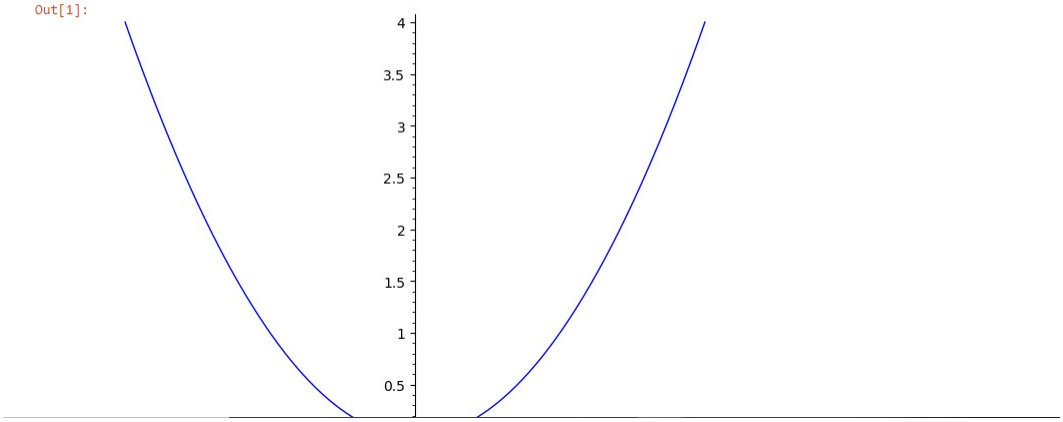
We first use Sagemath to calculate curvatures of plane curves and familiarize with some basic commands:

- to input a function/formula, one first defines the variables (eg `var('x')`), then the function (eg `f = sin(x)`).
- To differentiate a function with respect to a variable, one may use the command '`diff`' (e.g. `diff(f,x)`, or `diff(sin(x), x)`). Higher derivatives may be computed as well: eg `diff(f,x,2)` computes 2nd derivative wrt to x of the expression f.
- The `plot` command may be used to generate graphs (eg `plot(sin(x), (x, 0, 2*pi))`). We will also make use of the `parametric_plot` command (eg `parametric_plot( ( cos(x), sin(x), x ) , ( x, 0, 2*pi) )`).
- Also useful are the commands `.simplify() , .full_simplify() , .trig_simplify()` which can be used to simplify formulas (eg `f.full_simplify()` will simplify the expression `f`).

For example (note that to execute commands in the notebook, one uses `shift + enter`) here we plot the graph of $y = x^2$:

```
In [1]:
var('x')

f = x^2

plot( f , (x, -2, 2))
```

Out[1]:



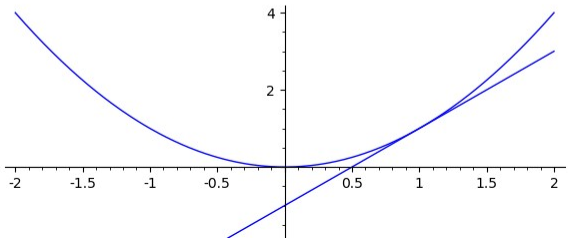
here we find derivatives of x^2 and plot the graph $y = x^2$, and its tangent line at $(1, 1)$ (using eg the command `f(x=1)` to evaluate the expression f at $x = 1$, and that '+' is used to display multiple plots simultaneously: eg `show(p1 + p2)` will display graphics items named $p1, p2$):

```
In [5]:
df = diff(f,x)
ddf = diff(f, x, 2)

show(df)
show(ddf)
```

$2x$

2

```
In [8]:
p1= plot(f , (x, -2,2))
p2 = plot(df(x=1)*(x-1) + f(x=1), (x,-2,2))

show(p1 + p2)
```



At the following link or here, you may find more commands and examples.

EXERCISE:

1. Plot the following plane curves ($a, b \geq 0$ some fixed constants) and compute their curvatures:
   $(a\cos t, b\sin t)$,
   $(a\cosh t, b\sinh t)$,
   (t, at^2),
   a Cycloid.

Also useful is a loop, allowing to repeat similar actions depending on an index, executing:

```
for i in range(5):
    print(i^2)
```

outputs $0, 1, 4, 9, 16$.
Loops may be used with Lists to store any type of indexed object. For example, executing:

```
L = [i for i in range(0,5)]
show(L)
```

outputs $[0, 1, 2, 3, 4]$. Note you may place *any* objects in these lists, for example graphic objects:

```
v = [point((i,i)) for i in range(0,5)]
show(sum(v))
```

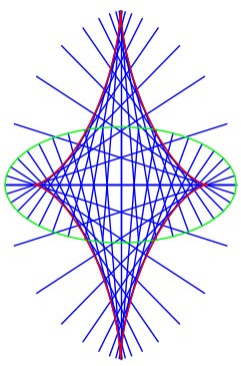
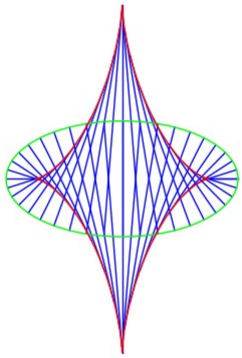
will output a graph consisting of the points $(0, 0), (1, 1), ..., (4, 4)$. Also, to access or modify a specific entry in a list, is done by say `v[1]` to access the $i = 1$ entry of the list.

For the following exercise, you may wish to also make use of the `circle`, `line`, `point` commands (eg `circle((1,1), 1)` will produce a circle of radius 1 centered at (1,1), `line([(0,0), (1,1)])` will produce a line from the origin to (1,1), `point((0,1))`) will produce a point at the given $x = 0, y = 1$ coordinates).

EXERCISE:

2. For the plane curves in the previous exercise, plot their centers of curvature and some normal lines.

an example output with an ellipse might be something like:


or


where in green is the ellipse, red the centers of curvature and in blue some normal lines (you may change colors of plots for example by adding:

```
parametric_plot( (cos(t), sin(t)), (t, 0, 2*pi), color = 'purple')
```