In [1]:
```
from sympy import *
init_printing()
```

In [2]:
```
a0,a1,a2,a3 = symbols('a_0, a_1, a_2, a_3')
EI,L,M,F = symbols('EI,L,M,F')
x = symbols('x')
```

Knowns

In [3]:
```
delta, phi = symbols('delta, phi')
```

Shape function

In [4]:
```
v = a0 + a1*x + a2*x**2 + a3*x**3
v
```

Out[4]: $a_0 + a_1 x + a_2 x^2 + a_3 x^3$

In [5]:
```
dvdx = diff(v, x)
dvdx
```

Out[5]: $a_1 + 2a_2 x + 3a_3 x^2$

In [6]:
```
d2vdx2 = diff(v,x,x)
d2vdx2
```

Out[6]: $2\left(a_2 + 3a_3 x\right)$

Boundary conditions

The beam is clamped at x=0 with deflection of zero and slope of zero

In [7]:
```
bc0 = v.subs('x', 0)
bc0
```

Out[7]: a_0

In [8]:
```
bc1 = dvdx.subs('x',0)
bc1
```

Out[8]: a_1

At x=0 the beam has displacement delta and slope phi

In [9]:
```
bc2 = v.subs('x','L')
bc2
```

Out[9]: $L^3 a_3 + L^2 a_2 + L a_1 + a_0$

In [10]:
```
bc3 = dvdx.subs('x','L')
bc3
```

Out[10]: $3L^2 a_3 + 2La_2 + a_1$

In [11]:
```
res = linsolve([bc0, bc1, bc2-delta, bc3-phi],['a_0', 'a_1','a_2','a_3'])
res
```

Out[11]: $$\left\{\left(0,\ 0,\ \frac{-L\phi + 3\delta}{L^2},\ \frac{L\phi - 2\delta}{L^3}\right)\right\}$$

And place the result in v

In [12]:
```
def repa(v):
    z = v.subs('a_0', res.args[0][0])
    z = z.subs('a_1', res.args[0][1])
    z = z.subs('a_2', res.args[0][2])
    z = z.subs('a_3', res.args[0][3])
    return z

z = repa(v)
z
```

Out[12]: $$\frac{x^2\left(-L\phi + 3\delta\right)}{L^2} + \frac{x^3\left(L\phi - 2\delta\right)}{L^3}$$

Derive force and moment from shape function and stiffness

In [13]:
```
M = EI * diff(v,x,x)
```

In [14]:
```
M
```

Out[14]: $2EI\left(a_2 + 3a_3x\right)$

In [15]:
```
F = diff(M,x)
```

In [16]:
```
F
```

Out[16]: $6EIa_3$

Check against "vergeetmenietjes" (standard formulas for deflections and slopes of cantilever beams"

Cantiliver beam with force P

In [17]:
```
MatL = M.subs('x','L')
MatL = repa(MatL)
M0 = Eq(MatL, 0)
M0
```

Out[17]: $$2EI\left(\frac{-L\phi + 3\delta}{L^2} + \frac{3\left(L\phi - 2\delta\right)}{L^2}\right) = 0$$

In [18]:
```
P = symbols('P')
FatL = F.subs('x','L')
FatL = repa(FatL)
Fp = Eq(FatL, P)
Fp
```

Out[18]:
$$\frac{6EI\left(L\phi - 2\delta\right)}{L^3} = P$$

In [19]:
```
res_P = solve([M0, Fp],'delta, phi')
res_P
```

Out[19]:
$$\left\{\delta : -\frac{L^3 P}{3EI},\ \phi : -\frac{L^2 P}{2EI}\right\}$$

This is as expected for a clamped beam with a force P acting on it

Cantiliver beam with moment Q

In [20]:
```
Q = symbols('Q')
MatL = M.subs('x','L')
MatL = repa(MatL)
MQ = Eq(MatL, Q)
MQ
```

Out[20]:
$$2EI\left(\frac{-L\phi + 3\delta}{L^2} + \frac{3\left(L\phi - 2\delta\right)}{L^2}\right) = Q$$

In [21]:
```
FatL = F.subs('x','L')
FatL = repa(FatL)
F0 = Eq(FatL, 0)
F0
```

Out[21]:
$$\frac{6EI\left(L\phi - 2\delta\right)}{L^3} = 0$$

In [22]:
```
res_Q = solve([MQ, F0],'delta, phi')
res_Q
```

Out[22]:
$$\left\{\delta : \frac{L^2 Q}{2EI},\ \phi : \frac{LQ}{EI}\right\}$$

This is as expected for a cantilever beam with a moment Q acting on it

Energy

Bending energy in a beam can be calculated as follows:

In [23]:
```
U = integrate(M**2/(2*EI),(x, 0, L))
U
```

Out[23]: $6EIL^3a_3^2 + 6EIL^2a_2a_3 + 2EILa_2^2$

In [24]:
```
simplify(U)
```

Out[24]: $2EIL\left(3L^2a_3^2 + 3La_2a_3 + a_2^2\right)$

In [25]:
```
U = repa(U)
U
```

Out[25]: $$\frac{2EI(-L\phi + 3\delta)^2}{L^3} + \frac{6EI\left(-L\phi + 3\delta\right)\left(L\phi - 2\delta\right)}{L^3} + \frac{6EI(L\phi - 2\delta)^2}{L^3}$$

Check energy for a cantiliver beam force P acting on it

In [26]:
```
UPbeam = U.subs('delta', res_P[delta])
UPbeam = UPbeam.subs('phi', res_P[phi])
```

In [27]:
```
UPbeam
```

Out[27]: $$\frac{L^3P^2}{6EI}$$

In [28]:
```
UPforce = delta * P / 2
```

In [29]:
```
UPforce = UPforce.subs('delta',res_P[delta])
UPforce
```

Out[29]: $$-\frac{L^3P^2}{6EI}$$

AS EXPECTED

In [30]:
```
UQbeam = U.subs('phi', res_Q[phi])
UQbeam = UQbeam.subs('delta', res_Q[delta])
UQbeam
```

Out[30]: $$\frac{LQ^2}{2EI}$$

In [31]:
```
UQmoment = phi * Q / 2
UQmoment = UQmoment.subs('phi',res_Q[phi])
UQmoment
```

Out[31]: $$\frac{LQ^2}{2EI}$$

AS EXPECTED

In [ ]: