```
clc
addpath('C:\Users\YuN\Desktop\Capstone_Project_204\mr');

% the initial configuration
T_sc_initial = RpToTrans(eye(3),[1,0,0.025]');
T_se_initial = RpToTrans(eye(3),[0,0,0.5]');
T_sc_final = RpToTrans(rotz(-pi/2),[0,-1,0.025]');

%the standoff configuration of the end-effector above the cube
a = pi/6;
T_ce_standoff = [[-sin(a),0,-cos(a),0]',[0,1,0,0]',[cos(a),0,-sin(a),0]',[0,0,0.25,1]'];
%the configuration of the e-e relative to the cube while grasping
T_ce_grasp = [[-sin(a),0,-cos(a),0]',[0,1,0,0]',[cos(a),0,-sin(a),0]',[0,0,0,1]'];
% end-effector planned configuration(reference)
T_standoff_initial = T_sc_initial * T_ce_standoff;
T_grasp = T_sc_initial * T_ce_grasp;
T_standoff_final = T_sc_final * T_ce_standoff;
T_release = T_sc_final * T_ce_grasp;
%Construct a cell array for the path
T_configure = {T_se_initial,T_standoff_initial,T_grasp,T_grasp,T_standoff_initial,T_standoff_final,T_release,T_release,T_standoff_final};
% Generating reference trajectory
dt = 0.01;% 0.01 second
Tf = calculateTf(20);%total time = 20 ;the weighted time for each piece
Traj = [];% N * 13 matrix, N is the number of reference frame
grasp_state = 0;
for i = 1:8
    if i == 3
        grasp_state = 1;
    elseif i == 7
        grasp_state = 0;
    end

    Trajectory = Mybot.TrajectoryGenerator(T_configure{i},T_configure{i+1},Tf(i),dt,grasp_state,'Cartesian',5);
    Traj = [Traj;Trajectory];
end
writematrix(Traj,'Traj_1.csv');
disp('Trajectory Generated');
```

```
Trajectory Generated
```

Published with MATLAB® R2020b