```
#%%
# Imports/Setup
import sympy as sp
from sympy import eye, sin, cos, diff, Matrix, symbols, Function,
pretty_print, simplify, init_printing, latex, sign
from sympy.physics.vector import dynamicsymbols
from sympy.physics.vector.printing import vpprint, vlatex
from IPython.display import Math, display

# Define symbols
g, m, k1, k2, t, f, b, c = symbols('g, m, k1, k2, t, f, b, c')
z = dynamicsymbols('z')
zdot = z.diff(t)
zddot = zdot.diff(t)

# Define generalized coordinates
q = Matrix([[z]])
qdot = q.diff(t)
qddot = qdot.diff(t)

# Define position and velocity
P = Matrix([[0], [z], [0]])
V = P.diff(t)
```

```
#%%
# Get/Display KE
'''
    The only KE is from the mass moving, there is
    no rotation, so no w.T*J*w term...also only
    one body in this system
'''
KE = simplify(0.5*m*V.T*V)
KE = KE[0,0]
display(Math(vlatex(KE)))
```

```
#%%
# Get/Display PE
'''
    The PE is defined by springs and mass forces due
    to gravity. We only have one mass and one spring
'''
PE = m*g*z + (0.5*k1*z**2 + 0.25*k2*z**4)
display(Math(vlatex(PE)))
```

```
#%%
# Get/Display Lagrangian
'''
    Lagrangian is simply KE-PE
'''
L = simplify(KE-PE)
```

```
display(Math(vlatex(L)))
```

```
#%%
# Get/Display Generalized Forces
'''
    Generalized forces include all other forces,
    including applied forces and non-conservative
    forces, such as damping and friction
'''
f_fric = c*sign(zdot)
f_damp = b*zdot
F = Matrix([[f - f_fric - f_damp]])
display(Math(vlatex(F)))
```

```
#%%
# Get/Display EOM
LHS = simplify((L.diff(qdot)).diff(t) - L.diff(q))
RHS = F
tempEOM = LHS - RHS
EOM = simplify(sp.solve(tempEOM, (zddot)))
zdd_EOM = EOM[zddot]
eoms = Matrix([[zdd_EOM]])
display(Math(vlatex(eoms)))
```

```
# %%
# Get/Display EOM w/ Variables subbed
EOMvals = eoms.subs([(b, 0.1), (g, 9.8), (l, 0.25), (m, 0.1), (k1, 0.02), (
k2, 0.01), (tau, 3)])
display(Math(vlatex(EOMvals)))
```