Lab Number 04
CoppeliaSim EDU – Position & Velocity Control of Different Joints

Introduction:

Joints and sensors play an important role in the field of robotics. CoppeliaSim EDU uses different types of joints such as revolute joint, prismatic joint and spherical joint. Sensors including proximity, force and vision also play a part in industry. This lab demonstrates the working of sensors and joints in CoppeliaSim EDU software.

Software Used:

CoppeliaSim EDU v4.7

Programming Language:

LUA

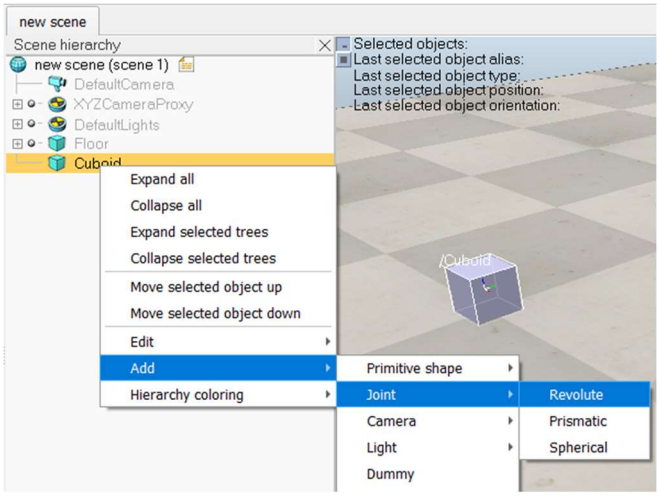
Objectives of the Lab:

- To learn how to program revolute and prismatic joints

VELOCITY CONTROL:

Add a Cuboid in the scene.

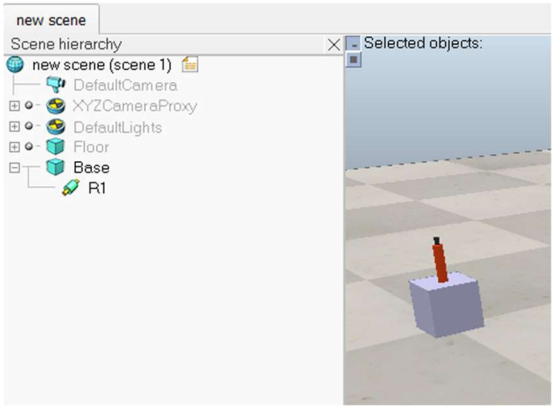
Right Click on cuboid and add revolute joint to make it a child object of cuboid.



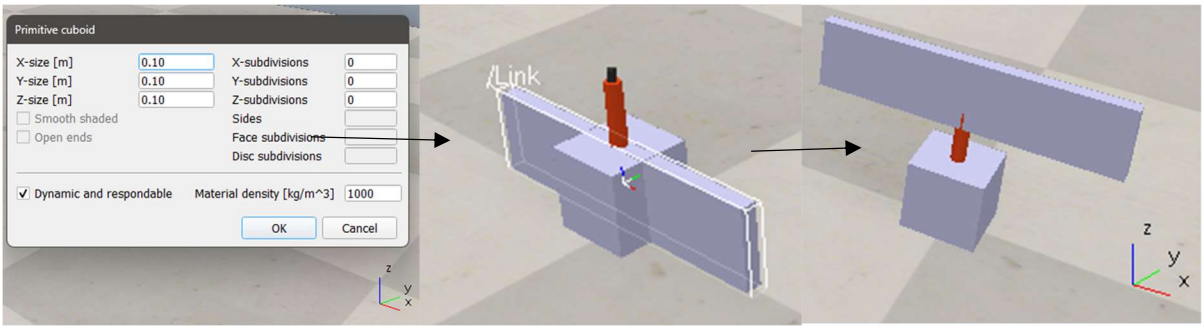
Make sure that revolute joint is a child object of cuboid

Change the name of cuboid as “Base” and revolute joint as “R1”.

Now translate the joint in upward direction. i.e, along 0.1 Z axis.

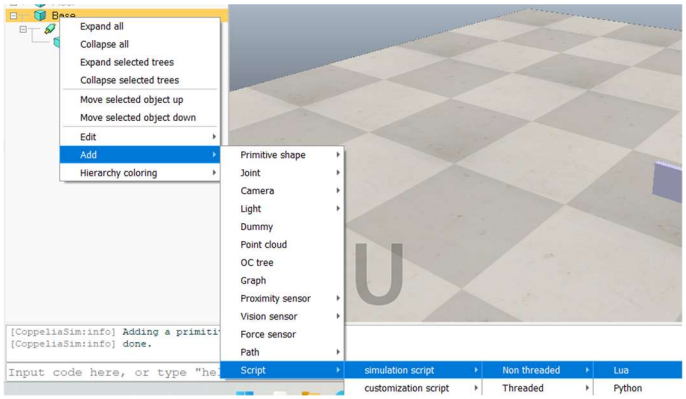


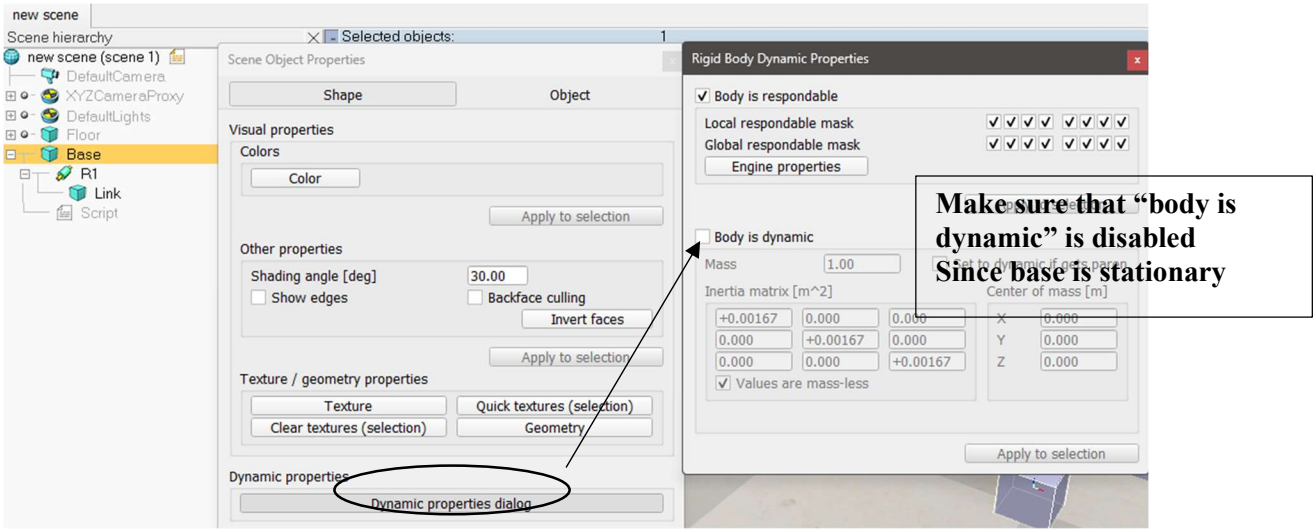
Add another cuboid. And make it a child object of revolute joint to make it connected. The dimensions are X-0.4, Y-0.02, Z-0.1 and name the cuboid as Link. Translate the link upward to connect it to the joint. Translate along Z-axis (world frame) – 0.2



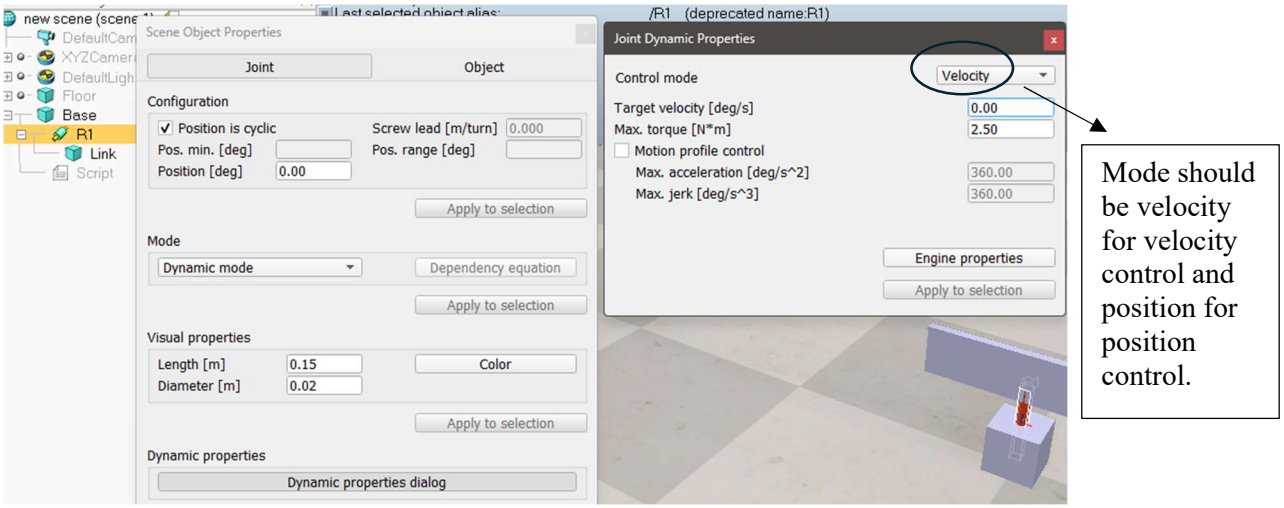
Now let's program!!!

Add a script in scene hierarchy as child script of "Base". Also change the property of base as static.





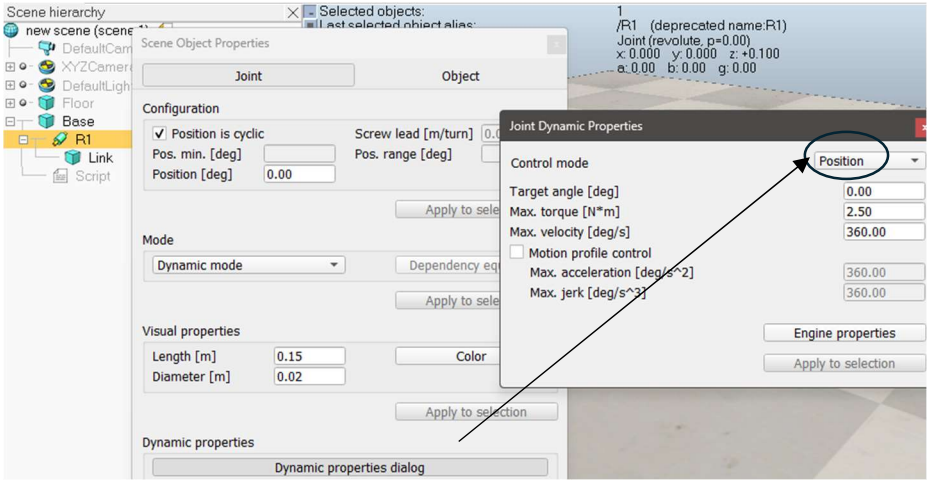
Now open the joint properties and go to Dynamic properties dialog.




```
function sysCall_init()
    sim = require('sim')
    A = sim.getObjectHandle('../R1')
    -- do some initialization here
end
function sysCall_actuation()
    -- put your actuation code here
    sim.setJointTargetVelocity(A,0.01)
end
```

This will rotate the joint with the velocity of 0.01m/s.

POSITION CONTROL:




```
function sysCall_init()
    sim = require('sim')
    A = sim.getObjectHandle('../R1')
    -- do some initialization here
end
function sysCall_actuation()
    -- put your actuation code here
    sim.setJointTargetPosition(A, math.pi/4)
end
```

This will rotate the joint by 45-degree angle.

TASK

Your task for this Lab is to implement velocity and speed control for prismatic joints.

Hint:

- Add a prismatic joint as a child object of Base.
- Add a cuboid that will be connected to prismatic joint.
- Don't forget to make Base static.
- In coding portion: Just change object handle from revolute or prismatic.