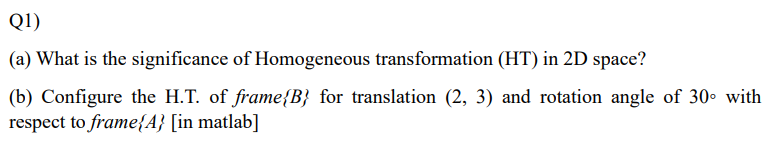
**Introductions to ai robotocs**

**Name: Girish S Roll No: AM.EN.U4AIE22044**

# **22AIE214 – Labsheet 3**



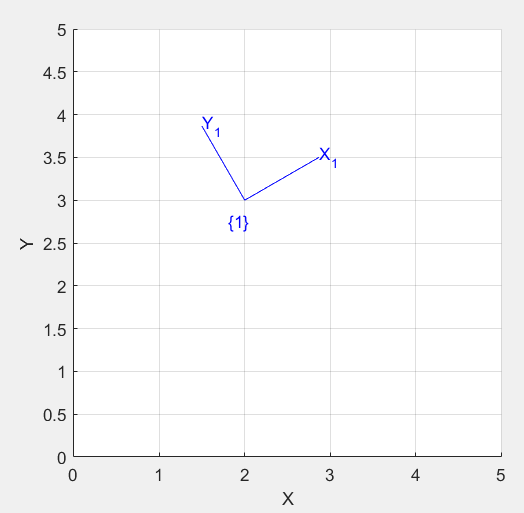
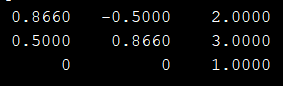
* simplify and unify the representation of various geometric transformations such as translation, rotation, and scaling.
* translation involves adding a translation vector to the coordinates of a point.
* Rotation and Scaling transformations, which are linear, can be naturally expressed as matrix multiplications in homogeneous coordinates.

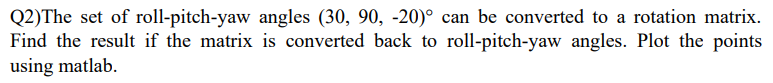
B = transl2(2, 3)\*trot2(30, 'deg');

disp(B);

plotvol([0 5 0 5]);

trplot2(B, 'frame', '1', 'color', 'b');



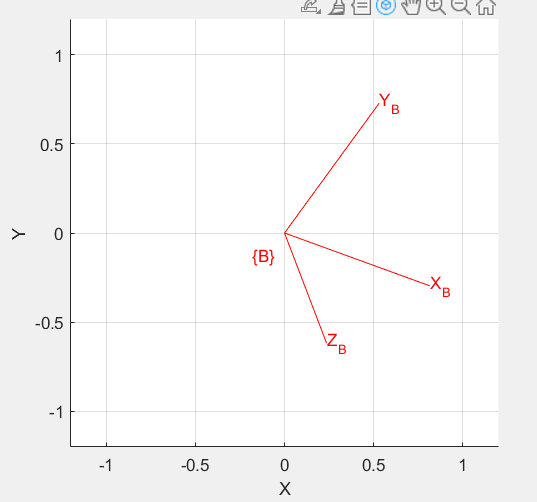
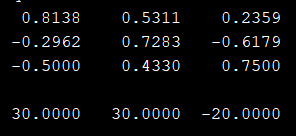


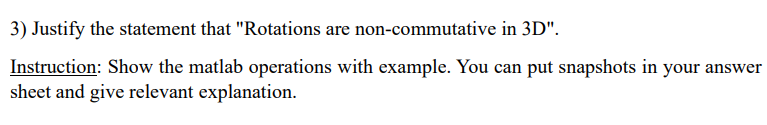
R=rpy2r(30,30,-20,'deg');

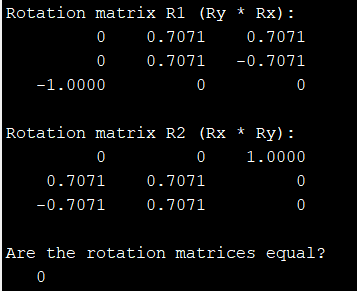
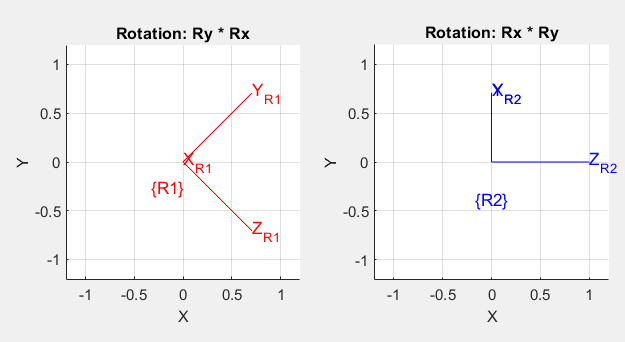
disp(R);

trplot(R, 'frame', 'B', 'color', 'r');

disp(tr2rpy(R,'deg'));







Rx = rotx(45, 'deg');

Ry = roty(90, 'deg');

R1 = Ry \* Rx;

R2 = Rx \* Ry;

disp('Rotation matrix R1 (Ry \* Rx):');

disp(R1);

disp('Rotation matrix R2 (Rx \* Ry):');

disp(R2);

isEqual = isequal(round(R1, 10), round(R2, 10));

disp('Are the rotation matrices equal?');

disp(isEqual);

figure;

subplot(1,2,1);

trplot(R1, 'frame', 'R1', 'color', 'r');

title('Rotation: Ry \* Rx');

subplot(1,2,2);

trplot(R2, 'frame', 'R2', 'color', 'b');

title('Rotation: Rx \* Ry');