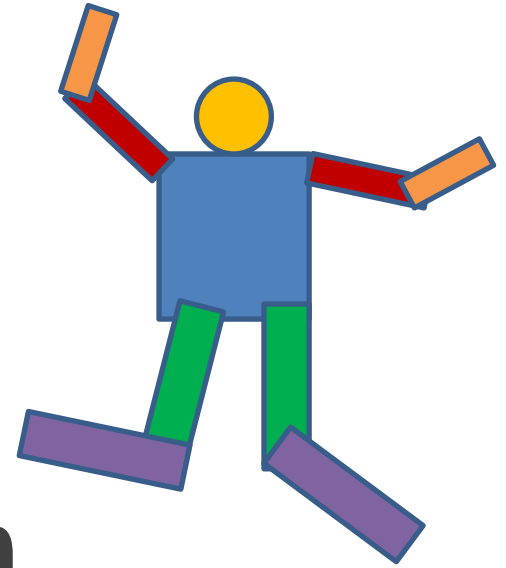


디지털 그래픽스 [7주차]

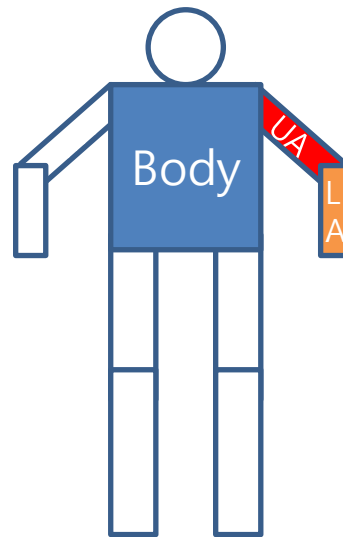
# Advanced Transformation

---

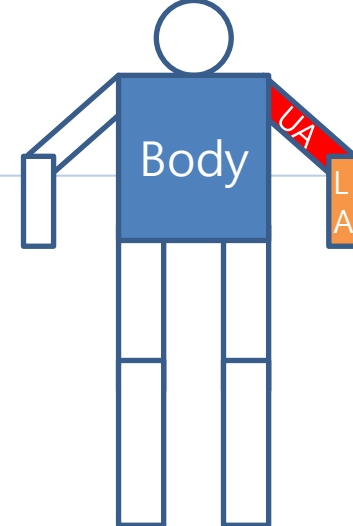


# 3D Model Hierarchy

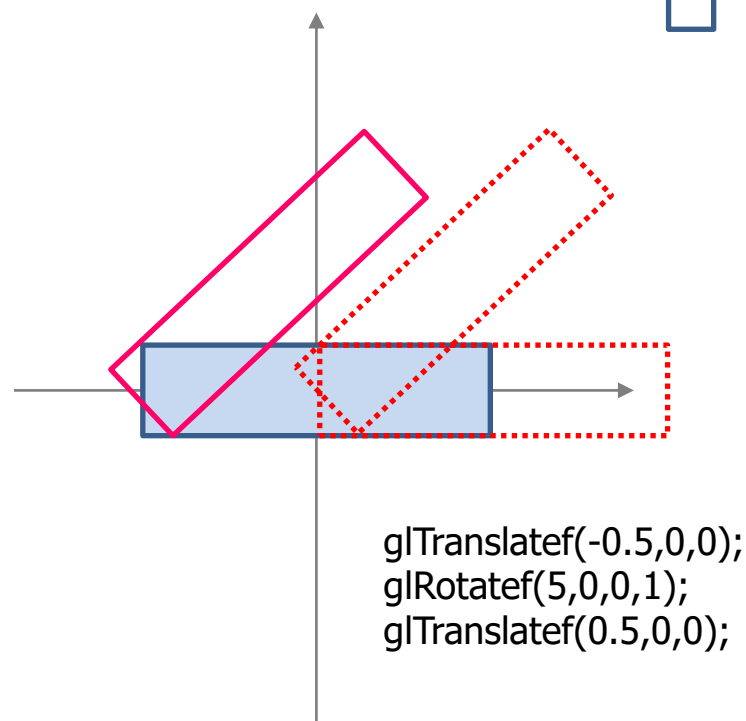
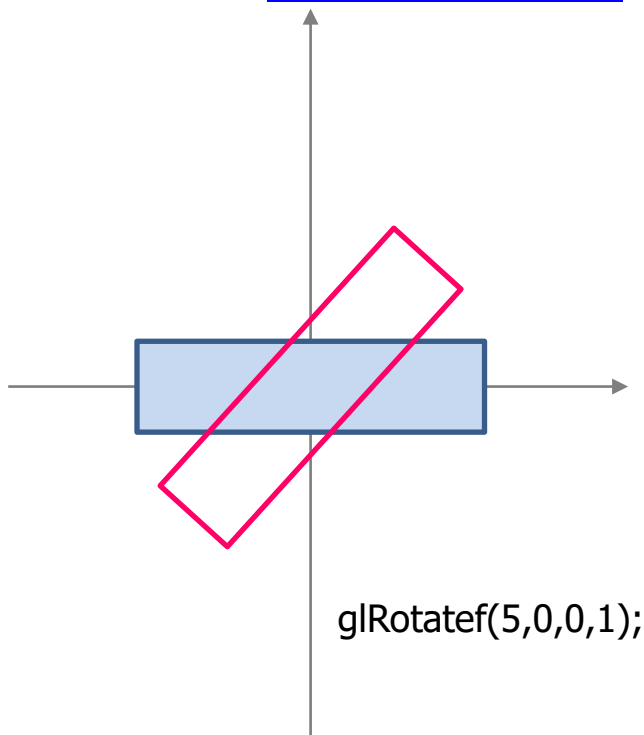
- Head
- Body
- *Arm*
  - *Left Arm*
    - Upper Arm
    - Lower Arm
  - *Right Arm*
    - Upper Arm
    - Lower Arm
- *Leg*
  - *Left Leg*
    - Upper Leg
    - Lower Leg
  - *Right Leg*
    - Upper Leg
    - Lower Leg



# 3D Model Joint Rotation

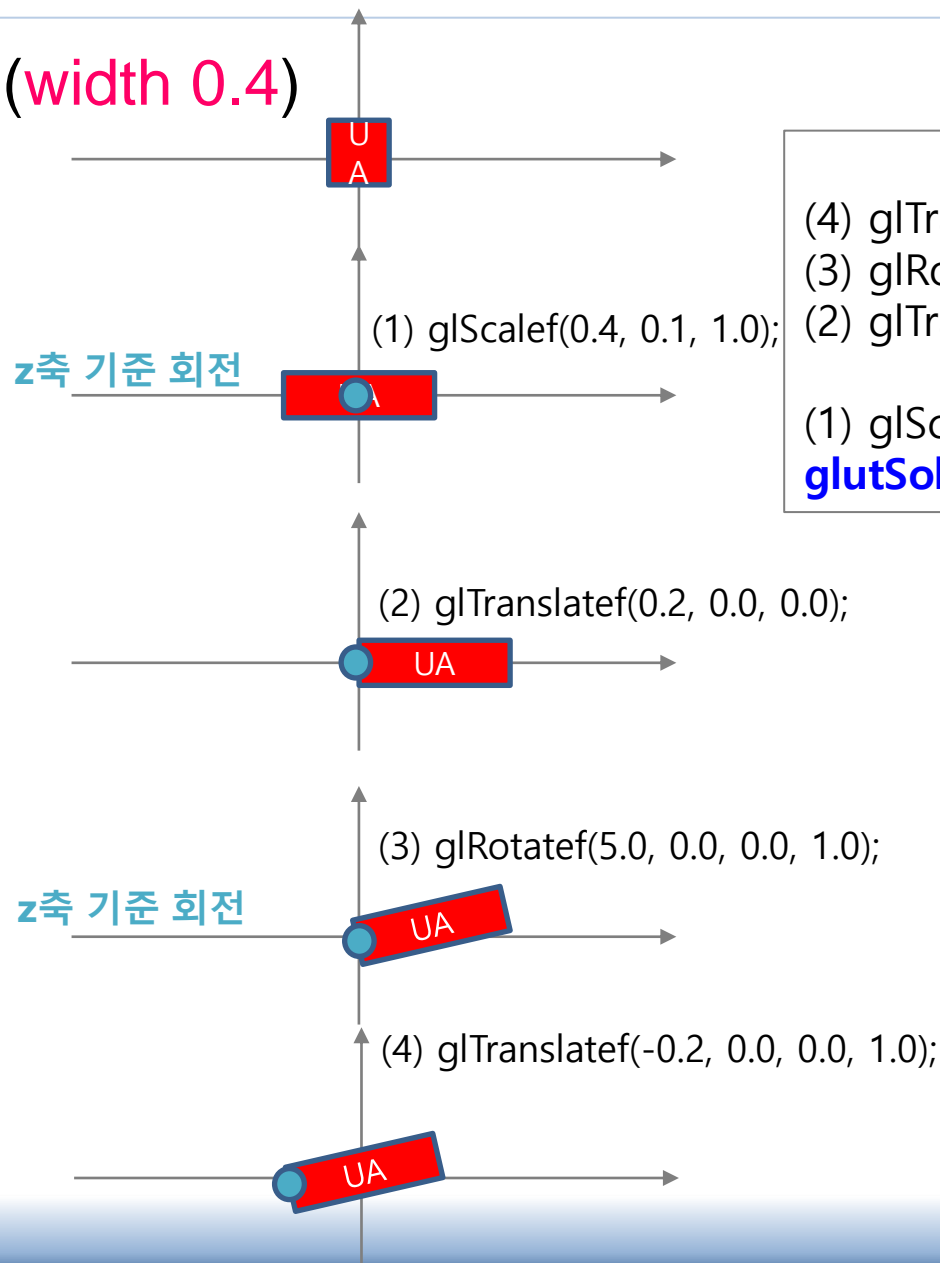
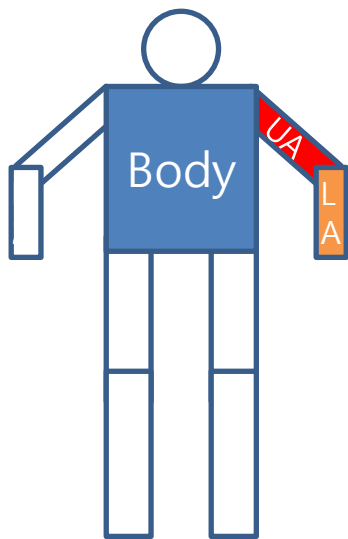


- `glutSolidCube(1)`
  - 원점에 길이 1인 cube를 그린다.
- 물체의 중심 기준으로 회전
- 물체의 왼쪽 끝 기준으로 회전



# 3D Model Joint Rotation

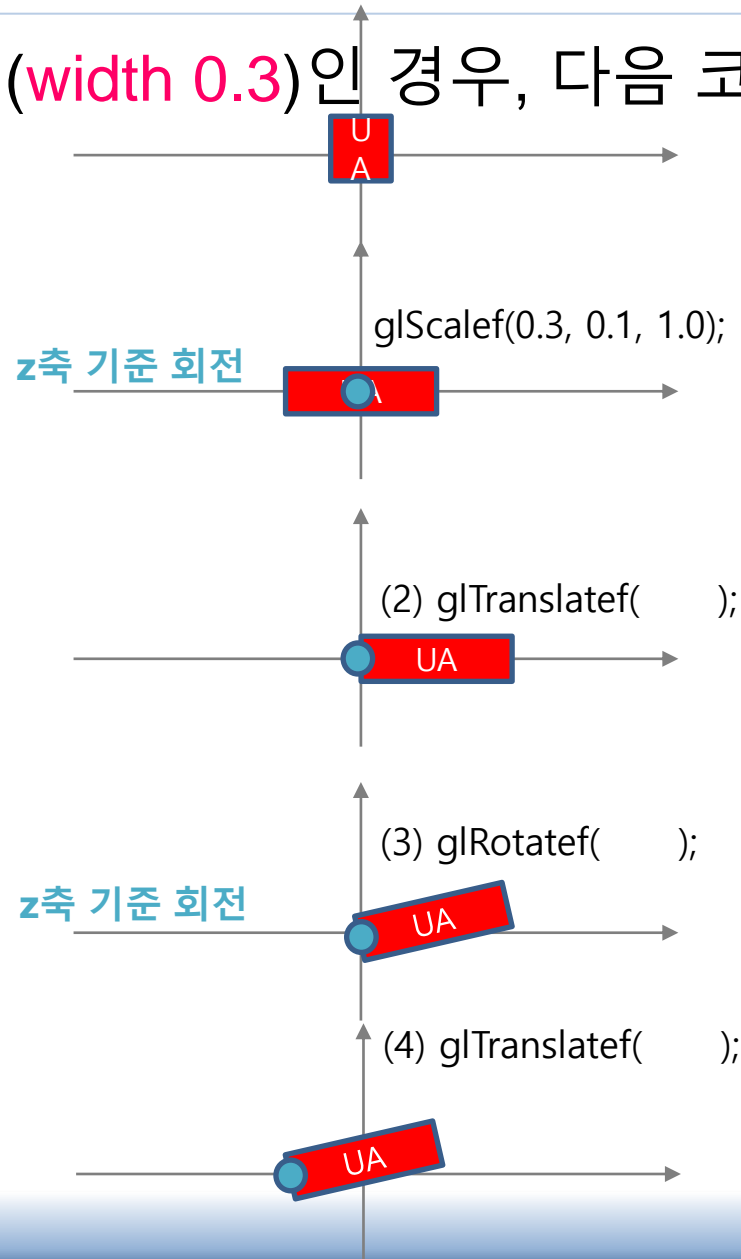
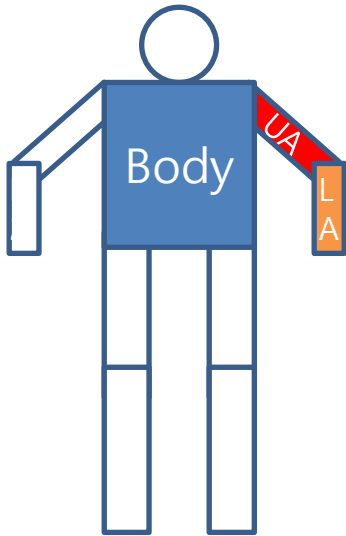
- Upper arm (width 0.4)



```
(4) glTranslatef(-0.2, 0.0, 0.0);  
(3) glRotatef(5.0, 0.0, 0.0, 1.0);  
(2) glTranslatef(0.2, 0.0, 0.0);  
  
(1) glScalef(0.4, 0.1, 1.0);  
glutSolidCube(1.0);
```

# 3D Model Joint Rotation

- Upper arm (**width 0.3**)인 경우, 다음 코드 수정

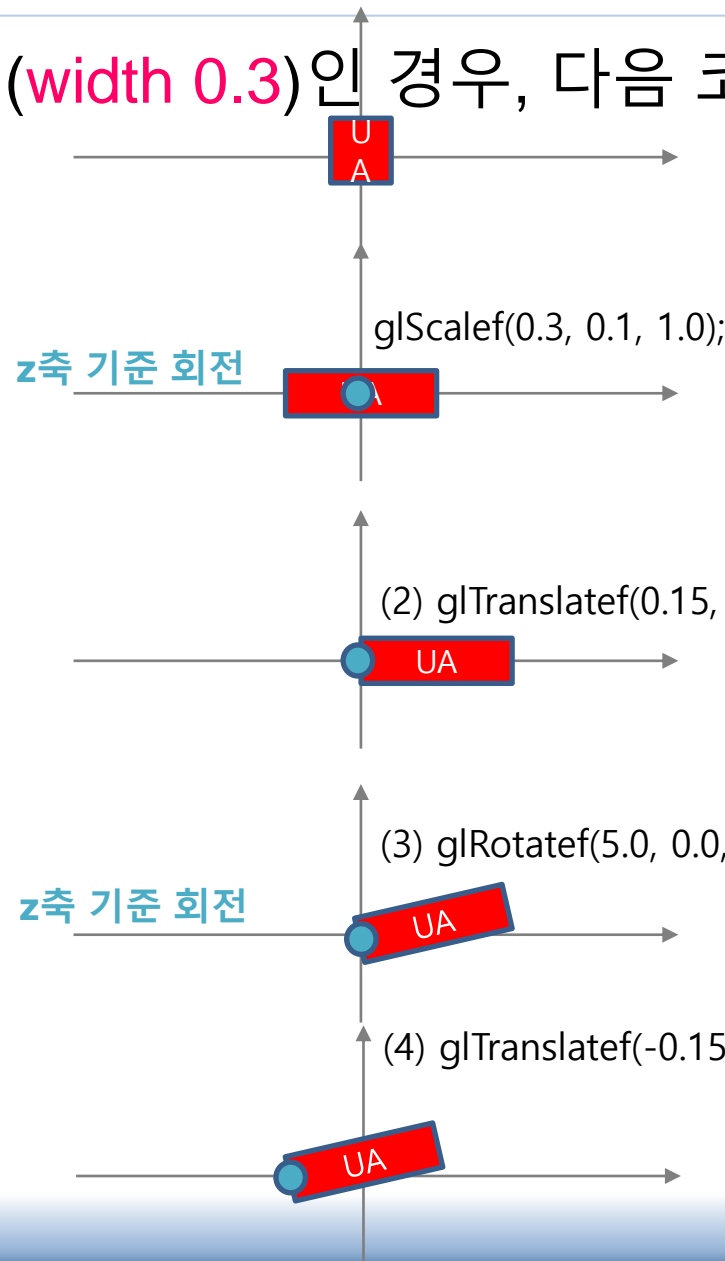
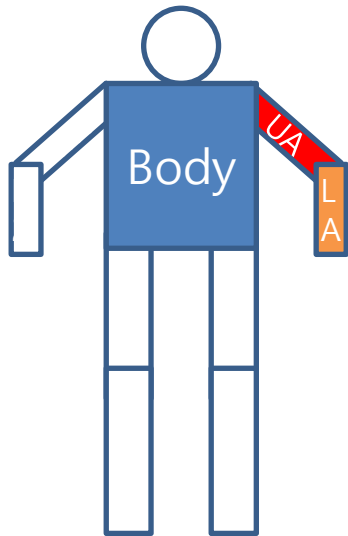


```
glTranslatef(-0.2, 0.0, 0.0);  
glRotatef(5.0, 0.0, 0.0, 1.0);  
glTranslatef(0.2, 0.0, 0.0);
```

```
glScalef(0.2, 0.1, 1.0);  
glutSolidCube(1.0);
```

# 3D Model Joint Rotation

- Upper arm (**width 0.3**)인 경우, 다음 코드 수정



`glScalef(0.3, 0.1, 1.0);`

(2) `glTranslatef(0.15, 0.1, 1.0);`

(3) `glRotatef(5.0, 0.0, 0.0, 1.0);`

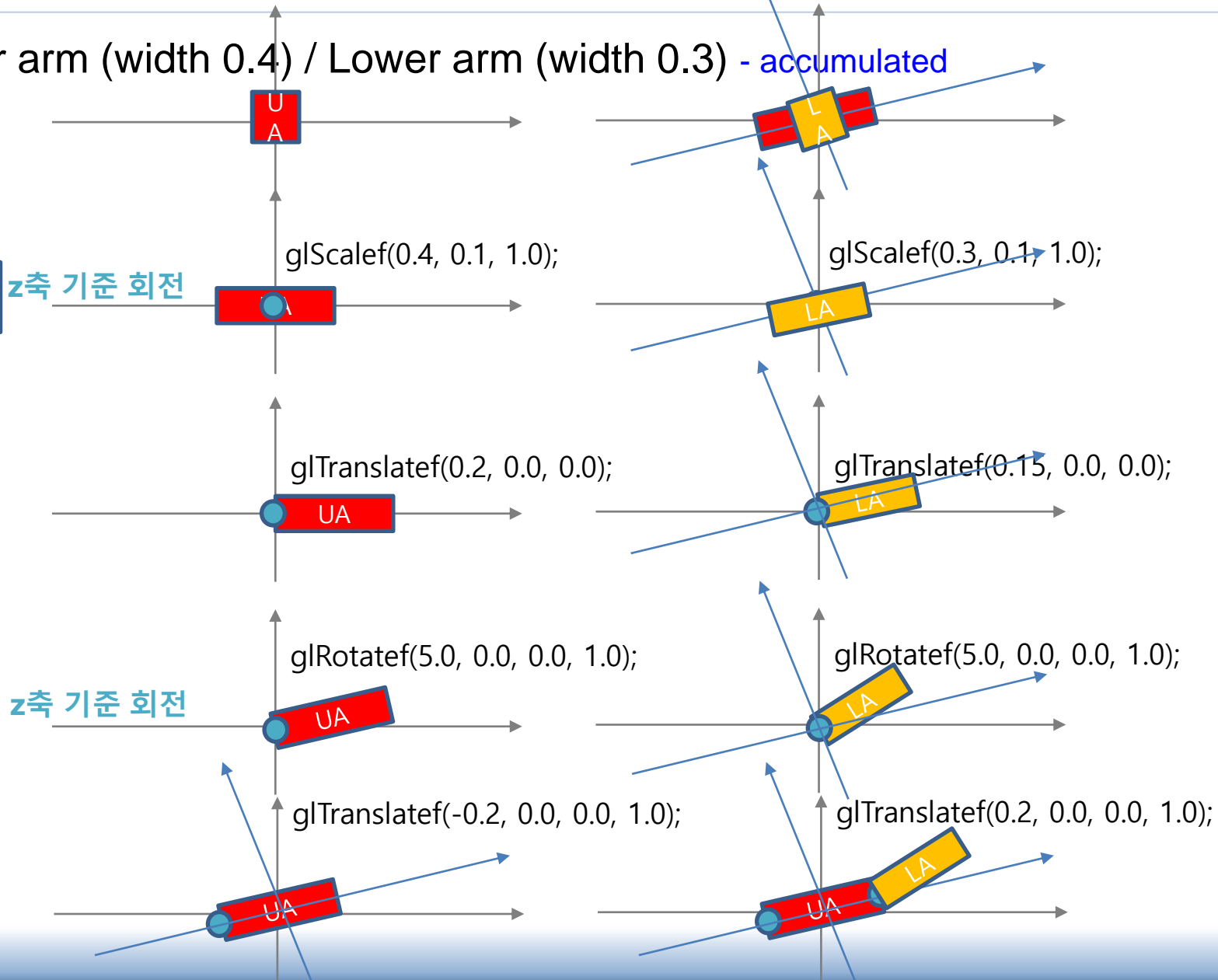
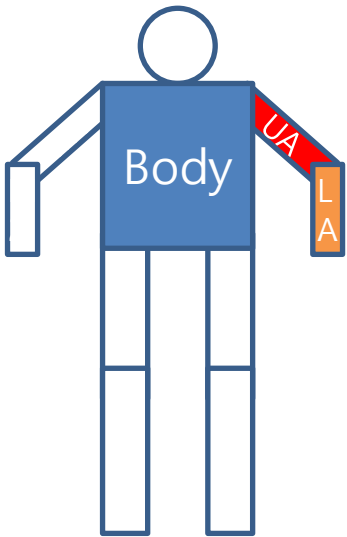
(4) `glTranslatef(-0.15, 0.0, 1.0);`

```
glTranslatef(-0.15, 0.0, 0.0);
glRotatef(5.0, 0.0, 0.0, 1.0);
glTranslatef(0.15, 0.0, 0.0);
```

```
glScalef(0.3, 0.1, 1.0);
glutSolidCube(1.0);
```

# Upper Arm & Lower Arm

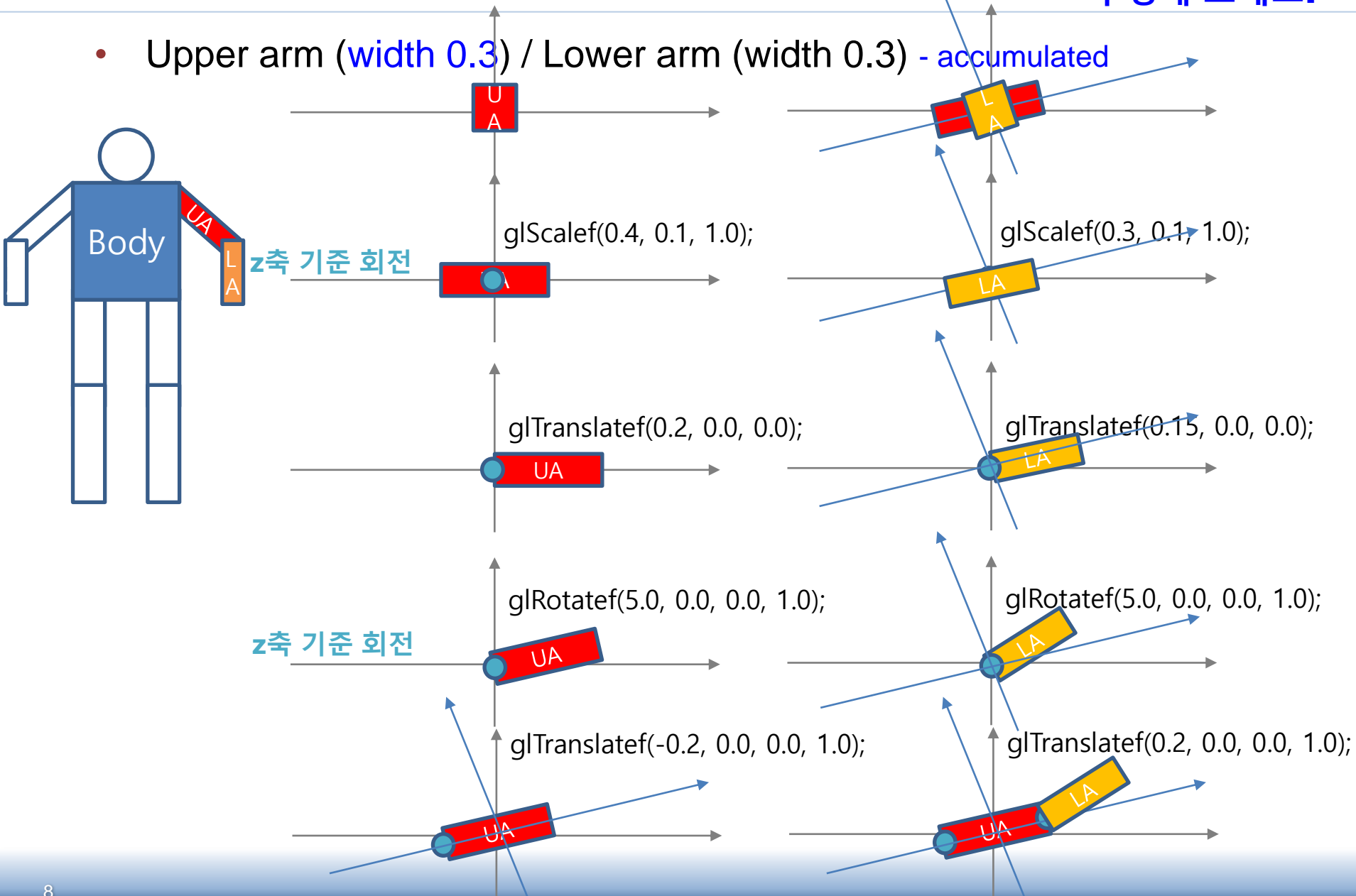
- Upper arm (width 0.4) / Lower arm (width 0.3) - accumulated



# Q1. Upper Arm & Lower Arm

수정해 보세요!

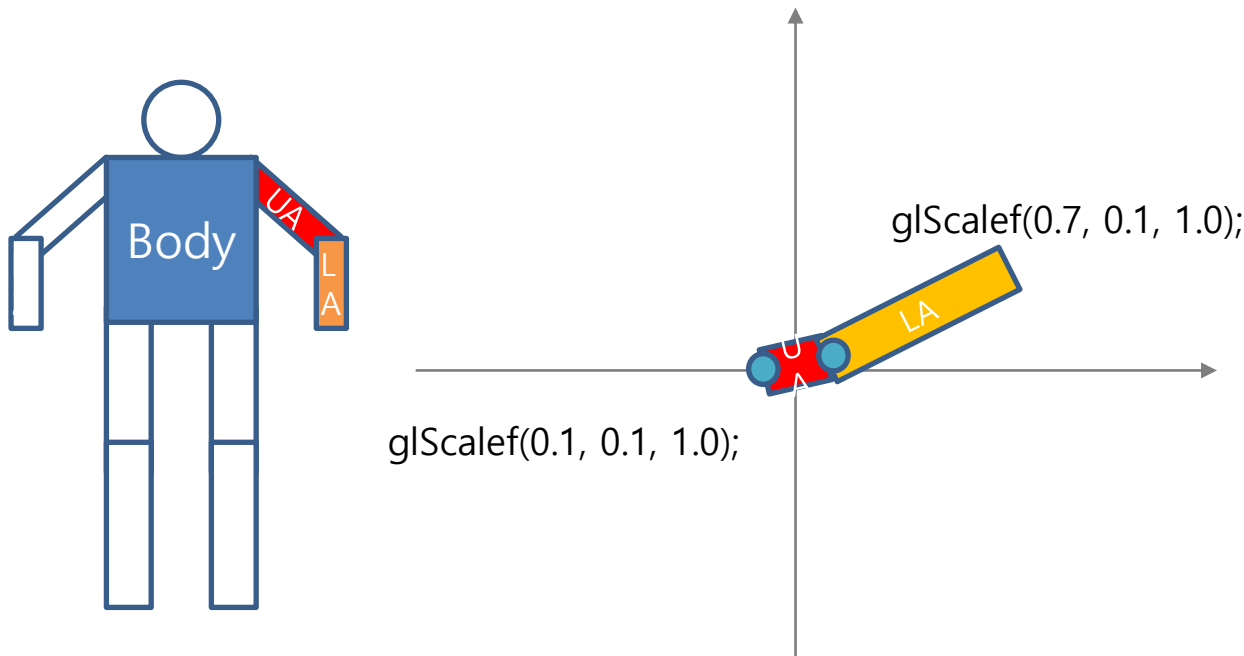
- Upper arm (width 0.3) / Lower arm (width 0.3) - accumulated





# Q. Upper Arm & Lower Arm

- Upper arm (width 0.1) / Lower arm (width 0.7)인 경우에 joint rotation이 제대로 동작하도록 아래 코드를 수정하세요.



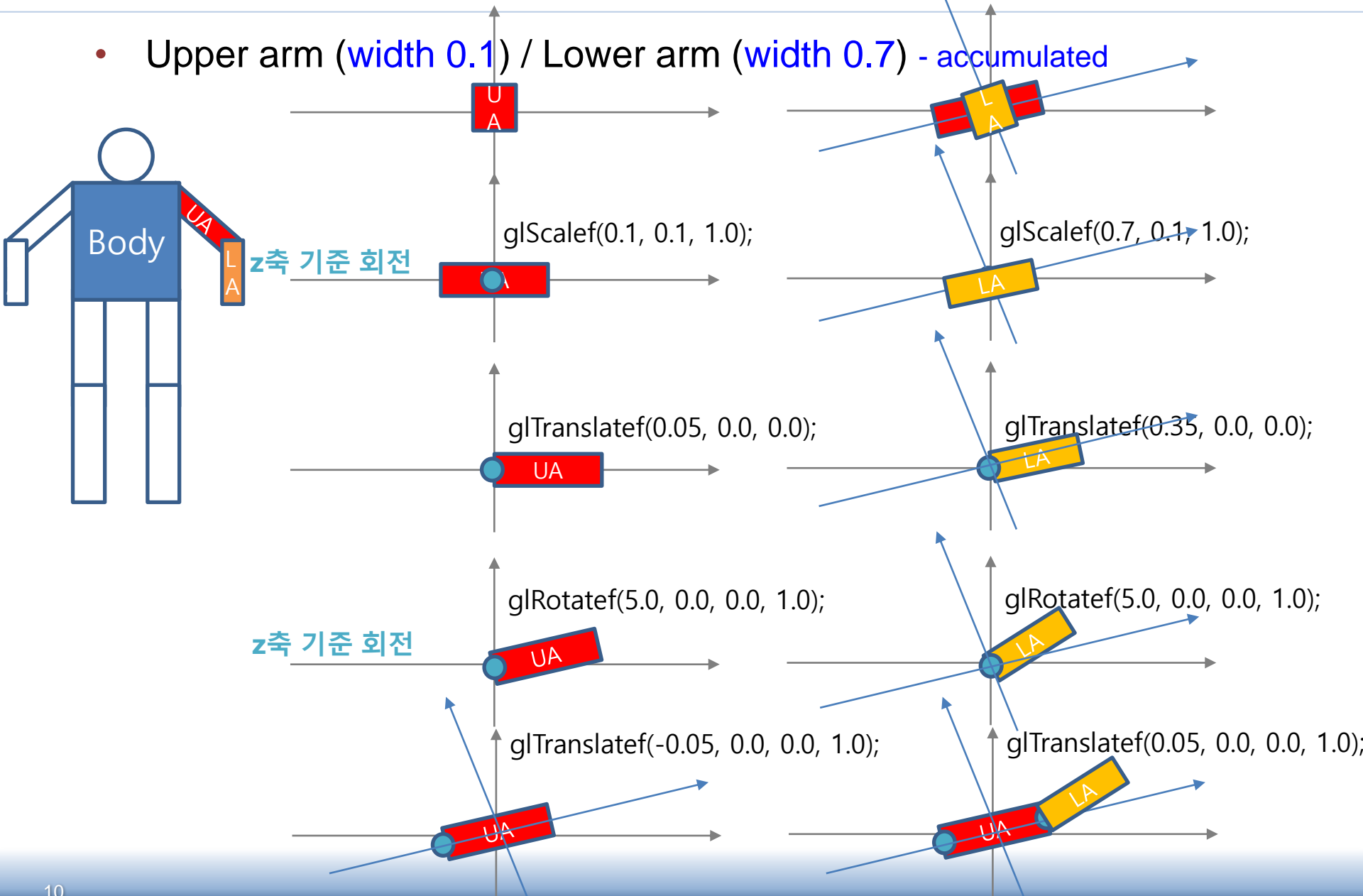
```
//accumulation
Upper Arm transformation

glTranslatef(0.2, 0.0, 0.0);
glRotatef(5.0, 0.0, 0.0, 1.0);
glTranslatef(0.15, 0.0, 0.0);

glScalef(0.3, 0.1, 1.0);
glutSolidCube(1.0);
```

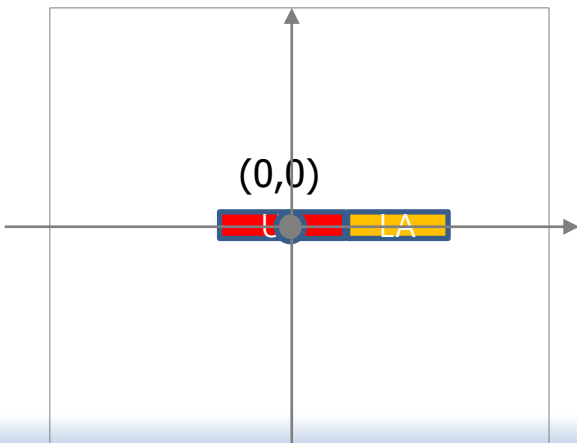
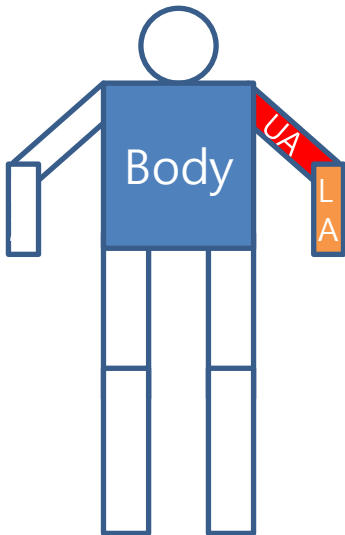
# A. Upper Arm & Lower Arm

- Upper arm (width 0.1) / Lower arm (width 0.7) - accumulated



# Code: 3D Model Joint Rotation

- Upper arm(0.4) & Lower arm(0.3) with push, pop matrix



**glPushMatrix();** ← display function의 일부

// upper arm

**glTranslatef(-0.2, 0.0, 0.0);**

**glRotatef((GLfloat)shoulder, 0.0, 0.0, 1.0);**

**glTranslatef(0.2, 0.0, 0.0);**

**glPushMatrix();**

**glScalef(0.4, 0.1, 1.0);**

**glColor3f(1.0, 0.0, 0.0);**

**glutSolidCube(1.0);**

**glColor3f(0.0, 0.0, 0.0);**

**glutWireCube(1.0);**

**glPopMatrix();**

// lower arm

**glTranslatef(0.2, 0.0, 0.0);**

**glRotatef((GLfloat)elbow, 0.0, 0.0, 1.0);**

**glTranslatef(0.15, 0.0, 0.0);**

**glPushMatrix();**

**glScalef(0.3, 0.1, 1.0);**

**glColor3f(1.0, 1.0, 0.0);**

**glutSolidCube(1.0);**

**glColor3f(0.0, 0.0, 0.0);**

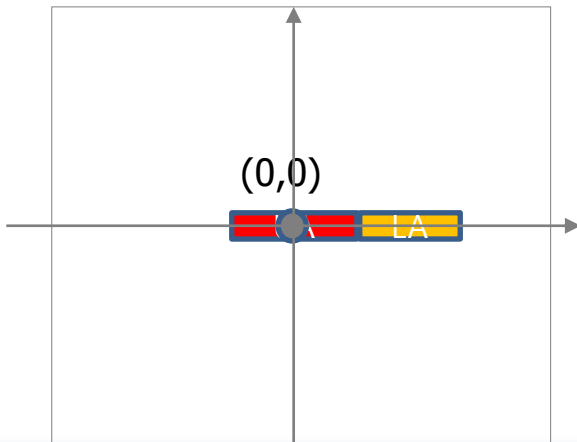
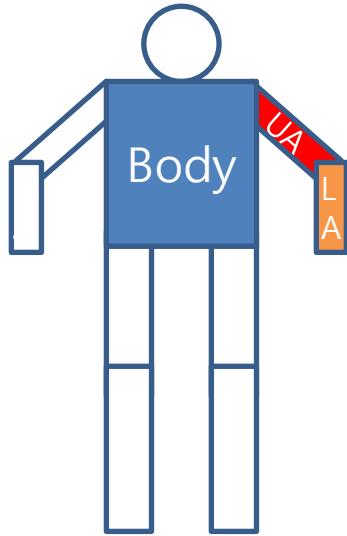
**glutWireCube(1.0);**

**glPopMatrix();**

**glPopMatrix();** ← 이 push/pop은 반드시 있어야 할까요?

## Q2. Upper Arm & Lower Arm

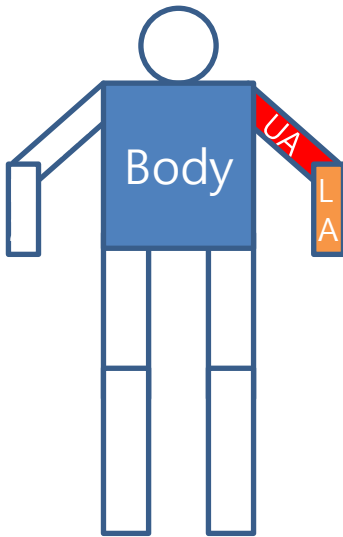
- Upper arm(width 0.6) & Lower arm(width 0.2)일 때, joint rotation이 제대로 작동하도록 아래 코드를 수정하세요.



```
glPushMatrix();  
    // upper arm  
    glTranslatef(-0.2, 0.0, 0.0);  
    glRotatef((GLfloat)shoulder, 0.0, 0.0, 1.0);  
    glTranslatef(0.2, 0.0, 0.0);  
    glPushMatrix();  
        glScalef(0.4, 0.1, 1.0);  
        glColor3f(1.0, 0.0, 0.0);  
        glutSolidCube(1.0);  
        glColor3f(0.0, 0.0, 0.0);  
        glutWireCube(1.0);  
    glPopMatrix();  
  
    // lower arm  
    glTranslatef(0.2, 0.0, 0.0);  
    glRotatef((GLfloat)elbow, 0.0, 0.0, 1.0);  
    glTranslatef(0.15, 0.0, 0.0);  
    glPushMatrix();  
        glScalef(0.3, 0.1, 1.0);  
        glColor3f(1.0, 1.0, 0.0);  
        glutSolidCube(1.0);  
        glColor3f(0.0, 0.0, 0.0);  
        glutWireCube(1.0);  
    glPopMatrix();  
glPopMatrix();
```

# Body

- Body(0.5) & Upper arm(0.4) & Lower arm(0.3)

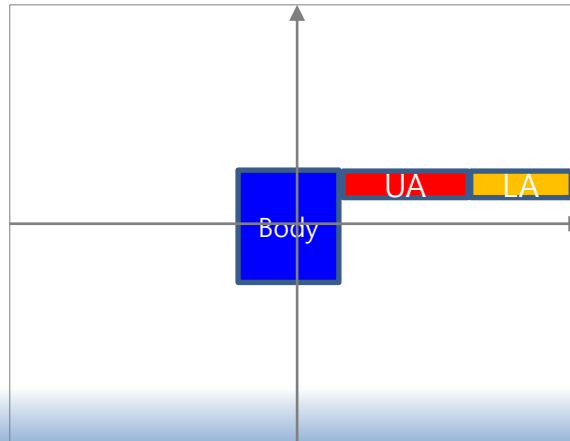
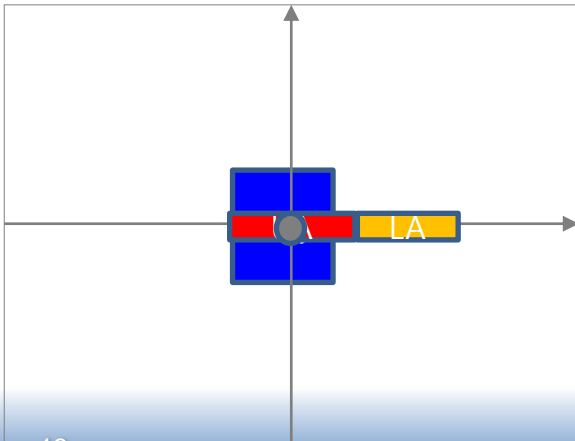


```
glPushMatrix();
// Body
glPushMatrix();
    glScalef(0.5, 0.5, 1.0);
    glColor3f(0.0, 0.0, 1.0);
    glutSolidCube(1.0);
    glColor3f(0.0, 0.0, 0.0);
    glutWireCube(1.0);
glPopMatrix();
glPopMatrix();
```

```
glPushMatrix();
// Upper Arm
glTranslatef(0.5, 0.25, 0.0);
glTranslatef(-0.2, 0.0, 0.0);
glRotatef(0.5, 0.0, 0.0, 1.0);
glTranslatef(0.2, 0.0, 0.0);
glPushMatrix();
    glScalef(0.4, 0.1, 1.0);
    glColor3f(1.0, 0.0, 0.0);
    glutSolidCube(1.0);
    glColor3f(0.0, 0.0, 0.0);
    glutWireCube(1.0);
glPopMatrix();

// Lower Arm
glTranslatef(0.2, 0.0, 0.0);
glRotatef(0.5, 0.0, 0.0, 1.0);
glTranslatef(0.15, 0.0, 0.0);
glPushMatrix();
    glScalef(0.3, 0.1, 1.0);
    glColor3f(1.0, 1.0, 0.0);
    glutSolidCube(1.0);
    glColor3f(0.0, 0.0, 0.0);
    glutWireCube(1.0);
glPopMatrix();

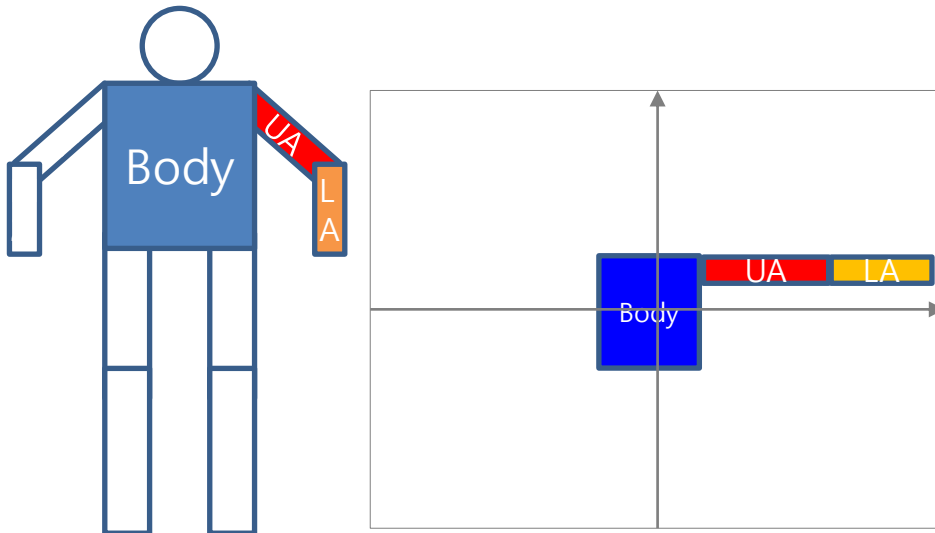
glPopMatrix();
glPopMatrix();
```



# code: Body

## display() 함수 revised

- Body(0.5)
- Upper arm(0.4)
- Lower arm(0.3)



```
void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glPushMatrix();
    // Body
    glPushMatrix();
        glScalef(0.5, 0.5, 1.0);
        glColor3f(0.0, 0.0, 1.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

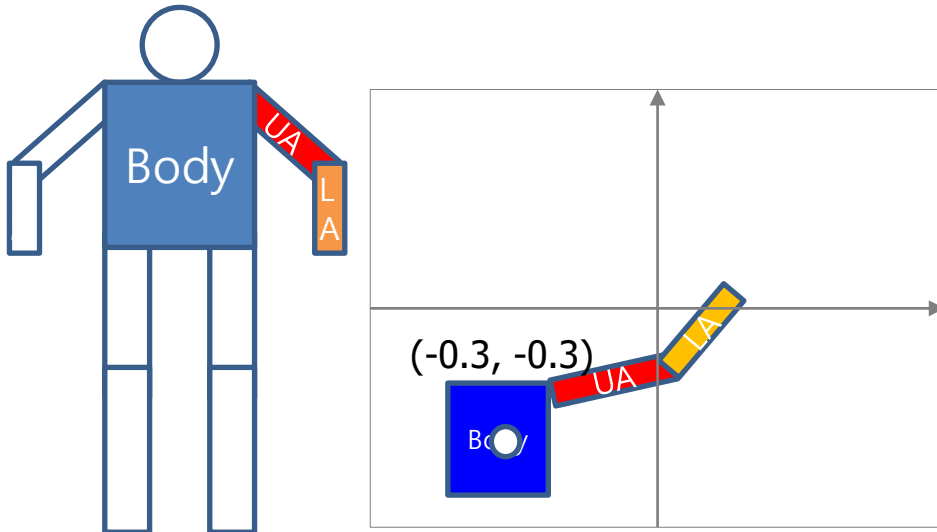
    glPushMatrix();
    // Upper Arm
    glTranslatef(0.5, 0.2, 0.0);
    glTranslatef(-0.2, 0.0, 0.0);
    glRotatef((GLfloat)shoulder, 0.0, 0.0, 1.0);
    glTranslatef(0.2, 0.0, 0.0);
    glPushMatrix();
        glScalef(0.4, 0.1, 1.0);
        glColor3f(1.0, 0.0, 0.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

    // Lower Arm
    glTranslatef(0.2, 0.0, 0.0);
    glRotatef((GLfloat)elbow, 0.0, 0.0, 1.0);
    glTranslatef(0.15, 0.0, 0.0);
    glPushMatrix();
        glScalef(0.3, 0.1, 1.0);
        glColor3f(1.0, 1.0, 0.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

    glPopMatrix();
    glFlush();
}
```

# Q. Body

- Body를  $(-0.3, -0.3)$  이동시킨 후에도 upper arm과 lower arm의 joint rotation이 제대로 작동하도록 다음 코드를 수정하세요.
  - Body(0.5)
  - Upper arm(0.4)
  - Lower arm(0.3)



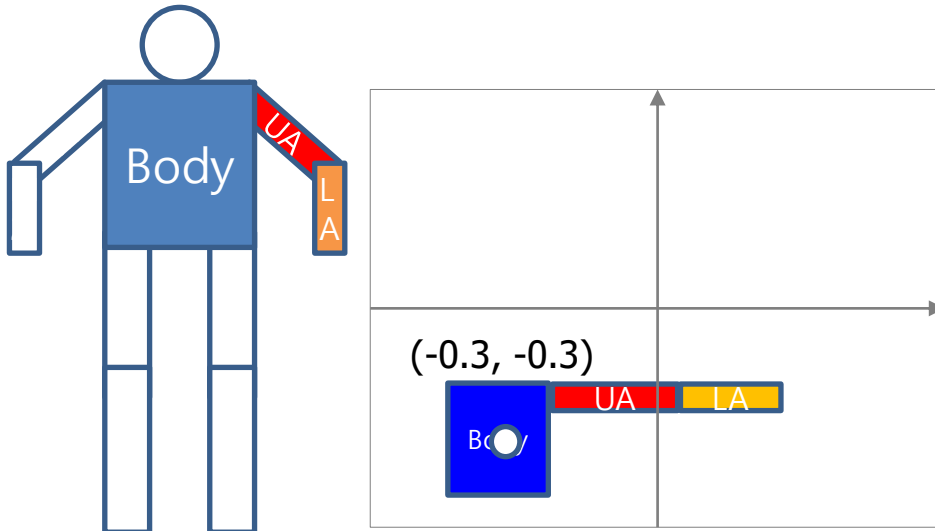
```
void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glPushMatrix();
    // Body
    glPushMatrix();
        glScale(0.5, 0.5, 1.0);
        glColor3f(0.0, 0.0, 1.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

    glPushMatrix();
    // Upper Arm
    glTranslatef(0.25, 0.2, 0.0);
    glRotatef((GLfloat)shoulder, 0.0, 0.0, 1.0);
    glTranslatef(0.2, 0.0, 0.0);
    glPushMatrix();
        glScale(0.4, 0.1, 1.0);
        glColor3f(1.0, 0.0, 0.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

    // Lower Arm
    glTranslatef(0.2, 0.0, 0.0);
    glRotatef((GLfloat)elbow, 0.0, 0.0, 1.0);
    glTranslatef(0.15, 0.0, 0.0);
    glPushMatrix();
        glScale(0.3, 0.1, 1.0);
        glColor3f(1.0, 1.0, 0.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();
    glPopMatrix();
    glFlush();
}
```

# A. Body

- Body를 (-0.3, -0.3) 이동시킨 후에도 upper arm과 lower arm의 joint rotation이 제대로 작동하도록 다음 코드를 수정하세요.
  - Body(0.5)
  - Upper arm(0.4)
  - Lower arm(0.3)



```
void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glPushMatrix();
    // Body
    glTranslatef(-0.3, -0.3, 0);
    glPushMatrix();
        glScale(0.5, 0.5, 1.0);
        glColor3f(0.0, 0.0, 1.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

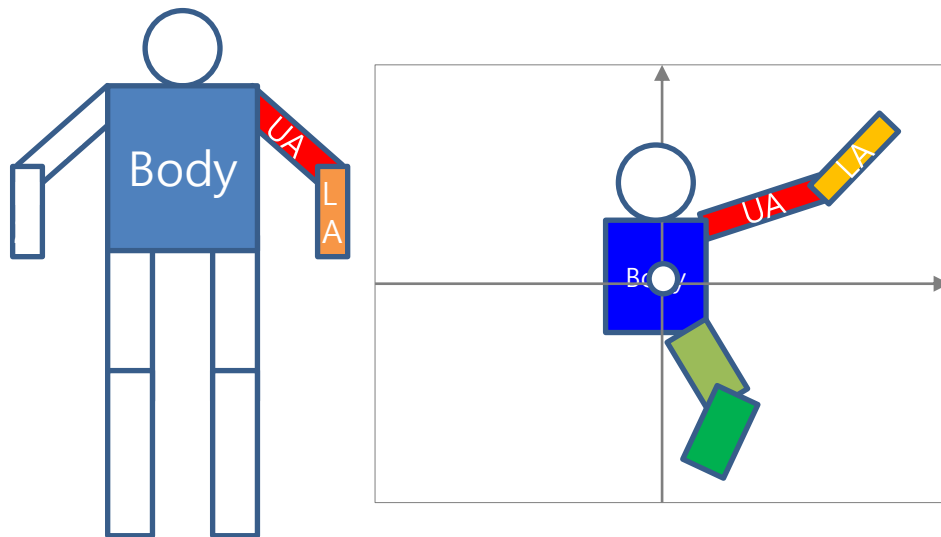
    glPushMatrix();
    // Upper Arm
    glTranslatef(0.25, 0.2, 0.0);
    glRotatef((GLfloat)shoulder, 0.0, 0.0, 1.0);
    glTranslatef(0.2, 0.0, 0.0);
    glPushMatrix();
        glScale(0.4, 0.1, 1.0);
        glColor3f(1.0, 0.0, 0.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

    // Lower Arm
    glTranslatef(0.2, 0.0, 0.0);
    glRotatef((GLfloat)elbow, 0.0, 0.0, 1.0);
    glTranslatef(0.15, 0.0, 0.0);
    glPushMatrix();
        glScale(0.3, 0.1, 1.0);
        glColor3f(1.0, 1.0, 0.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();
    glPopMatrix();
    glFlush();
}
```



## Q3. 3D Model Hierarchy

- 왼쪽 코드에 Head, Upper Leg(0.4), Lower Leg(0.6)을 추가하는 코드를 작성하세요.
  - Body(0.5)
  - Upper arm(0.4)
  - Lower arm(0.3)
  - Upper leg(0.3)
  - Lower leg(0.4)



```
void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glPushMatrix();
    // Body
    glTranslatef(-0.3, -0.3, 0);
    glPushMatrix();
        glScale(0.5, 0.5, 1.0);
        glColor3f(0.0, 0.0, 1.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

    glPushMatrix();
    // Upper Arm
    glTranslatef(0.25, 0.2, 0.0);
    glRotatef((GLfloat)shoulder, 0.0, 0.0, 1.0);
    glTranslatef(0.2, 0.0, 0.0);
    glPushMatrix();
        glScale(0.4, 0.1, 1.0);
        glColor3f(1.0, 0.0, 0.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

    // Lower Arm
    glTranslatef(0.2, 0.0, 0.0);
    glRotatef((GLfloat)elbow, 0.0, 0.0, 1.0);
    glTranslatef(0.15, 0.0, 0.0);
    glPushMatrix();
        glScale(0.3, 0.1, 1.0);
        glColor3f(1.0, 1.0, 0.0);
        glutSolidCube(1.0);
        glColor3f(0.0, 0.0, 0.0);
        glutWireCube(1.0);
    glPopMatrix();

    glPopMatrix();
    glPopMatrix();
    glFlush();
}
```

# Keyboard 기능

- *glutKeyboardFunc()*

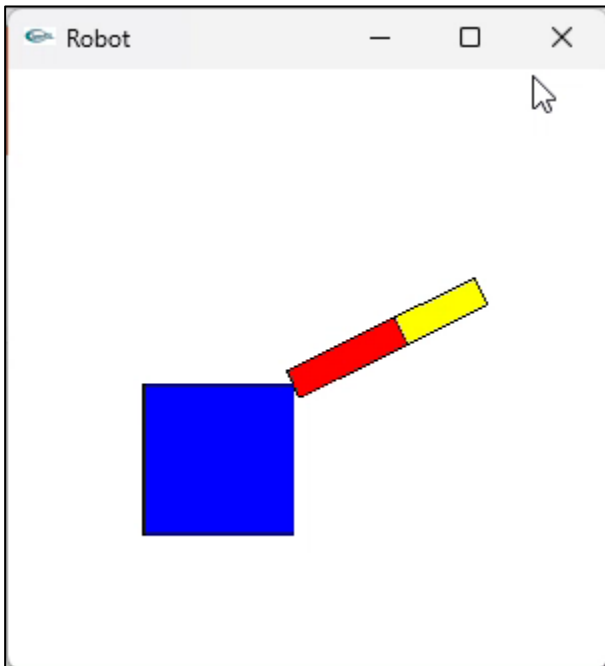
- If 's' is pressed, rotate shoulder
- If 'S' is pressed, rotate shoulder reversely
- If 'e' is pressed, rotate elbow
- If 'E' is pressed, rotate elbow reverse

```
float shoulder = 0, elbow = 0;

void keyboard(unsigned char key, int x, int y)
{
    switch (key)
    {
        case 's': shoulder += 5;
                  glutPostRedisplay();
                  break;
        case 'S': shoulder -= 5;
                  glutPostRedisplay();
                  break;
        case 'e': elbow += 5;
                  glutPostRedisplay();
                  break;
        case 'E': elbow -= 5;
                  glutPostRedisplay();
                  break;
        default: break;
    }
}
```

# Animation 기능

- glutIdleFunc()*



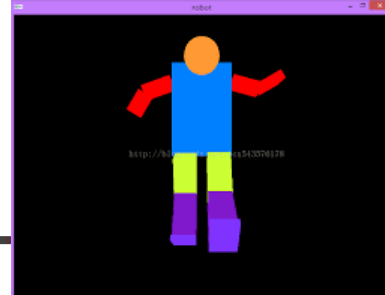
```
float shoulder = 0;
float elbow = 0;
int anim = 0; // animation mode
string direction = "clockwise"; // motion direction

void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    if (anim) {
        if (direction == "clockwise") shoulder += 0.05;
        else shoulder -= 0.05;
        if (shoulder >= 90) direction = "anticlockwise";
        else if (shoulder <= 0) direction = "clockwise";
    }
    ....
}

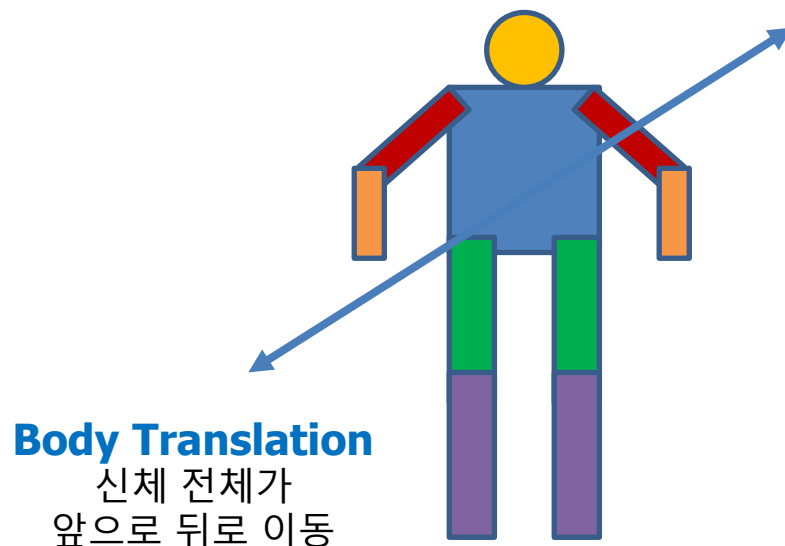
int main(int argc, char** argv)
{
    .....
    glutIdleFunc(display);
    .....
    glutMainLoop();

    return 0;
}
```

# 과제. Rainbow Man



- Rainbow Man 신체의 **모든** 부분을 키보드로 움직여 **재미있는 자세**를 만들어 보세요. 아래의 각 키보드로 각 조인트를 회전하는 기능을 구현합니다. (키보드 추가 가능)
- Animation 기능으로 Rainbow man이 걷는 기능을 구현합니다.
  - S, s : Shoulder Rotation
  - E, e : Elbow Rotation
  - W, w : Wrist Rotation
  - N, n : Neck Rotation
  - T, t : Body Translation
  - L, l : Leg Rotation



---

Hierarchical Transformation  
Thank you !