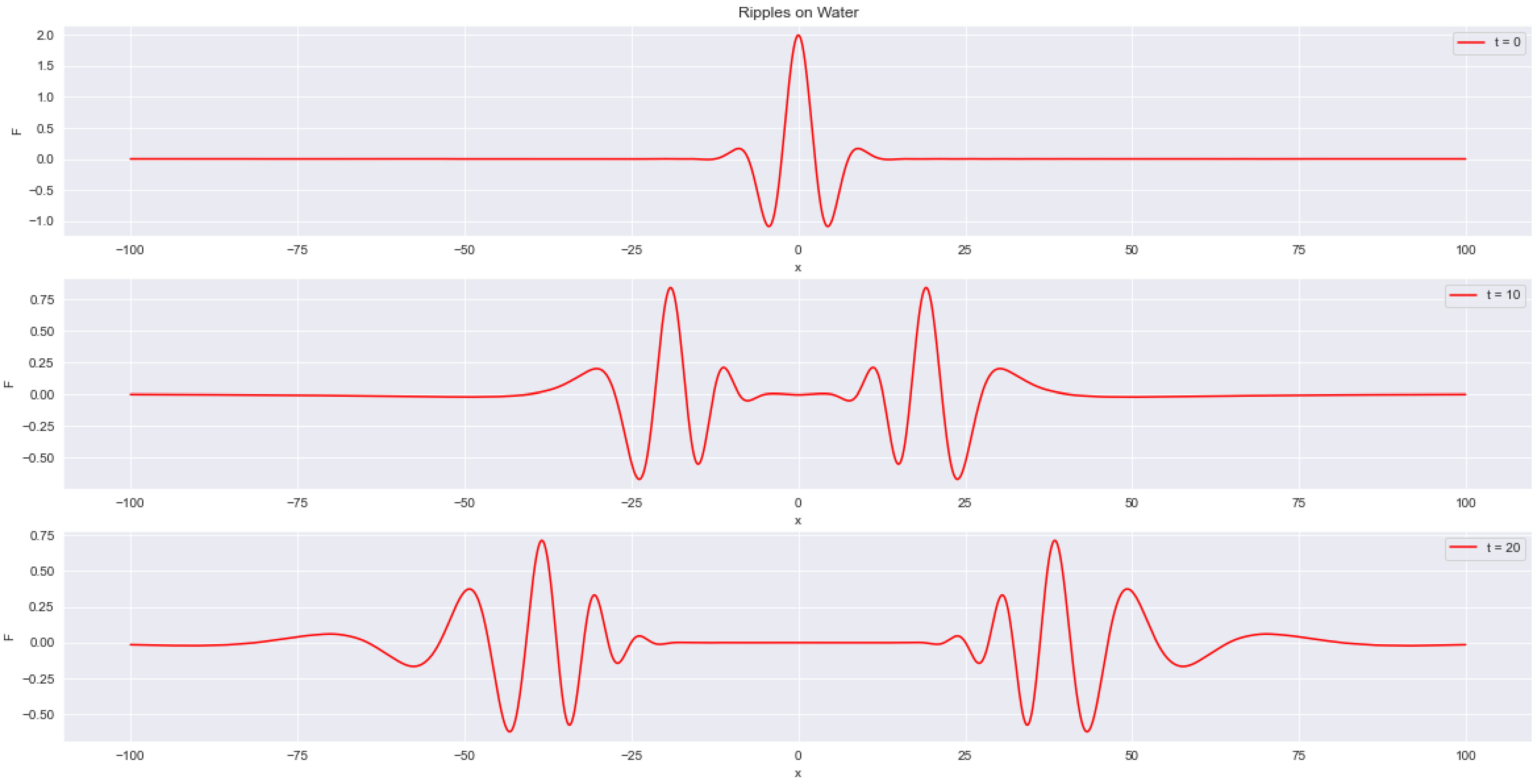
Welcome to the Course on "Numerical Methods in Physics and Mathematics"



Why we need to learn this subject ?

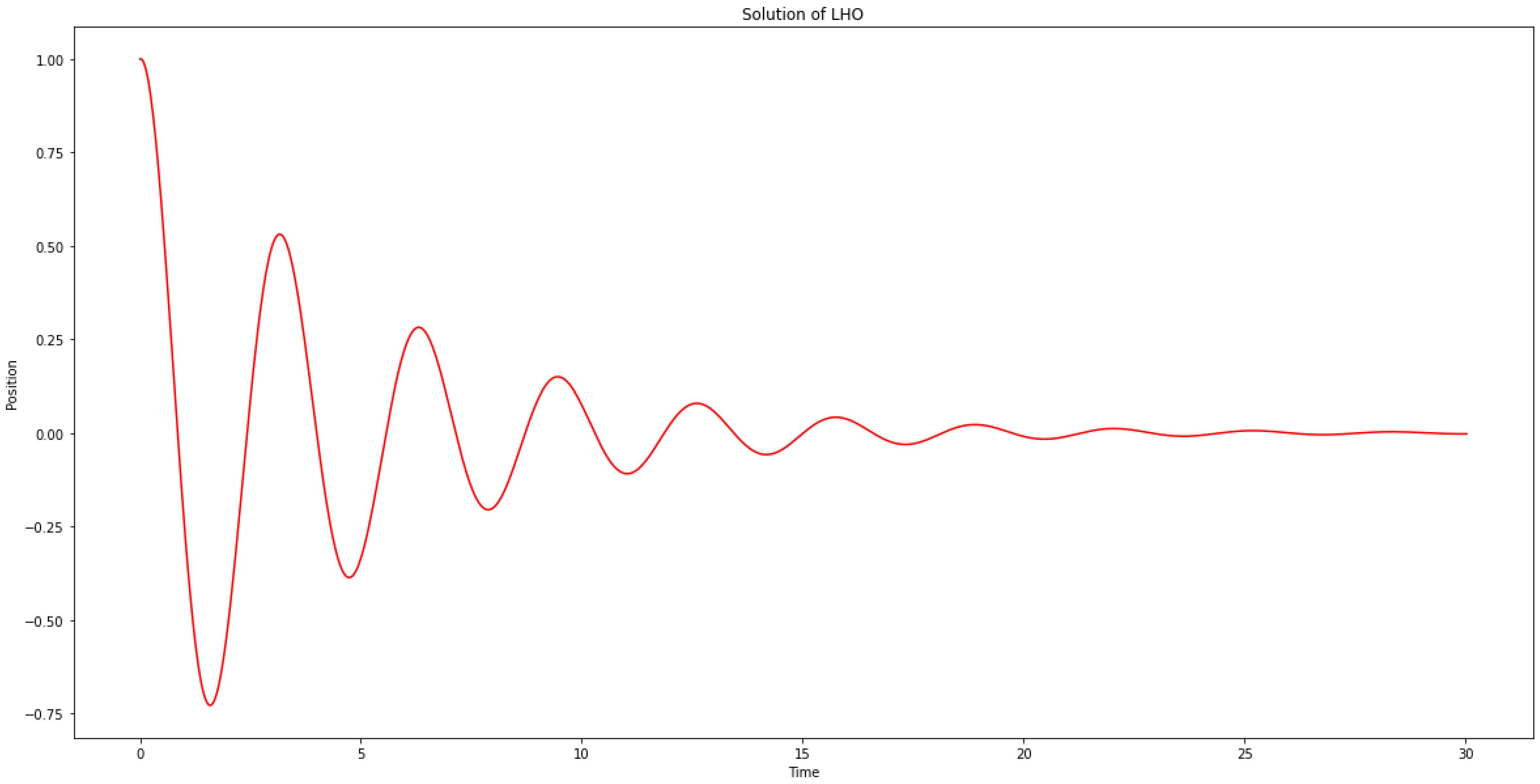
It's the basis for solving ordinary differential equations descrbing the motion of harmonic oscillators to doing black hole simulations.

In [10]:

```
import matplotlib.pyplot as plt

def f(t,x,v):
    return -w0**2*x - g*v

g = 0.4 # light Damping
w0 = 2
x0 = 1.0
v0 = 0.0
ti = 0.0
tf = 30
n = 1000
h = (tf-ti)/n
t = ti
x = x0
v = v0
xd_list = [x0]
vd_list = [v0]
td_list = [t]

for i in range(0,n+1):
    #print(t,x,v)
    x = x + h*v
    v = v + h* f(t,x,v)
    t = t+h

    xd_list.append(x)
    vd_list.append(v)
    td_list.append(t)

plt.plot(td_list, xd_list, color='r')
plt.xlabel("Time")
plt.ylabel("Position")
plt.rcParams["figure.figsize"] = (20, 10)
plt.title("Solution of LHO")
```

Out[10]:

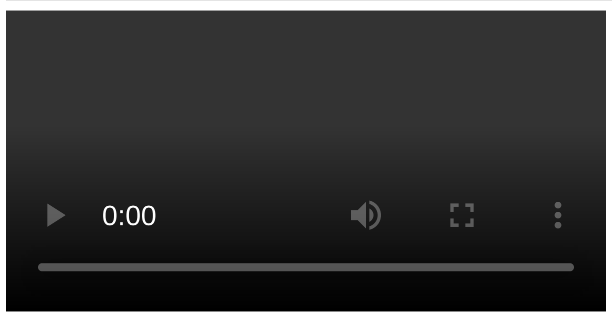
```
Text(0.5, 1.0, 'Solution of LHO')
```



In [1]:

```
from IPython.display import Video

Video("simulation.mp4")
```

Out[1]:



Syllabus

week 1: Python basics, classes, and objects.

week 2: Introduction to NumPy and SciPy.

week 3: Roots of algebraic equations: Bisection method, Secant method, Regula-falsi method, Newton-Raphson method.

week 4: Interpolation and numerical differentiations: Lagrange's interpolation, Newton's interpolation, Interpolation with SciPy, Differentiation through interpolation.

week 5: Numerical integrations: Rectangle rule, Trapezoidal rule, Simpson's 1/3 rule, Simpson's 3/8 rule.

week 6: Curve fitting: Least square method, NumPy polyfit module.

week 7: System of linear equations: Gauss elimination method, Gauss-Jordon method.

week 8: Ordinary Differential Equations: Euler's method, Runge-Kutta methods, Applications.