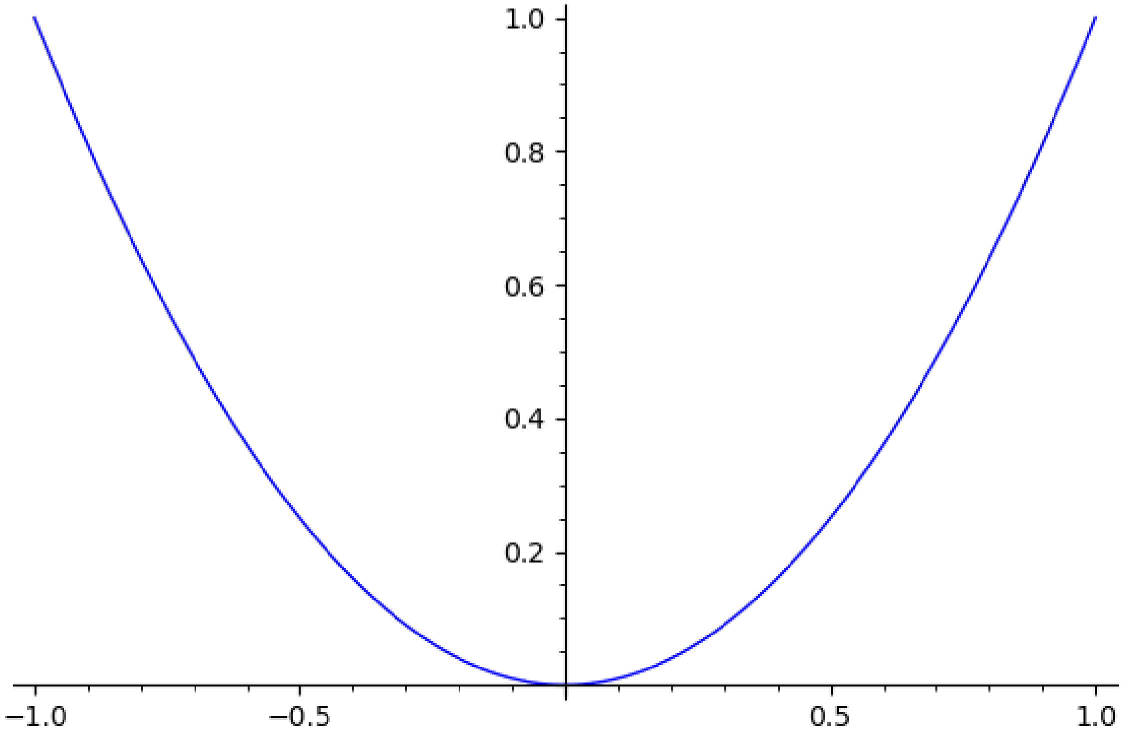
In [7]:
```
p1=plot(f,(x,-2,2),figsize=3,color='red',title='parabola-1',thickness=5,gri
dlines=True)
p1
```

Out[7]:



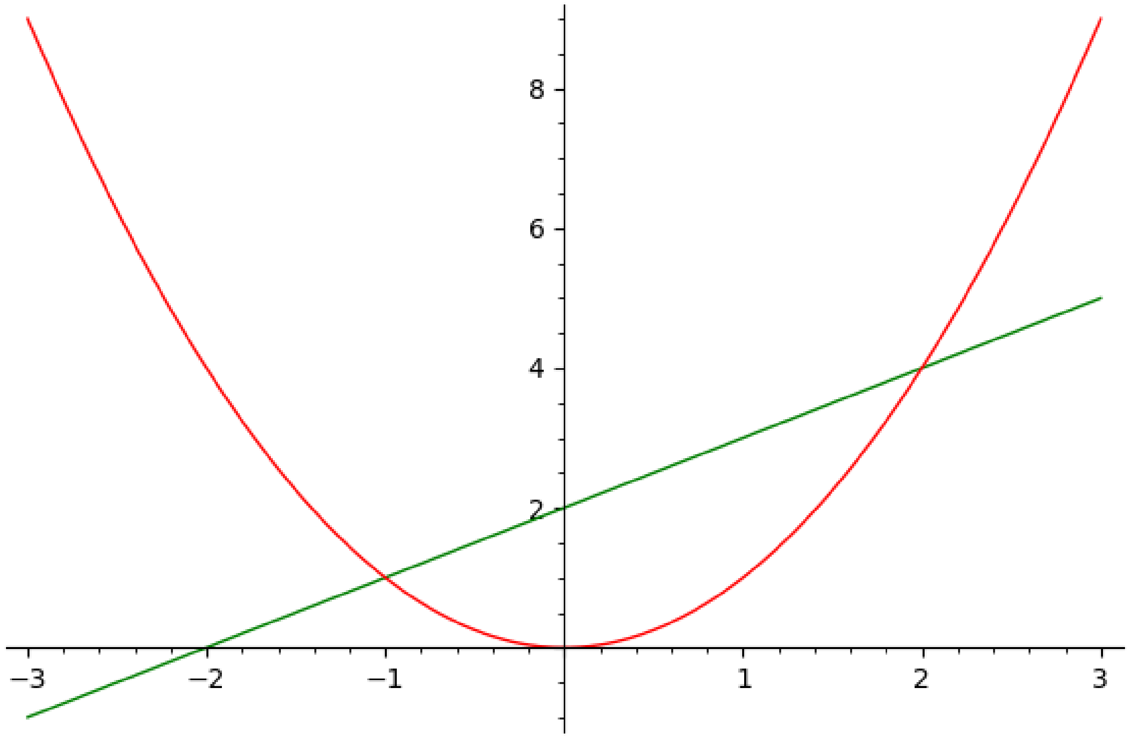
In [8]:
```
plot(f)
```

Out[8]:



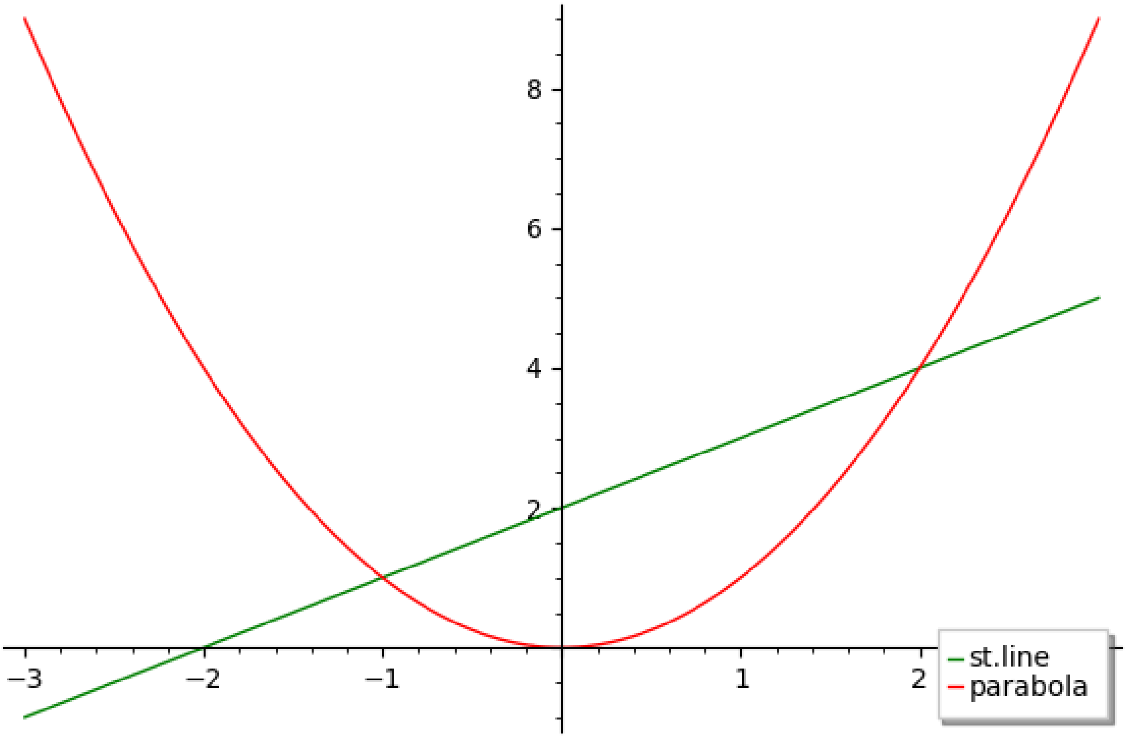
In [10]:
```
plot(x+2,(-3,3),color='green')+plot(x^2,(-3,3),color='red')
```

Out[10]:



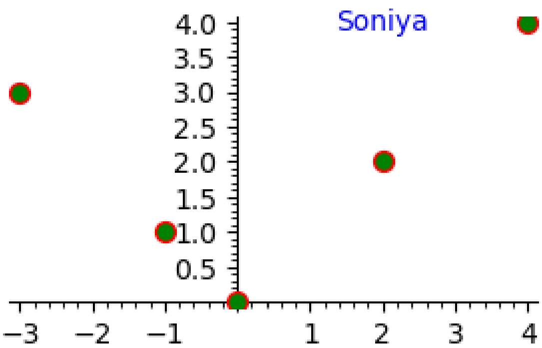
In [14]:
```
p1=plot(x+2,(-3,3),color='green',legend_label="st.line")
p2=plot(x^2,(-3,3),color='red',legend_label="parabola")
p1+p2
```

Out[14]:



In [1]:
```
data=[(0,0),(-1,1),(2,2),(-3,3),(4,4)]
v1=scatter_plot(data,figsize=3,facecolor='green',edgecolor='red')+text('Son
iya',(2,4))
v1
```

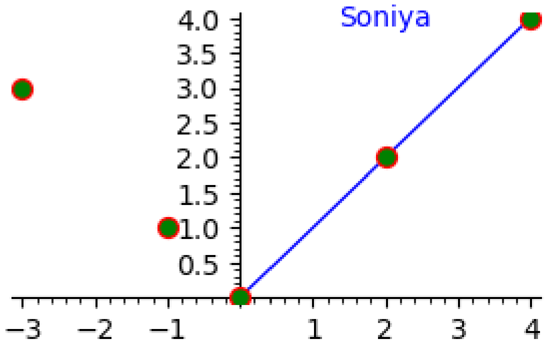
Out[1]:



In [2]:
```
v2=plot(x,(x,0,4))
```

In [3]:
```
v1+v2
```

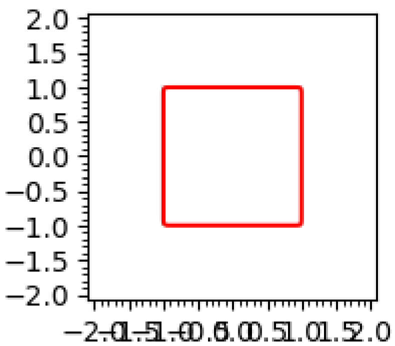
Out[3]:



Implicit plotting

In [27]:
```
var('x,y')
n=50
f(x,y)=(x^n+y^n)-1
y1=implicit_plot(f,(x,-2,2),(y,-2,2),color='red',figsize=3)
y1
## when n is even we will get close graph and if odd it will be open graph#
#
```

Out[27]:

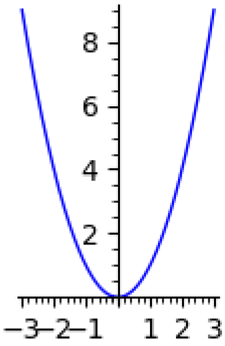


Parametric plotting

$x = tandy = t^2$

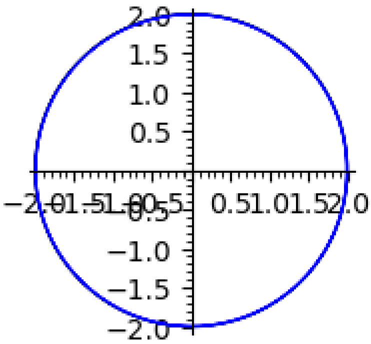
In [4]:
```
var('t')
q1=parametric_plot([t,t^2],(t,-3,3),figsize=3)
q1
```

Out[4]:



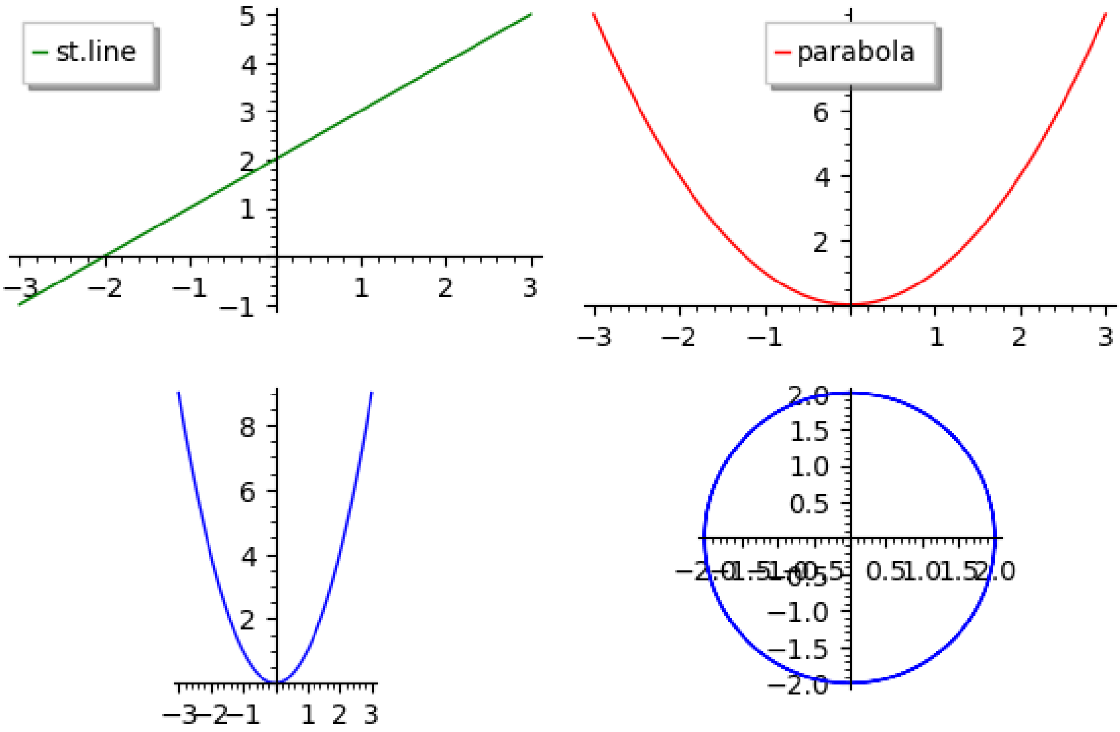
In [5]:
```
var('t')
q2=parametric_plot([2*sin(t),2*cos(t)],(t,-2*pi,2*pi),figsize=3)
q2
```

Out[5]:



In [37]:
```
List=[p1,p2,q1,q2]
graphics_array(List,2,2)
```

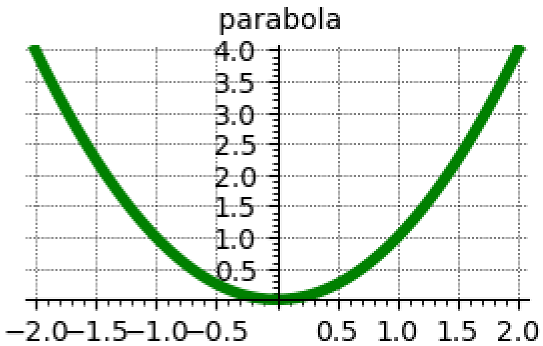
Out[37]:



ex. ii) Plot $f(x) = x^2$

In [7]:
```
plot(x^2,(-2,2),color='green',title=" parabola",figsize=3,thickness=4,gridlines='True')
```

Out[7]:

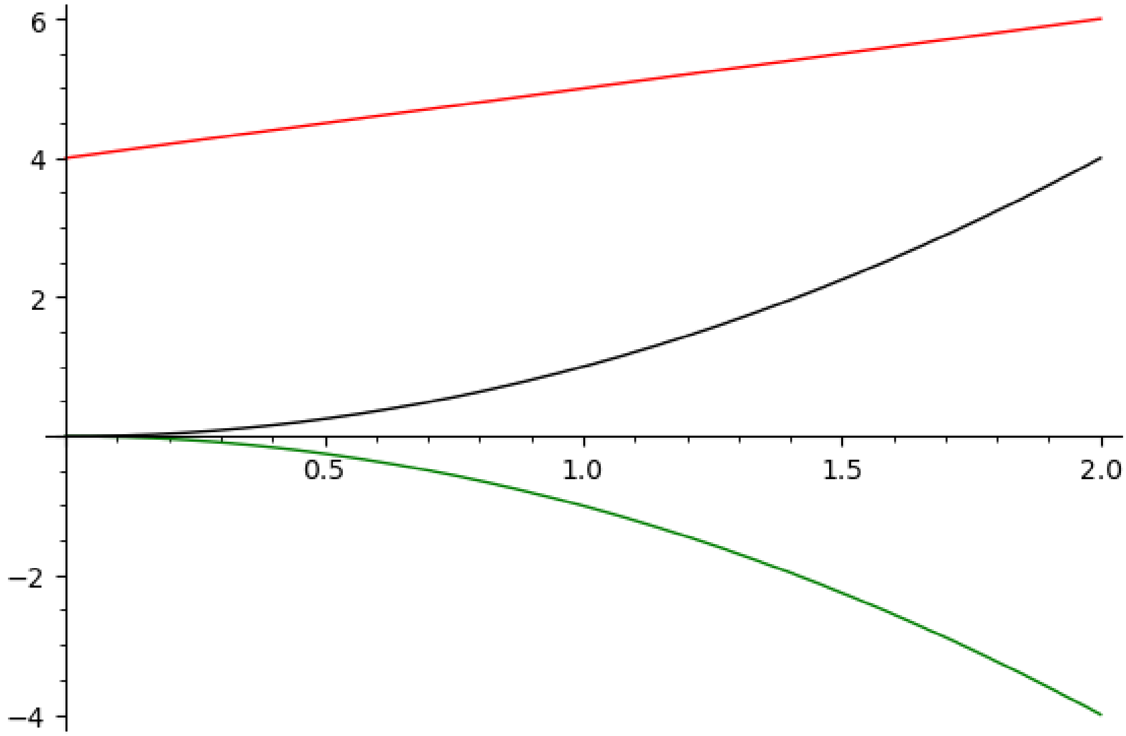


3.Draw the combine graph of $y = -x^2, y = x^2, y = 4 + x$

In [8]:

```
p1=plot(-x^2,0,2,color="green")
p2=plot(+x^2,0,2,color="black")
p3=plot(4+x,0,2,color="red")
p1+p2+p3
```

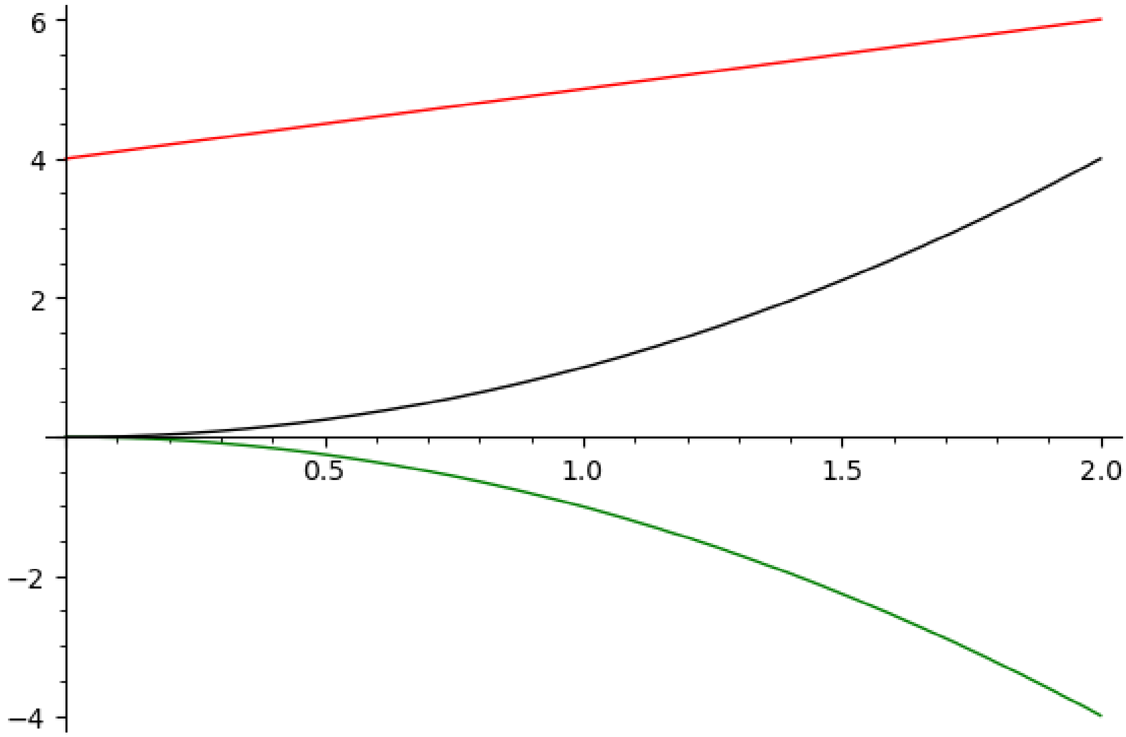
Out[8]:



In [10]:

```
plot(-x^2,0,2,color="green")+plot(+x^2,0,2,color="black")+plot(4+x,0,2,colo
r="red")
# Another method of drawing combine graph.
```

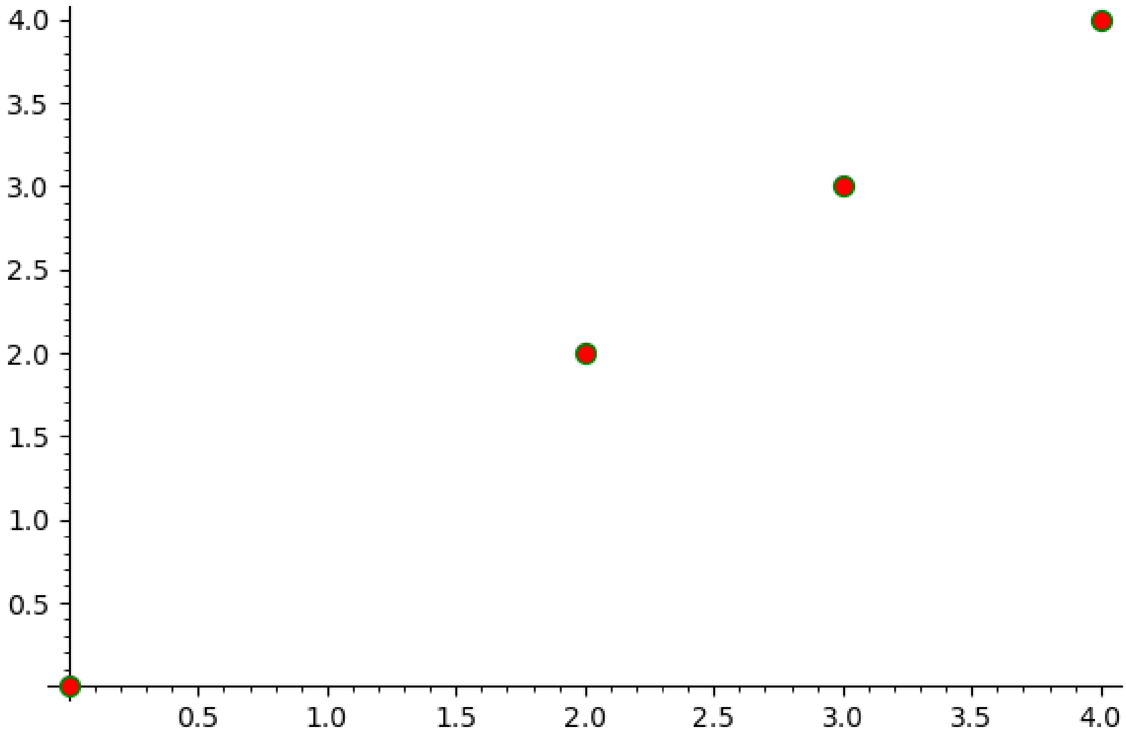
Out[10]:



4.Scatter plot

In [11]:
```
data=[(0,0),(2,2),(3,3),(4,4)]# L gives list
scatter_plot(data,facecolor='red',edgecolor='green')
```

Out[11]:

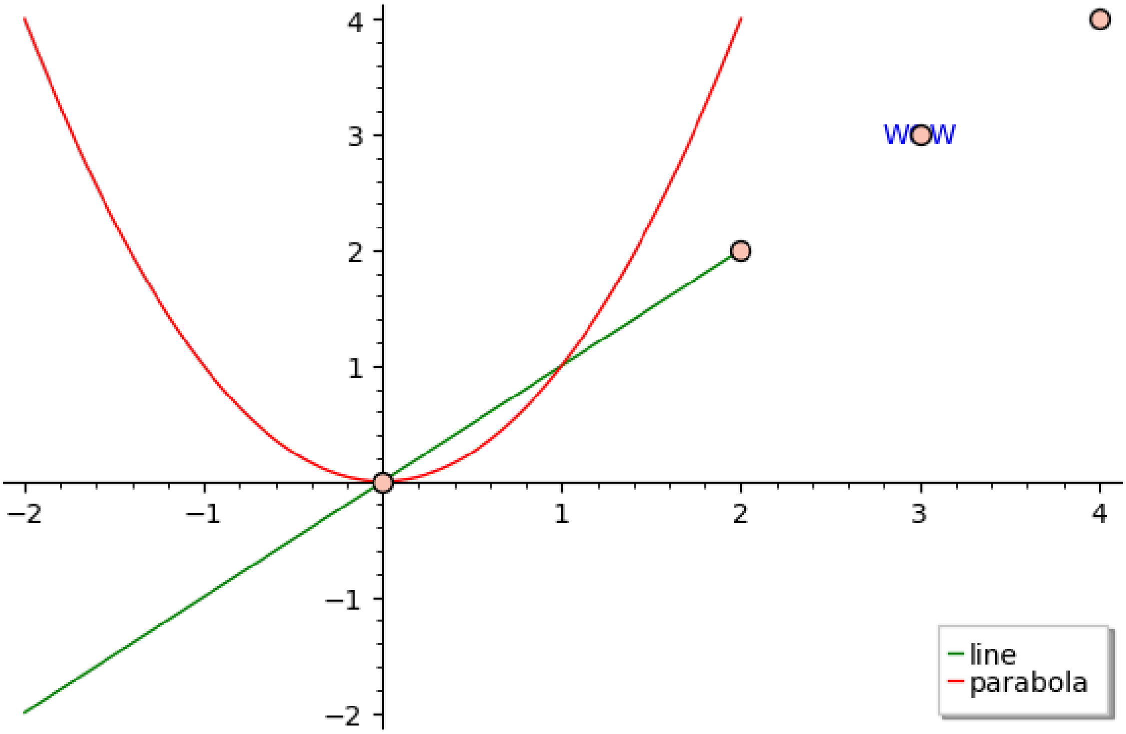


In [13]:
```
scatter_plot?
```

5.combine graphs

In [16]:
```
a=[(0,0),(2,2),(3,3),(4,4)]
scatter_plot(a)+plot(x,(x,-2,2),legend_label='line',color= 'green')+plot(x^
2,(x,-2,2),legend_label='parabola',color="red")+text('WOW',(3,3))
```

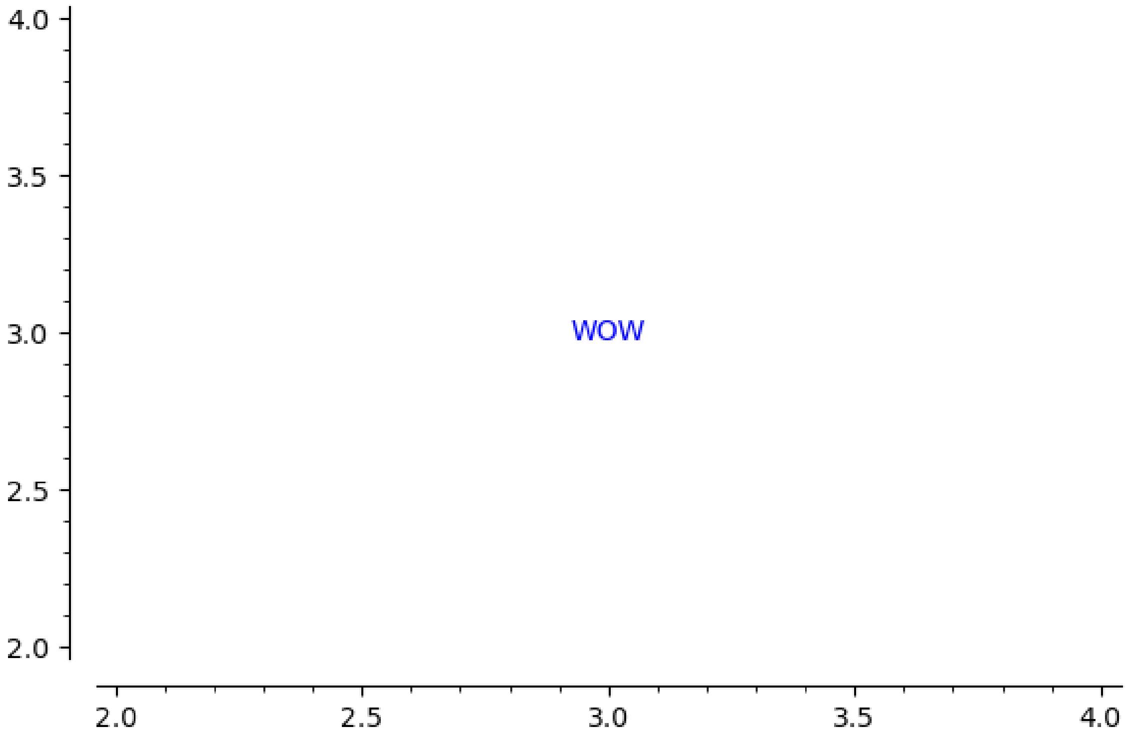
Out[16]:



6.text graph

In [17]:
```
text('WOW',(3,3))
```

Out[17]:

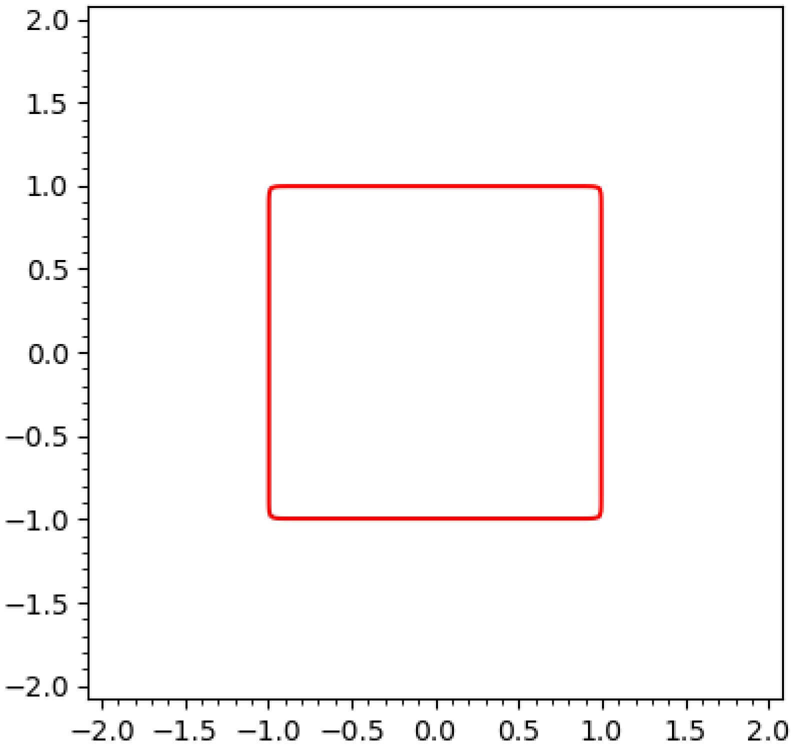


Implicit plot

1.Plot $x^n + y^n - 1 = 0$ using different values of n .

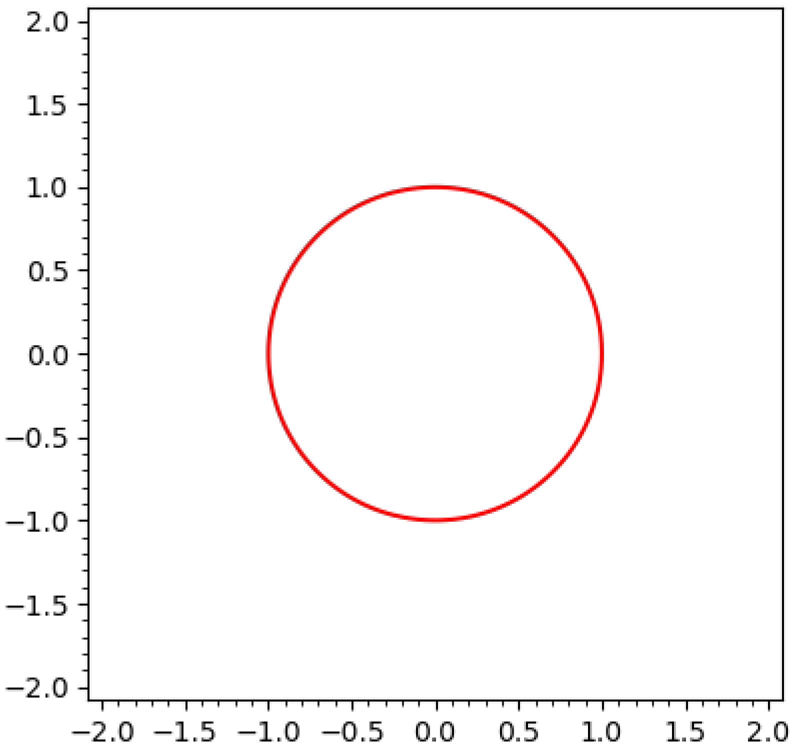
In [18]:
```
var('x,y')
n=50
f(x,y)=x^n+y^n-1
implicit_plot(f,(x,-2,2),(y,-2,2),color='red')
#n=even gives closed graph h #n=odd gives open graph
```

Out[18]:



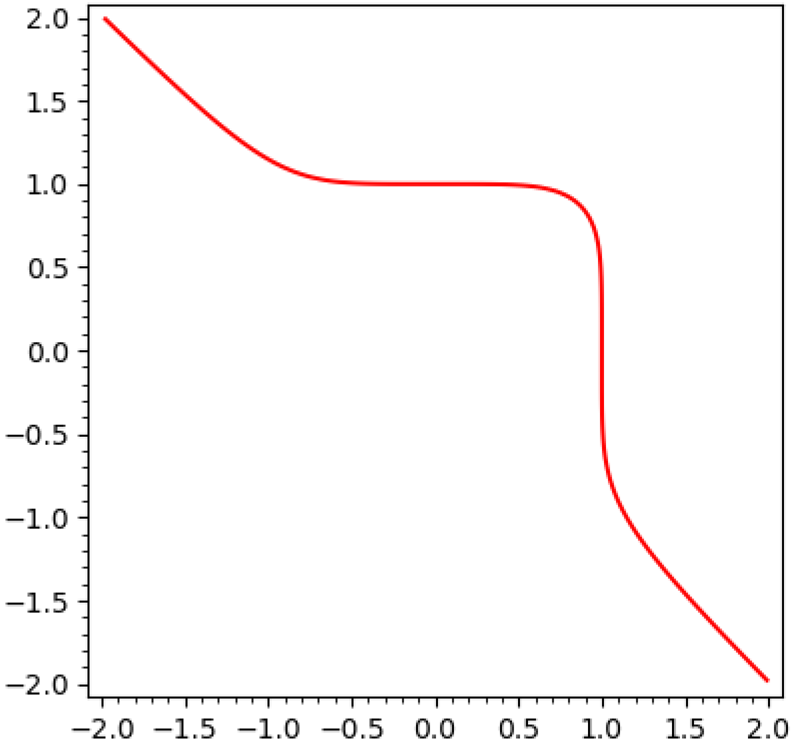
In [19]:
```
var('x,y')
n=2
f(x,y)=x^n+y^n-1
implicit_plot(f,(x,-2,2),(y,-2,2),color='red')
```

Out[19]:



In [20]:
```
var('x,y')
n=5
f(x,y)=x^n+y^n-1
implicit_plot(f,(x,-2,2),(y,-2,2),color='red')
#n=odd gives open curve
```

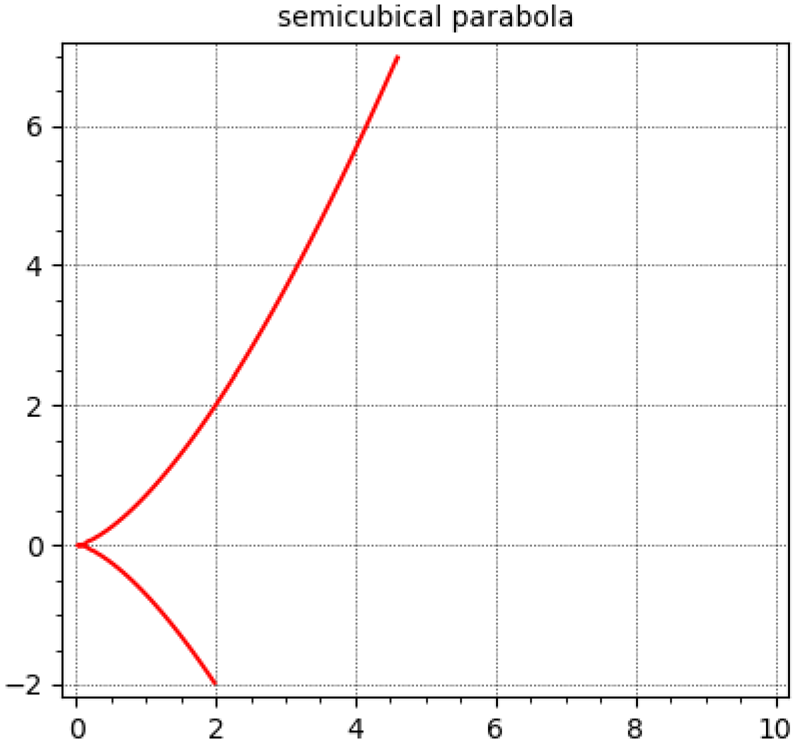
Out[20]:



2.Plot $2y^2 - x^3 = 0$ between the mentioned range.

In [22]:
```
var('x,y')
f(x,y)=2*y^2-x^3
implicit_plot(f,(x,0,10),(y,-2,7),color='red',title="semicubical parabola",
gridlines='True')
```

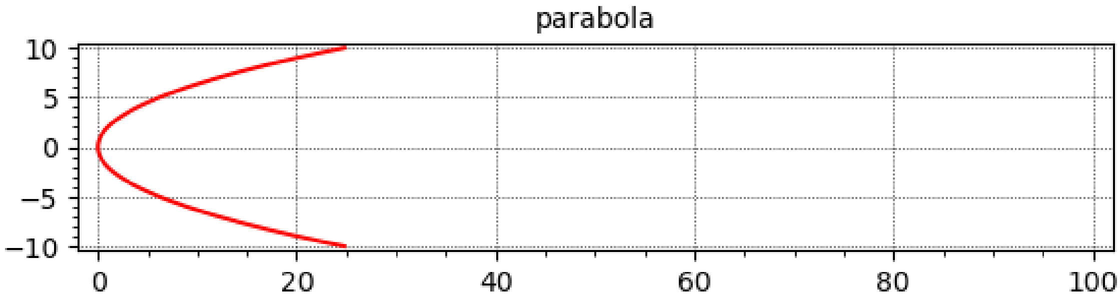
Out[22]:



3.Plot in the range $y^2 - 4x = 0$ in Range $(x, 0, 100), (y, -10, 10)$

In [24]:
```
var('x,y')
f(x,y)=y^2-4*x
implicit_plot(f,(x,0,100),(y,-10,10),color='red',title="parabola",gridlines
='True')
```

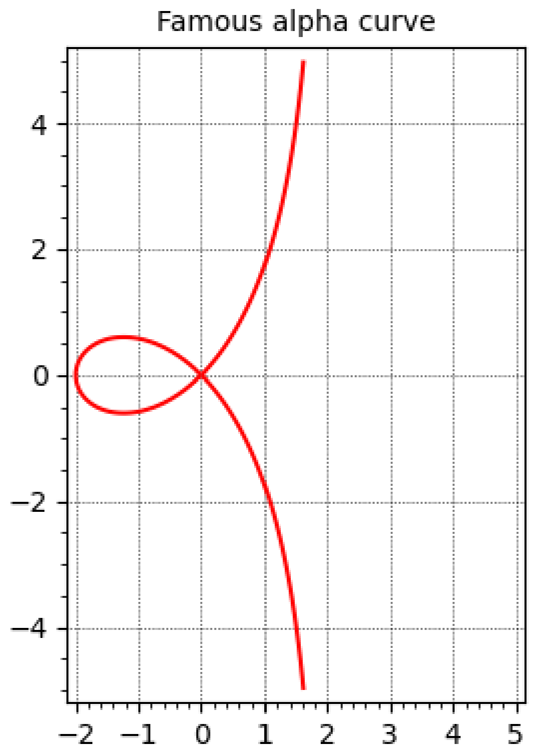
Out[24]:



4.Plot graph of $y^2(2-x) - x^2(2+x) = 0$

In [25]:
```
var('x,y')
f(x,y)=y^2*(2-x)-x^2*(2+x)
implicit_plot(f,(x,-2,5),(y,-5,5),color='red',title="Famous alpha curve",gr
idlines='True')
```

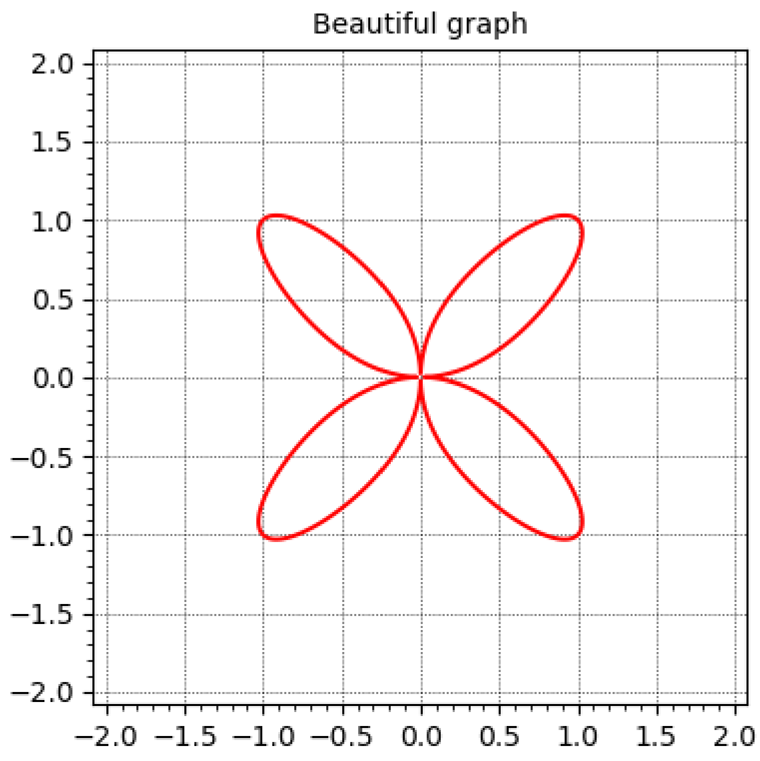
Out[25]:



5.Plot the graph of $x^6 + y^6 - 2x^2y^2 = 0$

In [28]:
```
var('x,y')
f(x,y)=x^6+y^6-2*x^2*y^2
implicit_plot(f,(x,-2,2),(y,-2,2),color='red',title="Beautiful graph",plot_
points=5000,gridlines='True')
```

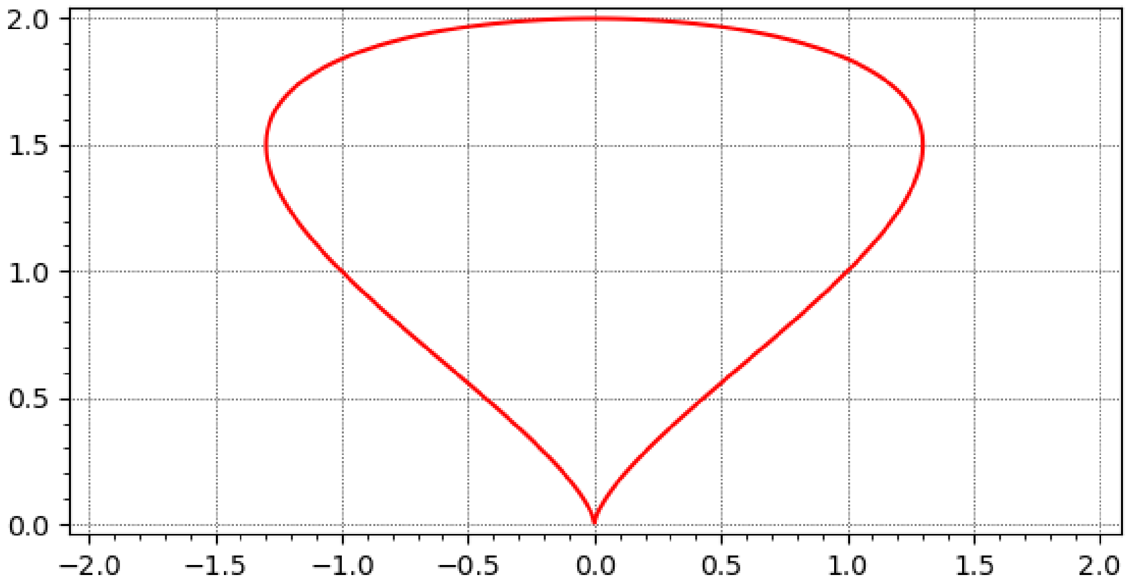
Out[28]:



6.plot the graph of $x^2 - y^3(2-y) = 0$

In [29]:
```
var('x,y')
f(x,y)=x^2-y^3*(2-y)
implicit_plot(f,(x,-2,2),(y,0,2),color='red',title="",plot_points=5000,grid
lines='True')
```

Out[29]:

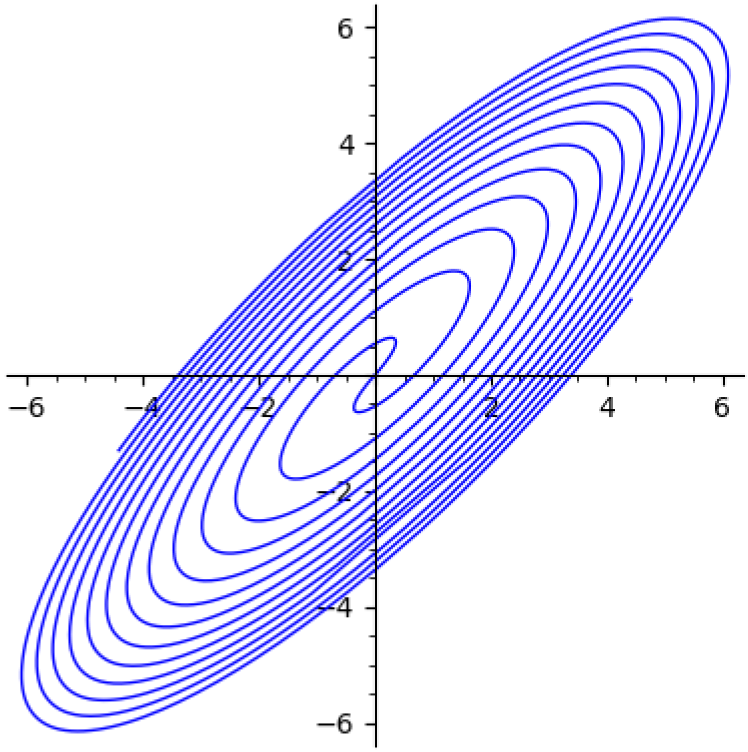


Parametric Curves in 2D

1.plot the parametric curve $x = tsin(1-t^2), y = tcos(t^2)$

In [30]:
```
var('t')
parametric_plot([t*sin(1-t^2),t*cos(t^2)],(t,-2*pi,2*pi))
```

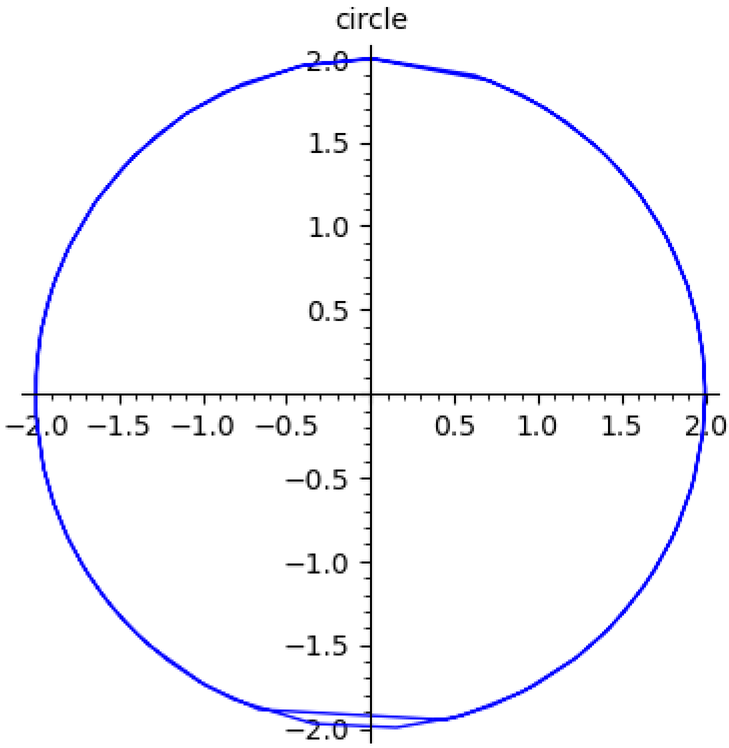
Out[30]:



2.plot the graph of $x = 2sint, y = 2cost$

In [31]:
```
var('t')
parametric_plot([2*sin(t),2*cos(t)],(t,-2*pi,2*pi),title = "circle",plot_po
ints=20)
```

Out[31]:



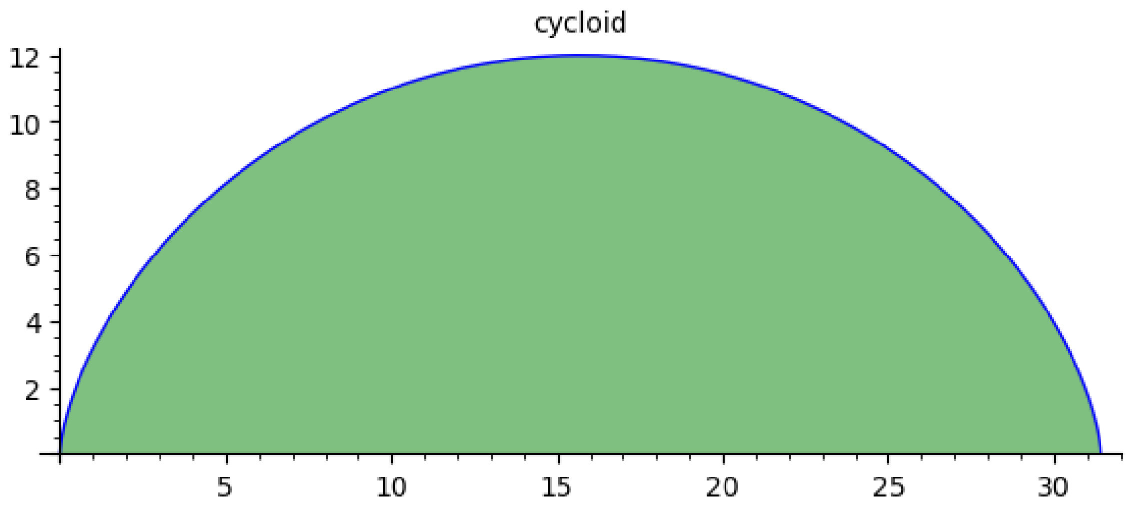
In [32]:
```
plot_points?
```

```
Object `plot_points` not found.
```

3.plot the graph of $x = 2(t - sint), y = 2(1 - cost)$

In [34]:
```
var('t')
parametric_plot([5*(t-sin(t)),6*(1-cos(t))],(t,0,2*pi),title="cycloid",fill
color="green",fill=6*(1-cos(t)))
```

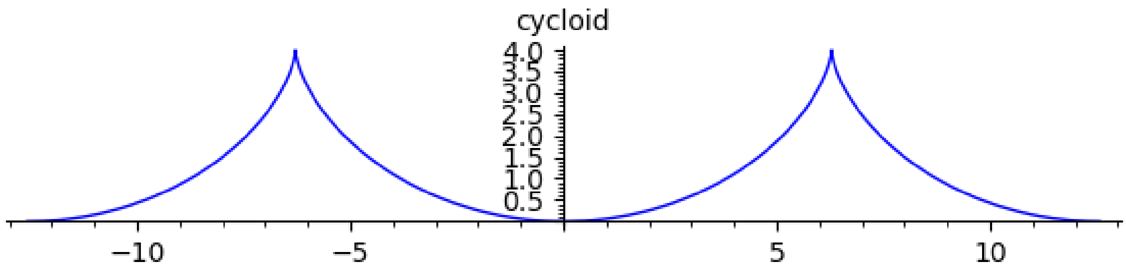
Out[34]:



4.plot the graph of $x = 2(t + sint), y = 2(1 - cost)$

In [35]:
```
var('t')
parametric_plot([2*(t+sin(t)),2*(1-cos(t))],(t,-2*pi,2*pi),title="cycloid")
```

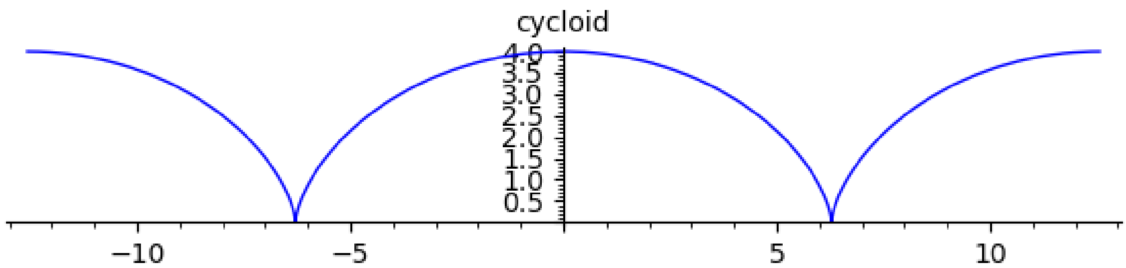
Out[35]:



5.plot the graph of $x = 2(t + sint), y = 2(1 + cost)$

In [36]:
```
var('t')
parametric_plot([2*(t+sin(t)),2*(1+cos(t))],(t,-2*pi,2*pi),title="cycloid")
```

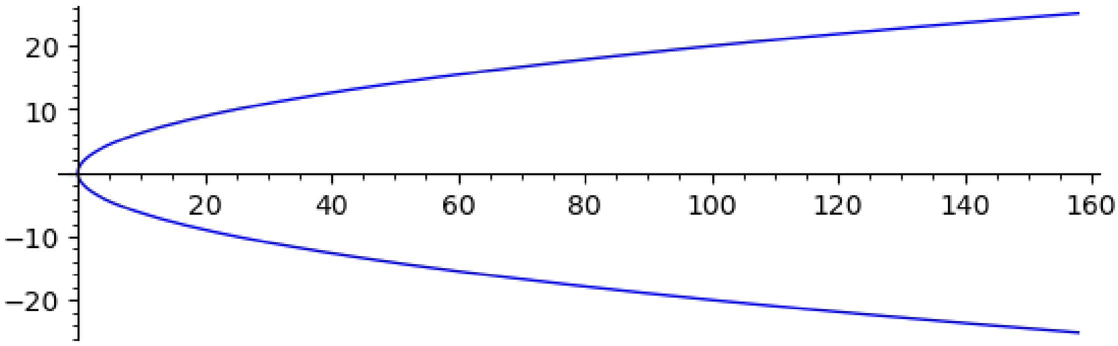
Out[36]:



6.plot the graph of $x = 4t^2, y = 4t$

In [37]:
```
var('t')
parametric_plot([4*t^2,4*t],(t,-2*pi,2*pi))
```

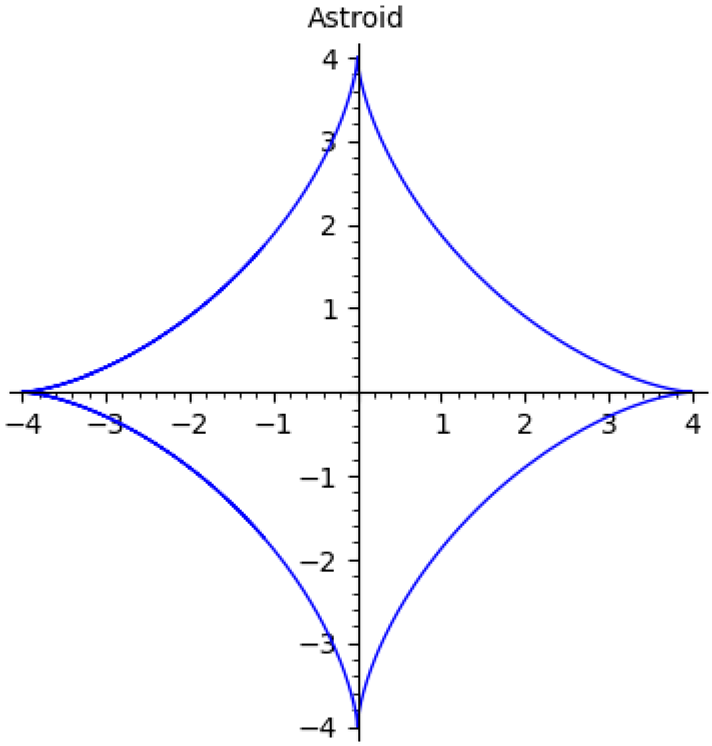
Out[37]:



7.plot the graph of $x = 4cos^3t, y = 4sint^3t$

In [38]:
```
var('t')
parametric_plot([4*cos(t)^3,4*sin(t)^3],(-4,4),title="Astroid")
```

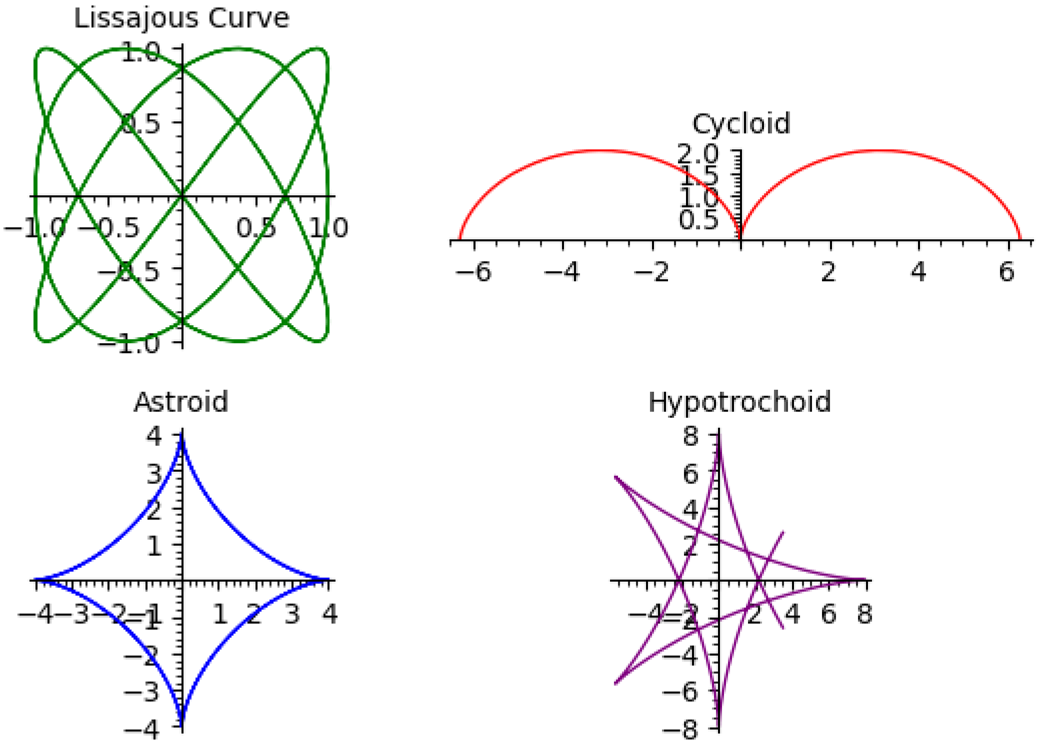
Out[38]:



Arrange the graphs of
$i)x = sin3t, y = sin4tii)x = t - sint, y = 1 - costiii)x = 4cos^3t, y = 4sin^3tiv)x = 5cost + 3c$
using a two by two grid.

In [39]:
```
var('t, x, y')
# 1. Lissajous curve
p1 = parametric_plot(
[sin(3*t), sin(4*t)],
(t, -2*pi, 2*pi),
color="green",
title="Lissajous Curve"
)
# 2. Cycloid
p2 = parametric_plot(
[t - sin(t), 1 - cos(t)],
(t, -2*pi, 2*pi),
color="red",
title="Cycloid"
)
# 3. Astroid
p3 = parametric_plot(
[4*cos(t)^3, 4*sin(t)^3],
(t, -2*pi, 2*pi),
color="blue",
title="Astroid"
)
# 4. Hypotrochoid
p4 = parametric_plot(
[5*cos(t) + 3*cos(5*t/3),
5*sin(t) - 3*sin(5*t/3)],
(t, -2*pi, 2*pi),
color="purple",
title="Hypotrochoid"
)
L = [p1, p2, p3, p4]
graphics_array(L, 2, 2)
```

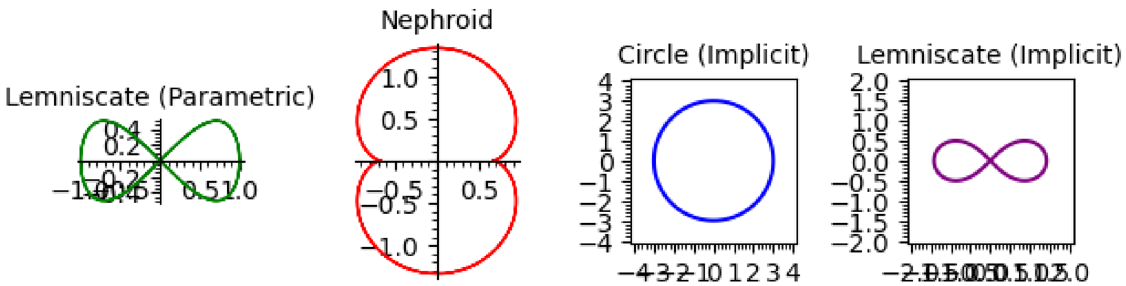
Out[39]:



Arrange the graphs of $i) x = cost, y = sintcost$ $ii) x - cost - 1/3cos3t, y = sint = 1/3sin3t$ $iii) x^2 + y^2 = 9$ $iv) (x^2 + y^2)^2 = 2(x^2 - y^2)$ using a one by four grid.

In [41]:
```
var('t, x, y')
# Parametric: Lemniscate of Gerono
p1 = parametric_plot(
[cos(t), sin(t)*cos(t)],
(t, -2*pi, 2*pi),
color="green",
title="Lemniscate (Parametric)"
)
# Parametric: Nephroid
p2 = parametric_plot(
[cos(t) - cos(3*t)/3,
sin(t) - sin(3*t)/3],
(t, -2*pi, 2*pi),
color="red",
title="Nephroid"
)
# Implicit: Circle
p3 = implicit_plot(
x^2 + y^2 == 9,
(x, -4, 4), (y, -4, 4),
color="blue",
title="Circle (Implicit)"
)
# Implicit: Lemniscate of Bernoulli
p4 = implicit_plot(
(x^2 + y^2)^2 == 2*(x^2 - y^2),
(x, -2, 2), (y, -2, 2),
color="purple",
title="Lemniscate (Implicit)"
)
L = [p1, p2, p3, p4]
graphics_array(L, 1, 4)
```

Out[41]:

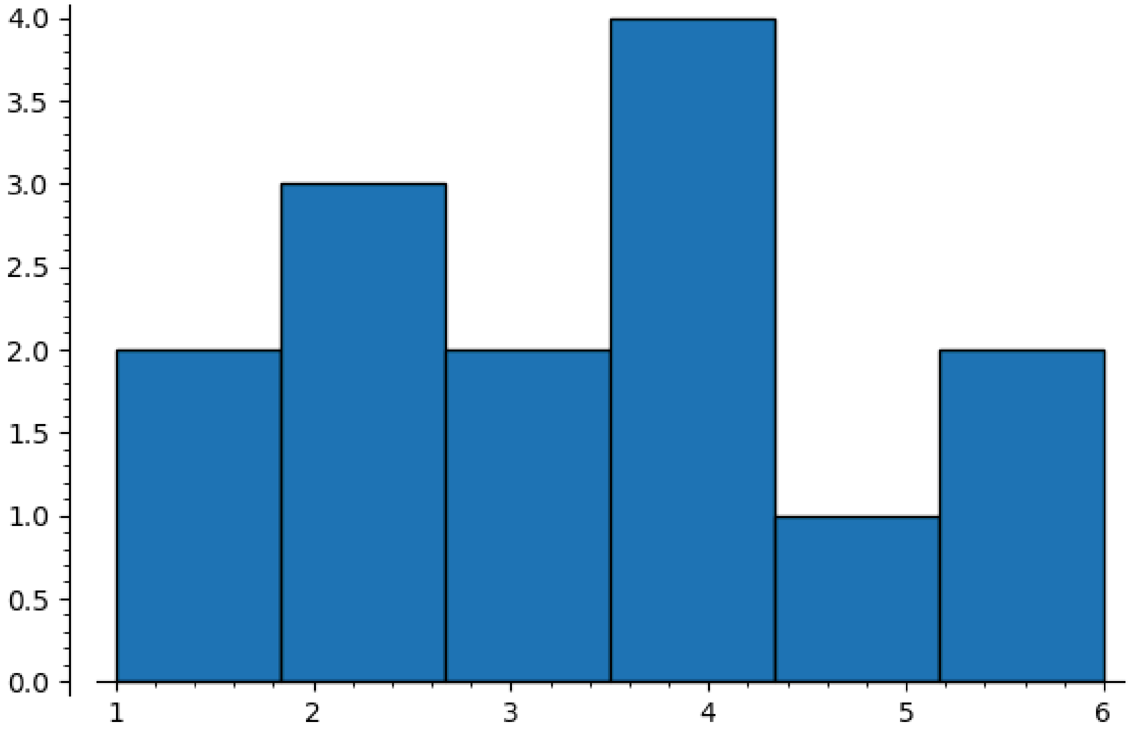


Histogram

1. Draw the Histogram for the data set {1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 6}

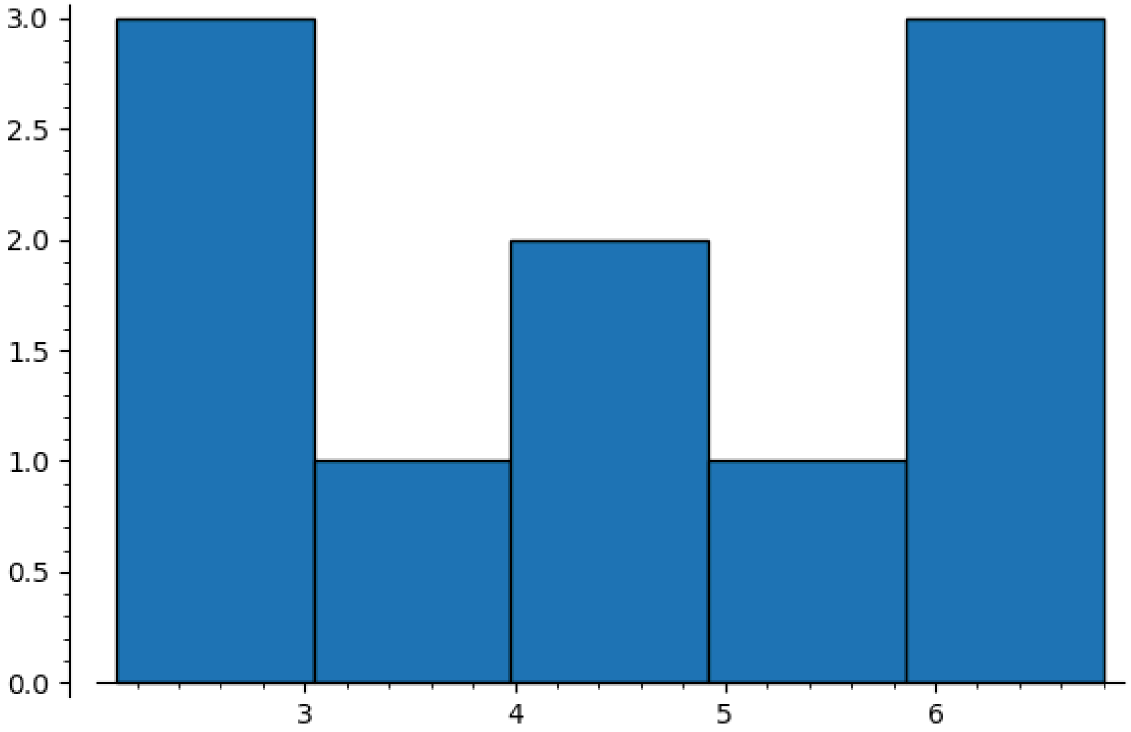
In [42]:
```
# Discrete data
data = [ 1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 5, 6,6]
hist = histogram(data,bins=6)
hist
```

Out[42]:



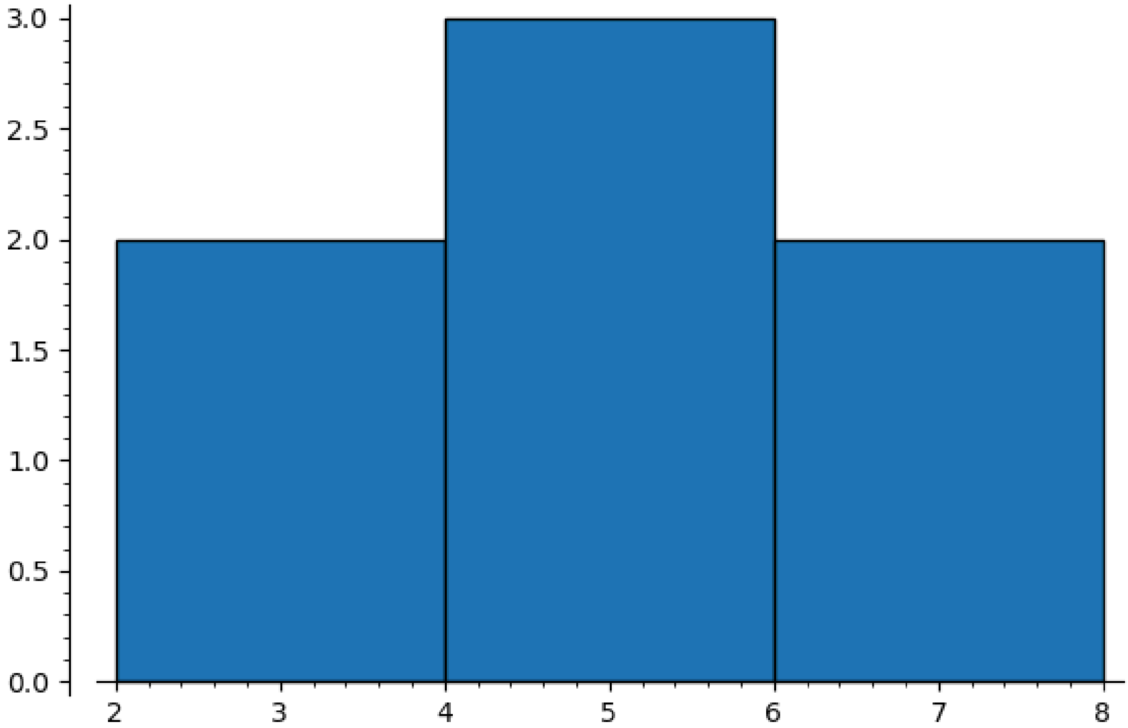
In [43]:
```
# Sample continuous data
data1 = [2.1, 2.5, 3.0, 3.2, 4.1, 4.7, 5.3, 5.9, 6.0, 6.8]
# Histogram with specified number of bins
hist = histogram(data1, bins=5)
hist
```

Out[43]:



In [44]:
```
#customized class intervals
# Continuous data
data1 = [ 2.5, 3.0, 4.1, 4.7, 5.3, 6.0, 6.8]
# Custom class intervals (bin edges)
bins = [2, 4, 6, 8]
# Histogram with customized frequency classes
hist = histogram(data1, bins=bins)
hist
```

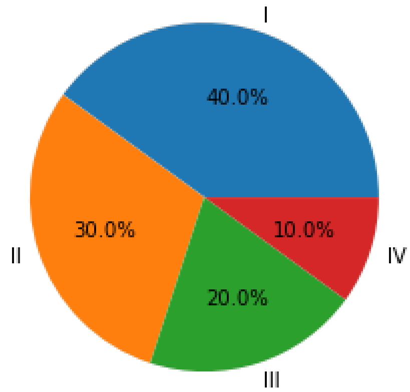
Out[44]:



Pie Chart

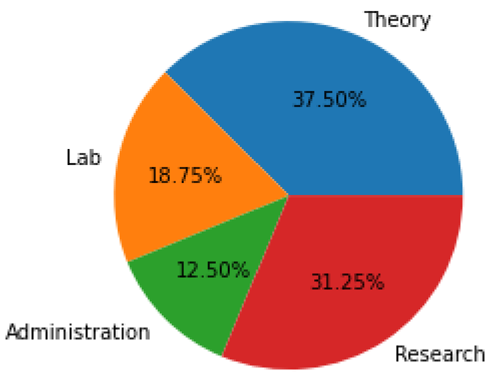
1. A data shows the following percentage of Students were placed in four different classes in an Exam. Draw a Pie chart to visualize the data\ Class: I II III IV \ Students: 40 30 20 10

In [48]:
```
import matplotlib.pyplot as plt
plt.pie([40, 30, 20, 10], labels=['I','II','III','IV'], autopct='%1.1f%%')
plt.show()
```



The Workload of a Faculty is distributed as Theory: 6 hour Lab: 3 hour Administration: 2 hour Research: 5 hours Draw a pie chart to visualize the distribution of the workload.

In [49]:
```
import matplotlib.pyplot as plt
plt.pie([6, 3, 2, 5], labels=['Theory','Lab','Administration','Research'],
autopct='%1.2f%%')
plt.show()
```

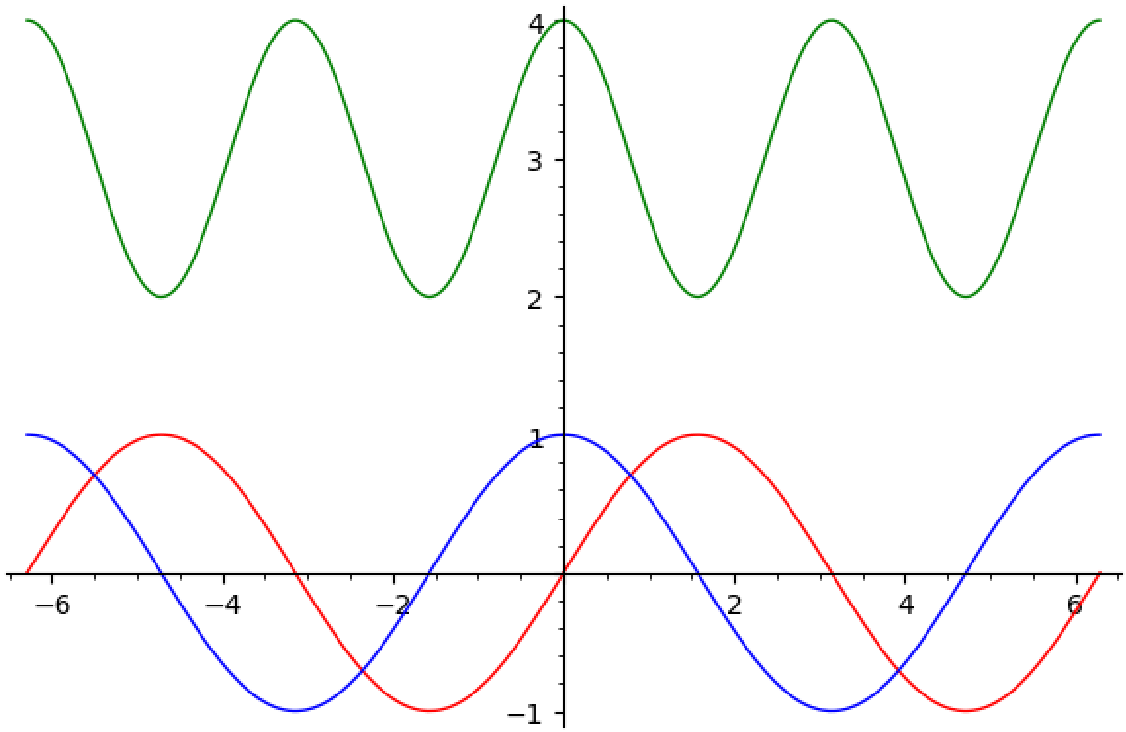


Exercises

1) Draw the combine graph of y=sinx,y=cosx and y=3+cos2x with different colors and legends.

In [60]:
```
p1=plot(sin(x),-2*pi,2*pi, color="red")
p2=plot(cos(x),-2*pi,2*pi, color="blue")
p3=plot(3+cos(2*x),-2*pi,2*pi, color="green")
p1+p2+p3
```

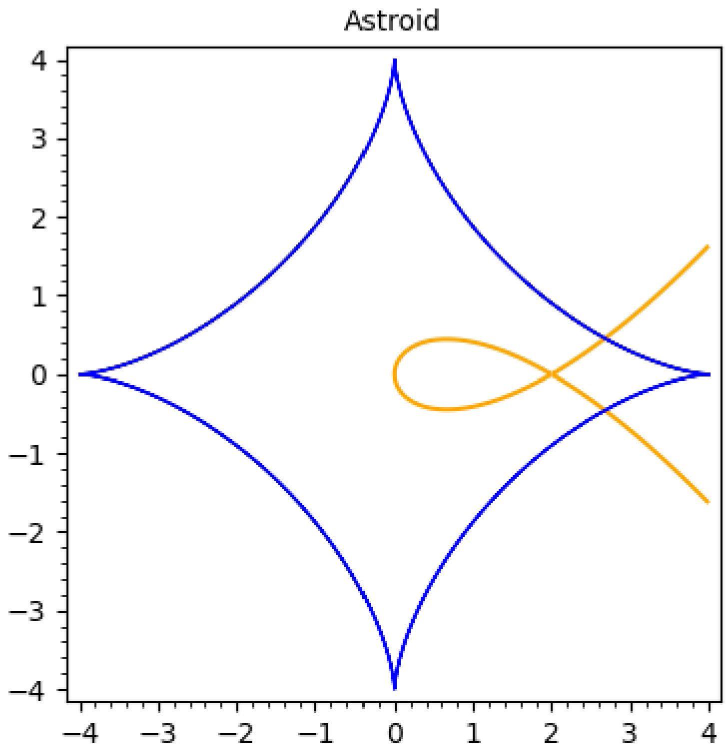
Out[60]:



2) Draw the graph of 1) Astroid $(x)^2/3 + (y)^2/3 = (4)^2/3$ and 2)$6y^2 = x(x-2)^2$with different colors and gridlines and figuresize.

In [65]:
```
p1 = parametric_plot(
[4*cos(t)^3, 4*sin(t)^3],
(t, -2*pi, 2*pi),
color="blue",
title="Astroid"
)
p2=implicit_plot(6*y^2 == x*(x - 2)^2, (x, -4, 4), (y, -3, 3),
                        color='orange')
p1+p2
```

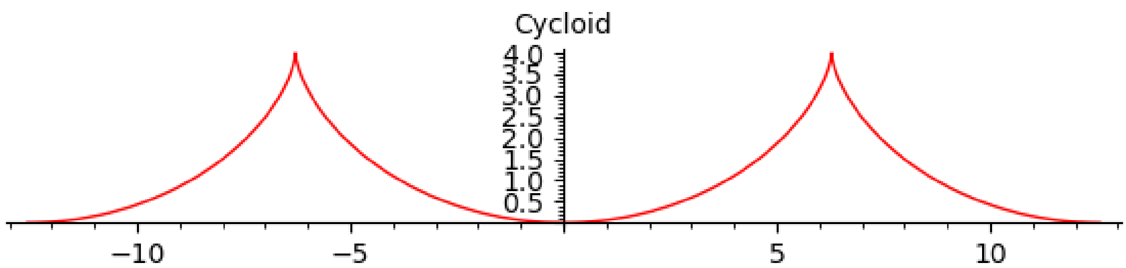
Out[65]:



3) Draw the parametric curve of cycloid x = 2(t + sint), y = 2(1 − cost) with proper legend and title.

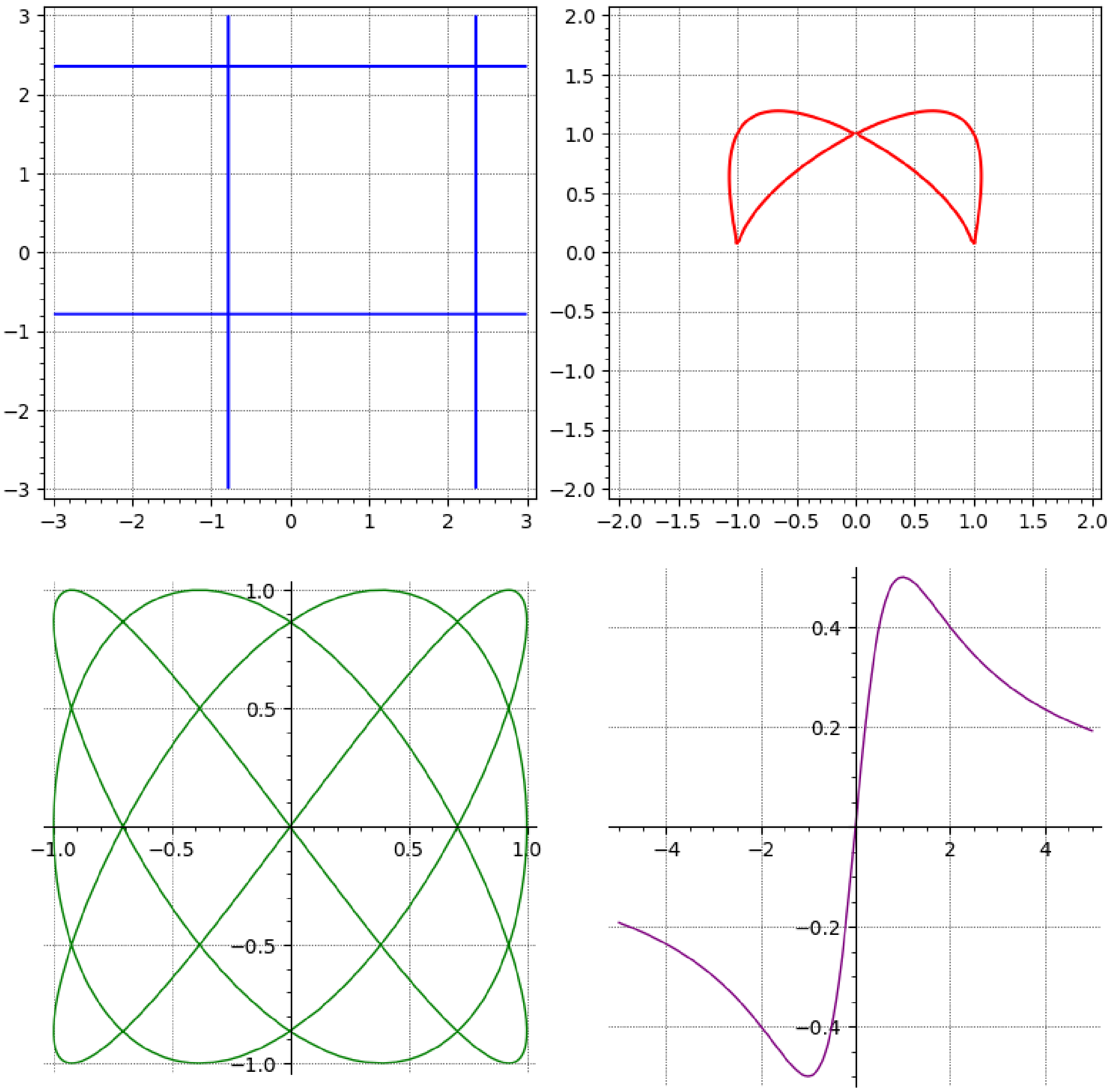
In [72]:
```
p1=parametric_plot(
[2*(t+sin(t)), 2*(1 - cos(t))],
(t, -2*pi, 2*pi),
color="red",
title="Cycloid"
)
p1
```

Out[72]:



4) Draw the graph of (a)$sin(x+y)+cos(x-y)=0$, (b)$(x^2+y^2-1)^2=x^2y^3$ within appropriate range. (c) Plot the Lissajous figure x=sin(3t), y=sin(4t) and (d)$y=x/x^2+1$ and arrange the in $2x2$ and grids

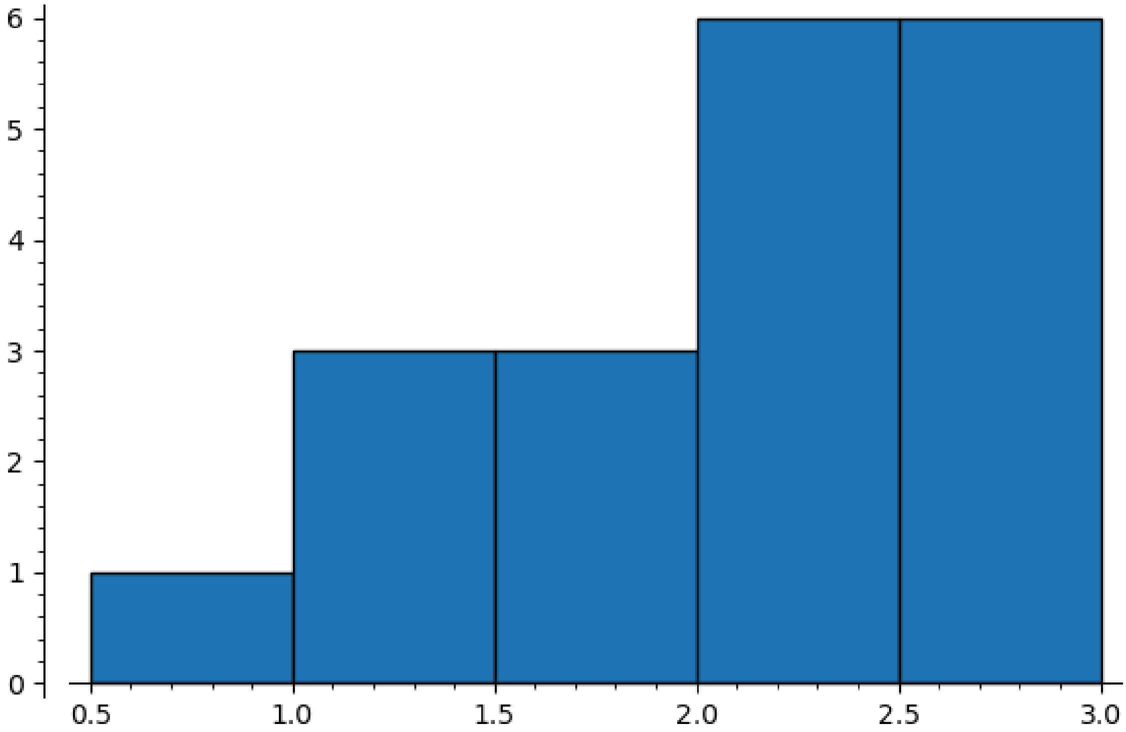
In [73]:
```
x, y, t = var('x y t')
p1 = implicit_plot(sin(x+y)+cos(x-y)==0, (x,-3,3), (y,-3,3), color="blue")
p2 = implicit_plot((x^2+y^2-1)^2==x^2*y^3, (x,-2,2), (y,-2,2), color="red")
p3 = parametric_plot((sin(3*t), sin(4*t)), (t,0,2*pi), color="green")
p4 = plot(x/(x^2+1), (x,-5,5), color="purple")
graphics_array([[p1,p2],[p3,p4]]).show(gridlines=True, figsize=[8,8])
```



5) The following are the waiting times (in minutes) of 20 customers at a service counter: $1/2, 1, 3/2, 2, 5/2, 2, 3, 3/2, 2, 5/2, 1, 3, 1, 2, 5/2, 3, 3/2, 2, 2.$ Draw a Histogram with customizedbins.

In [77]:
```
data = [1/2,1,3/2,2,5/2,2,3,3/2,2,5/2,1,3,1,2,5/2,3,3/2,2,2]
hist = histogram(data,bins=5)
hist
```

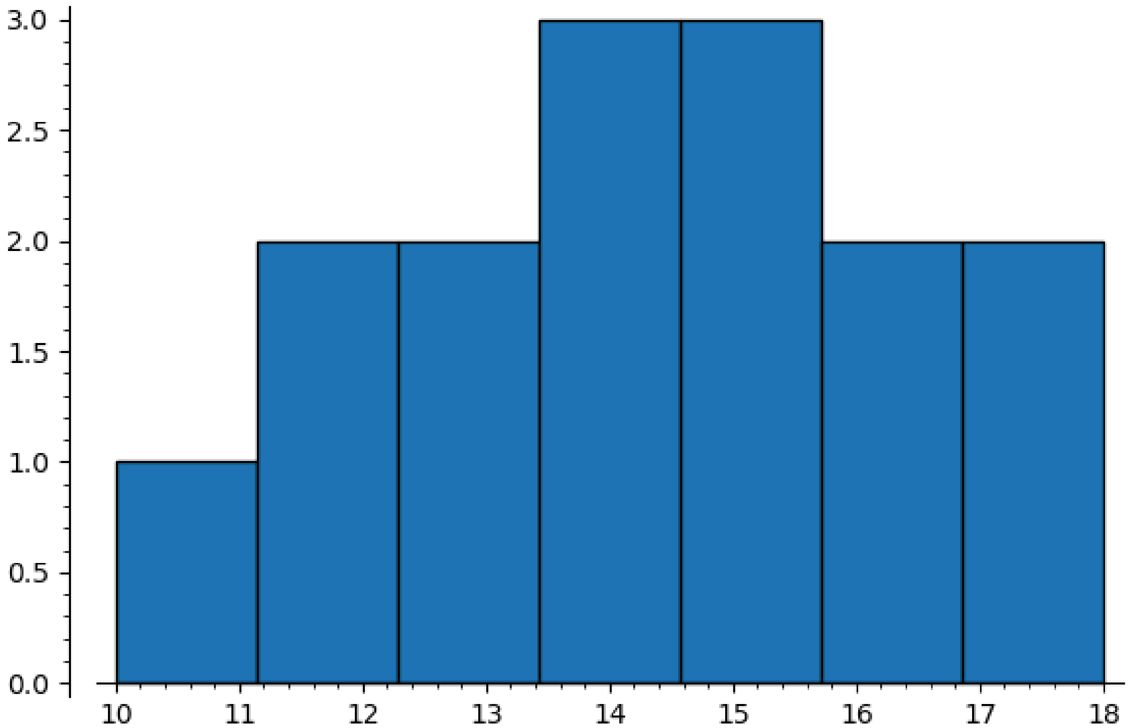
Out[77]:



6) The following are the marks obtained by 15 students in a quiz:
12,15,14,10,18,16,14,15,13,17,12,14,16,15,13 Draw a histogram to represnt the data.

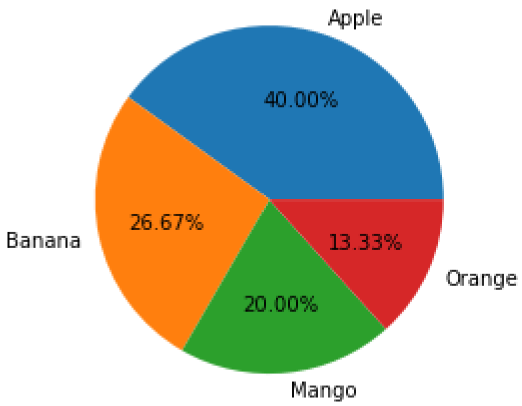
In [79]:
```
data=[12,15,14,10,18,16,14,15,13,17,12,14,16,15,13]
hist=histogram(data,bins=7)
hist
```

Out[79]:



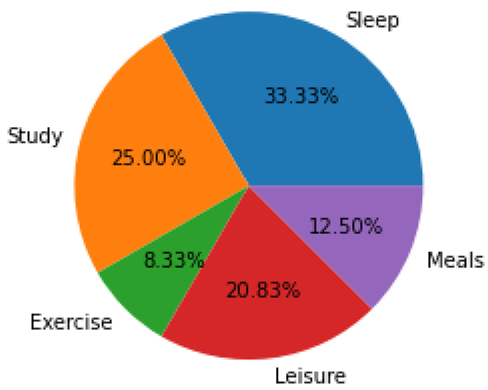
7) A survey of 30 people about their favorite fruit gave the following results: Apple: 12 Banana: 8 Mango: 6 Orange: 4 Draw a pie chart representing this data. Show fractional values on the chart up to 2 decimals.

In [82]:
```
import matplotlib.pyplot as plt
plt.pie([12,8,6,4], labels=['Apple','Banana','Mango','Orange'], autopct='%1.2f%%')
plt.show()
```



8) A student records hours spent on daily activities in a 24-hour day: Sleep: 8 Study: 6 Exercise: 2 Leisure: 5 Meals: 3 Draw a pie chart showing the distribution of time spent on activities. Represent fractions (not percentages) on the slices up to 2 decimal places.

In [84]:
```
import matplotlib.pyplot as plt
plt.pie([8,6,2,5,3], labels=['Sleep','Study','Exercise','Leisure','Meals'],
autopct='%1.2f%%')
plt.show()
```



In [ ]: