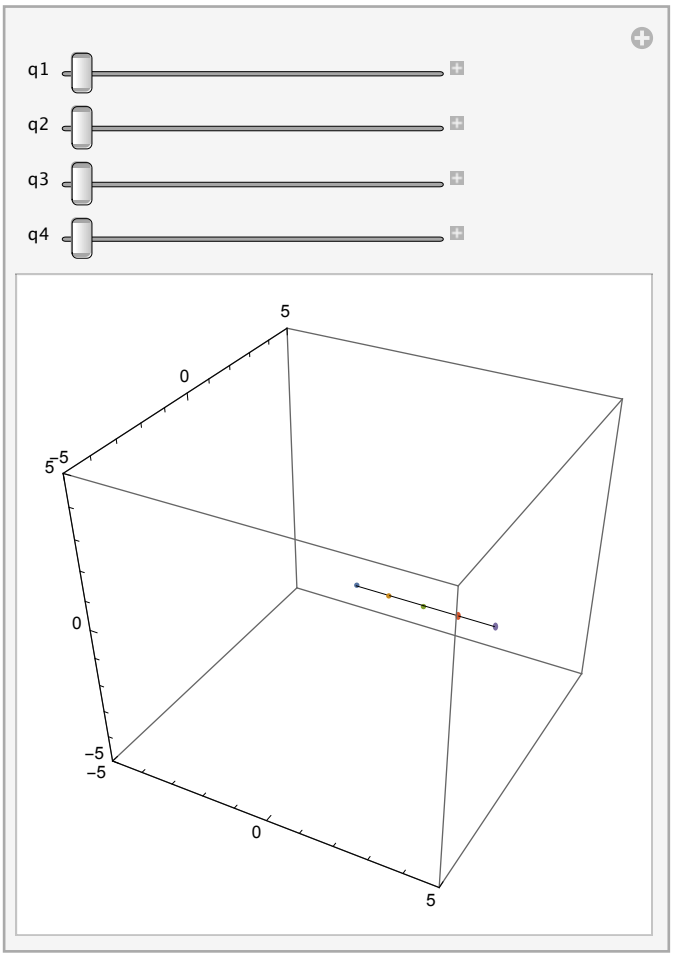
Part 1. Convert our old 2D model to 3D

In[9]:=
```
(*DO THIS: - Start with your 2d limb model from an earlier tutorial*)
(*DO THIS: - RENAME JOINTS from 0 to 4*)
(*DO THIS: - add z coordinates for each joint*)
(*DO THIS: - modify plot statements and define XYZ range*)
range = 5;(*the maximum +/- range of the plot below*)
XYZrange = {{-range, range}, {-range, range}, {-range, range}}
(*define segment lengths - we can change them later*)
l1 = 1; l2 = 1; l3 = 1; l4 = 1;
(*Writing out the equations for each x, y coordinate for each joint...*)
Manipulate[
 x0 = 0;
 y0 = 0;
 z0 = 0;
 x1 = x0 + l1 * Cos[q1];
 y1 = y0 + l1 * Sin[q1];
 z1 = z0 + 0;
 x2 = x1 + l2 * Cos[q2];
 y2 = y1 + l2 * Sin[q2];
 z2 = z1 + 0;
 x3 = x2 + l3 * Cos[q3];
 y3 = y2 + l3 * Sin[q3];
 z3 = z2 + 0;
 x4 = x3 + l4 * Cos[q4];
 y4 = y3 + l4 * Sin[q4];
 z4 = z3 + 0;

 j0 = {x0, y0, z0};
 j1 = {x1, y1, z1};
 j2 = {x2, y2, z2};
 j3 = {x3, y3, z3};
 j4 = {x4, y4, z4};
 l1 = 1;
 l2 = 1;
 l3 = 1;
 l4 = 1;
 (*define segment lengths - we can change them later*)
 JOINTS = {{j0}, {j1}, {j2}, {j3}, {j4}};
 (*compile a list of xy points for each joint location -
  put each joint within curly brackets which encloses it in a list.  Since
   each joint becomes a 'list-of-lists', Mathematica will then allow
  you to treat each joint as a separate data set which can be styled
  independently i.e. they can each have their own plot marker*)
 SEGMENTS = Flatten[JOINTS, 1];
 (*to plot the segment lines between joints,
 we'd rather not have lists-of-lists for each point like the above.  So,
 we use Flatten[JOINTS,1] to remove 1 level of list bracketing. If you did ...
  used 2, it would flatten the entire JOINTS structure into a 1d list.*)
 jointPlot = ListPointPlot3D[JOINTS, PlotRange → XYZrange, AspectRatio → 1];
 (*Plot the joints*)
 segmentPlot = Graphics3D[Line[SEGMENTS]];
 Show[jointPlot, segmentPlot, BoxRatios -> {1, 1, 1}, ImageSize → 300]
 (*superimpose the plots*)
 , {q1, 0, Pi}, {q2, 0, Pi}, {q3, 0, Pi}, {q4, 0, Pi}]
```

Out[10]=
```
{{-5, 5}, {-5, 5}, {-5, 5}}
```

Out[12]=



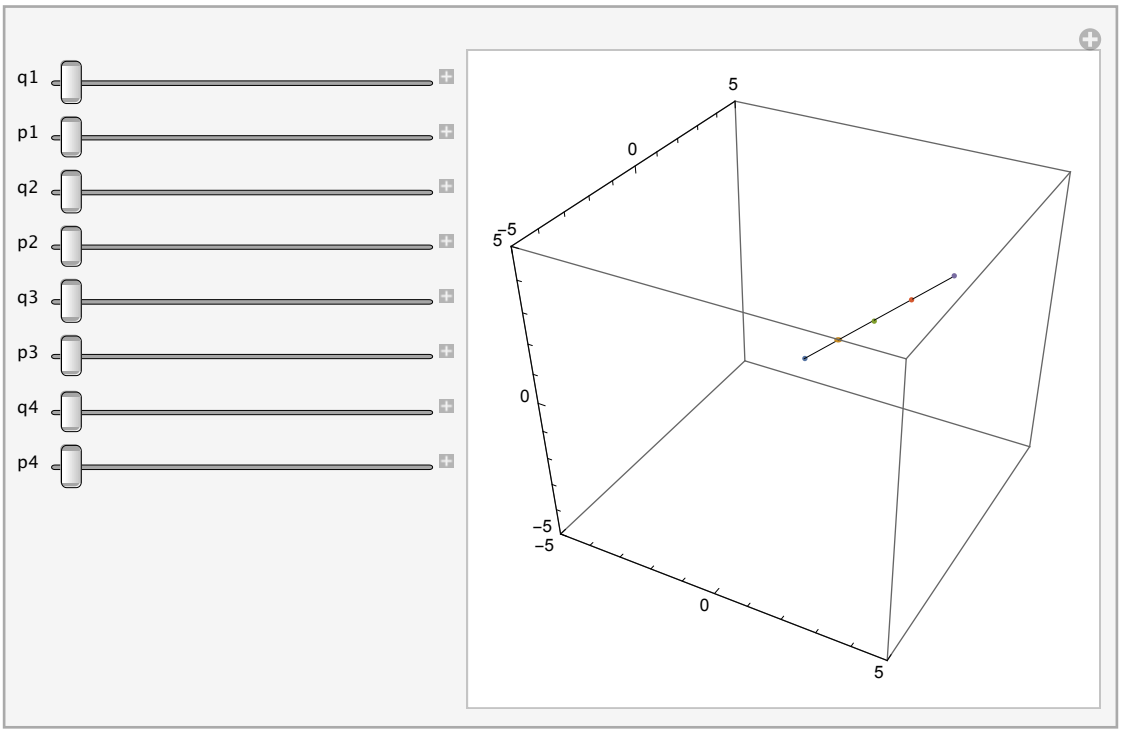
Dimensions[SEGMENTS]

{5, 3}

Part II. Allow for 3D angles

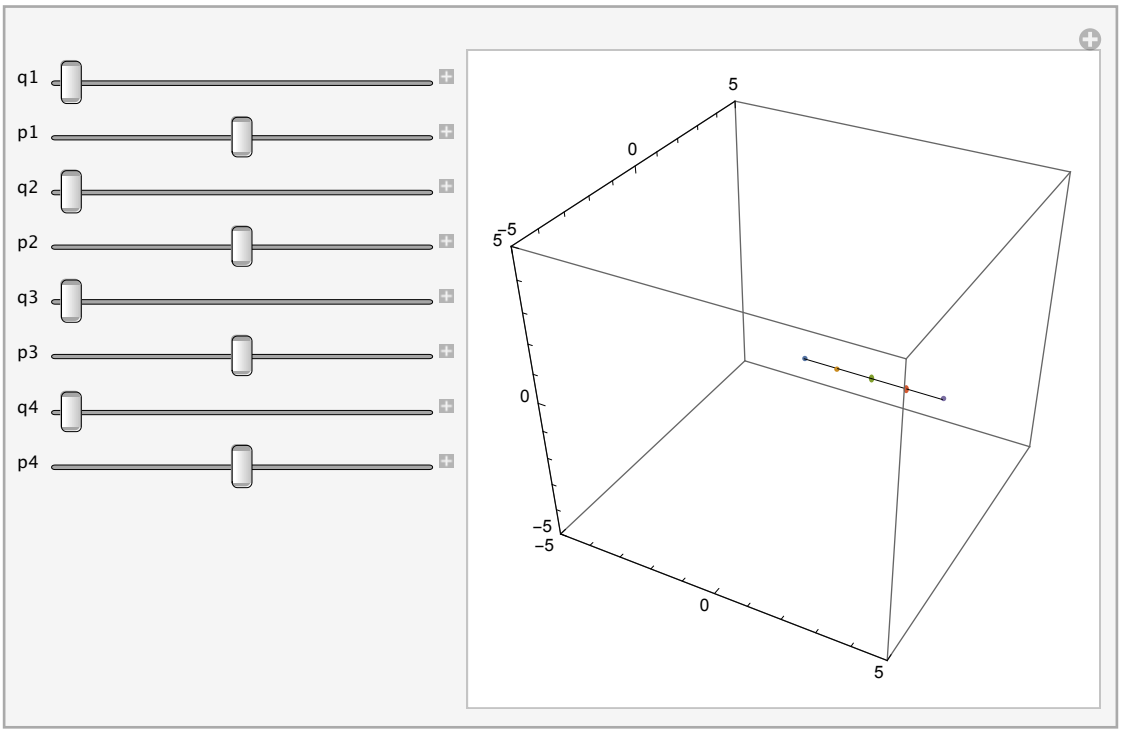
In[13]:=
```
(*DO THIS: - Split first joint angle
   into 1 and 2 for 'longitude' and 'latitude' angles,
respectively.  Note the additional Sin[latitude] term. Play
 with it to see how it works *)

range = 5;(*the maximum +/- range of the plot below*)
XYZrange = {{-range, range}, {-range, range}, {-range, range}};
l1 = 1;
l2 = 1;
l3 = 1;
l4 = 1;(*define segment lengths - we can change them later*)
Manipulate[
 x0 = 0;
 y0 = 0;
 z0 = 0;
 x1 = x0 + l1 * Cos[q1];
 y1 = y0 + l1 * Sin[q1];
 z1 = z0 + l1 * Cos[p1];
 x2 = x1 + l2 * Cos[q2];
 y2 = y1 + l2 * Sin[q2];
 z2 = z1 + l2 * Cos[p2];
 x3 = x2 + l3 * Cos[q3];
 y3 = y2 + l3 * Sin[q3];
 z3 = z2 + l3 * Cos[p3];
 x4 = x3 + l4 * Cos[q4];
 y4 = y3 + l4 * Sin[q4];
 z4 = z3 + l4 * Cos[p4];

 j0 = {x0, y0, z0};
 j1 = {x1, y1, z1};
 j2 = {x2, y2, z2};
 j3 = {x3, y3, z3};
 j4 = {x4, y4, z4};
 l1 = 1; l2 = 1; l3 = 1; l4 = 1;
 JOINTS = {{j0}, {j1}, {j2}, {j3}, {j4}};
 SEGMENTS = Flatten[JOINTS, 1];
 jointPlot = ListPointPlot3D[JOINTS, PlotRange → XYZrange, AspectRatio → 1];
 (*Plot the joints*)
 segmentPlot = Graphics3D[Line[SEGMENTS]];
 Show[jointPlot, segmentPlot, BoxRatios -> {1, 1, 1}, ImageSize → 300]
 (*superimpose the plots*)
 , {q1, 0, Pi}, {p1, 0, Pi}, {q2, 0, Pi}, {p2, 0, Pi},
 {q3, 0, Pi}, {p3, 0, Pi}, {q4, 0, Pi}, {p4, 0, Pi}]
```

Out[16]=



In[17]:=
```
(*DO THIS: - Split first joint angle
   into 1 and 2 for 'longitude' and 'latitude' angles,
respectively.  Note the additional Sin[latitude] term. Play
 with it to see how it works *)

range = 5;(*the maximum +/- range of the plot below*)
XYZrange = {{-range, range}, {-range, range}, {-range, range}};
l1 = 1;
l2 = 1;
l3 = 1;
l4 = 1;(*define segment lengths - we can change them later*)
Manipulate[
 x0 = 0;
 y0 = 0;
 z0 = 0;
 x1 = x0 + l1 * Cos[q1] * Sin[p1];
 y1 = y0 + l1 * Sin[q1] * Sin[p1];
 z1 = z0 + l1 * Cos[p1];
 x2 = x1 + l2 * Cos[q2] * Sin[p2];
 y2 = y1 + l2 * Sin[q2] * Sin[p2];
 z2 = z1 + l2 * Cos[p2];
 x3 = x2 + l3 * Cos[q3] * Sin[p3];
 y3 = y2 + l3 * Sin[q3] * Sin[p3];
 z3 = z2 + l3 * Cos[p3];
 x4 = x3 + l4 * Cos[q4] * Sin[p4];
 y4 = y3 + l4 * Sin[q4] * Sin[p4];
 z4 = z3 + l4 * Cos[p4];

 j0 = {x0, y0, z0};
 j1 = {x1, y1, z1};
 j2 = {x2, y2, z2};
 j3 = {x3, y3, z3};
 j4 = {x4, y4, z4};
 l1 = 1; l2 = 1; l3 = 1; l4 = 1;
 JOINTS = {{j0}, {j1}, {j2}, {j3}, {j4}};
 SEGMENTS = Flatten[JOINTS, 1];
 jointPlot = ListPointPlot3D[JOINTS, PlotRange → XYZrange, AspectRatio → 1];
 (*Plot the joints*)
 segmentPlot = Graphics3D[Line[SEGMENTS]];
 Show[jointPlot, segmentPlot, BoxRatios -> {1, 1, 1}, ImageSize → 300]
 (*superimpose the plots*)
 , {q1, 0, Pi}, {{p1, Pi / 2}, 0, Pi}, {q2, 0, Pi}, {{p2, Pi / 2}, 0, Pi},
 {q3, 0, Pi}, {{p3, Pi / 2}, 0, Pi}, {q4, 0, Pi}, {{p4, Pi / 2}, 0, Pi}]
```

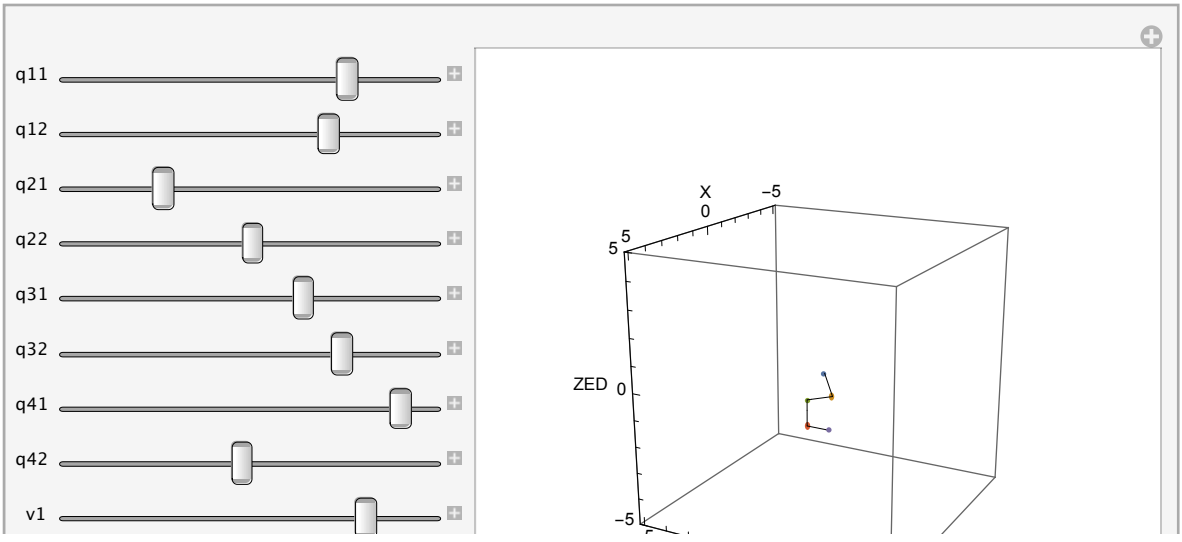
Out[20]=



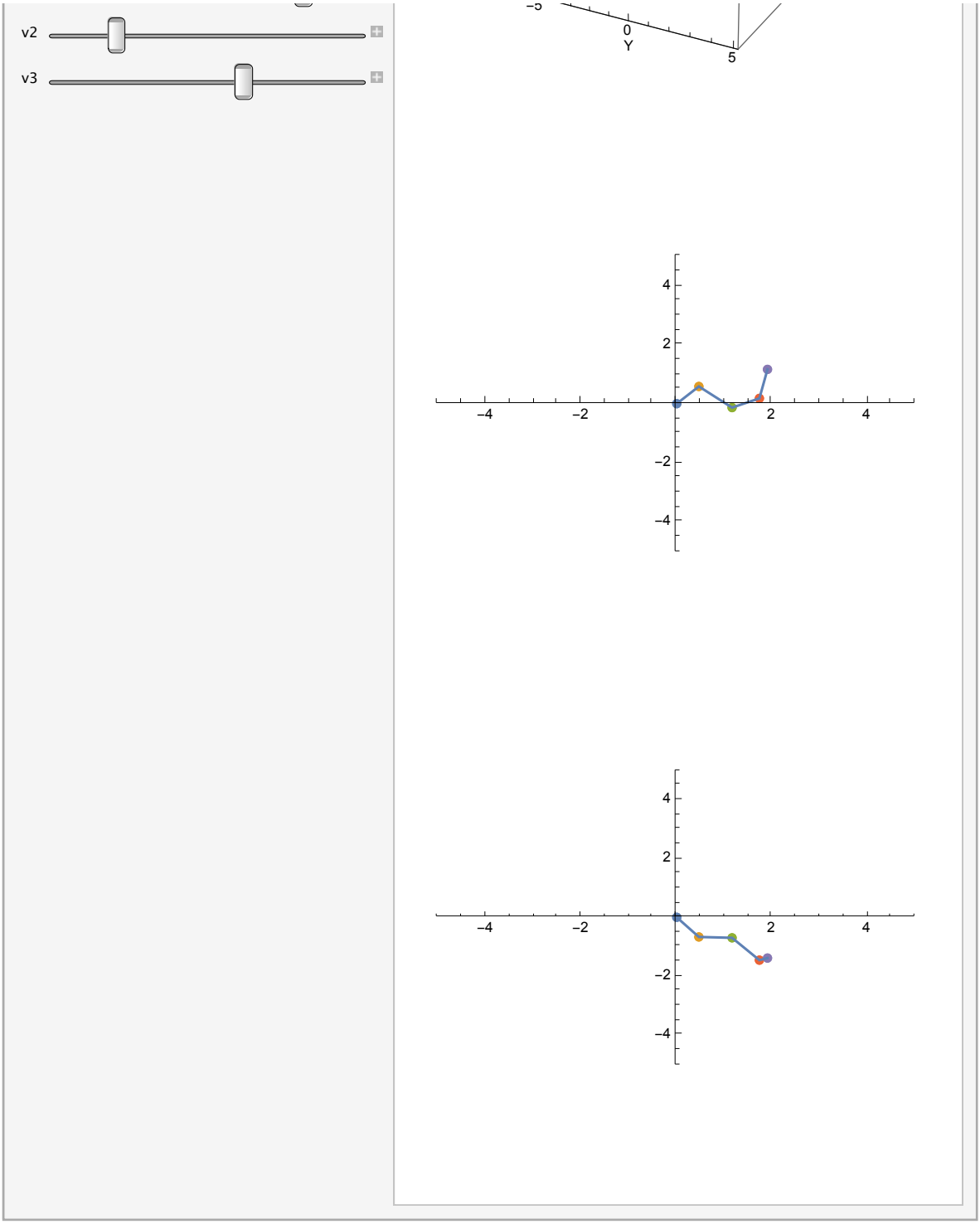
In[1]:=

```
(*DO THIS: - Clean up the code:
     make a function to build an xyz point based on the two angles theta =
 longitude and phi = latitude.  Also add the previous point*)
(*SDO THIS:- extract xz and yz points so we can make 2d
   plots in transverse (looking from the back of the animal),
and saggital (looking at the side) planes*)
(*DO THIS: - change the segment lengths so it's easier to see*)
(*DO THIS: set initial slider values*)

(*function to write x,y,z expressions in terms of prior joint position,
and theta and phi which are longitude and latitude angles, respectively*)
xyzJoint[jointXYZ_: {0, 0, 0}, theta_, phi_, length_] :=
 {jointXYZ[[1]] + length * Cos[theta] * Sin[phi]
  (*previous x coordinate + x coordinate*),
  jointXYZ[[2]] + length * Sin[theta] * Sin[phi]
  (*previous y coordinate + y coordinate*),
  jointXYZ[[3]] + length * Cos[phi](*previous z coordinate + z coordinate*)}
```

In[21]:=
```
range = 5;(*the maximum +/- range of the plot below*)
XYZrange = {{-range, range}, {-range, range}, {-range, range}};
l1 = 1; l2 = 1; l3 = .7; l4 = 1.5;
Qi = {0.9, 2.3, -0.8, 1.6, 0.5, 2.43, 1.4, 1.5};
(*initial slider values for angles*)
Manipulate[

 j0 = {0, 0, 0};
 j1 = xyzJoint[j0, q11, q12, l1];
 j2 = xyzJoint[j1, q21, q22, l2];
 j3 = xyzJoint[j2, q31, q32, l3];
 j4 = xyzJoint[j3, q41, q42, l4];
 JOINTS = {{j0}, {j1}, {j2}, {j3}, {j4}};

 JOINTSxy = JOINTS[[All, All, 1 ;; 2]];
 (*points projected in xy plane*)
 JOINTSxz = JOINTS[[All, All, {1, 3}]];
 (*points projected in xz plane*)
 SEGMENTS = Flatten[JOINTS, 1];
 SEGMENTSxy = Flatten[JOINTSxy, 1];

 SEGMENTSxz = Flatten[JOINTSxz, 1];
 jointPlot = ListPointPlot3D[JOINTS]; (*Plot the joints*)
 segmentPlot = Graphics3D[Line[SEGMENTS]];
 (*plot the projections*)
 jointPlotxy = ListPlot[JOINTSxy, PlotRange -> XYZrange[[ ;; 2]]];
 segmentPlotxy = ListLinePlot[SEGMENTSxy];
 jointPlotxz = ListPlot[JOINTSxz, PlotRange -> XYZrange[[ ;; 2]]];
 segmentPlotxz = ListLinePlot[SEGMENTSxz];
 plot1 = Show[jointPlot, segmentPlot, BoxRatios -> {1, 1, 1}, ImageSize → 300,
   PlotRange → XYZrange, AspectRatio → 1, AxesLabel → {"X", "Y", "ZED"},
   ImagePadding → 50, RotationAction → "Clip", ViewPoint → {v1, v2, v3}];
 (*superimpose the plots*)
 plot2 = Show[jointPlotxy, segmentPlotxy];
 plot3 = Show[jointPlotxz, segmentPlotxz];
 GraphicsColumn[{plot1, plot2, plot3}]
 (*assemble plots as a graphics column*)
 , {{q11, Qi[[1]]}, -Pi/2, Pi/2}, {{q12, Qi[[2]]}, 0, Pi},
 {{q21, Qi[[3]]}, -Pi/2, Pi/2}, {{q22, Qi[[4]]}, 0, Pi},
 {{q31, Qi[[5]]}, -Pi/2, Pi/2}, {{q32, Qi[[6]]}, 0, Pi},
 {{q41, Qi[[7]]}, -Pi/2, Pi/2}, {{q42, Qi[[8]]}, 0, Pi},
 {{v1, 3.4}, -5, 5}, {{v2, 1.8}, 0, 10}, {{v3, 1.3}, -5, 5}]
```



Out[25]=



Part III. Add a body to our 3D model

In[212]:=
```
(*STEP 1 - Build the body*)
range = 5;(*the maximum +/- range of the plot below*)
XYZrange = {{-range, range}, {-range, range}, {-range, range}};
```

```
lbod = 2; (*body length*)
wbod = 0.5; (*body width*)
yawangle = Pi / 2; (*swivel about z axis*)
pitchangle = Pi / 2; (*up-down pitching about x axis*)

xposition = 0;
yposition = 0;
zposition = 0;
(*the xy position of the animal's body*)
j0 = {xposition, yposition, zposition};
(*origin of body and leg as above*)
(*the body's orientation will be defined
 in terms of pitch (dorso-ventral rocking about x),
roll (left-right tilting about y) and yaw (left right swiveling about z*)
(*The body will be 4 points initially in the xy plane*)
(*we'll define an anterio-posterior axis starting at the origin, 0,0,0*)
(*the A-P axis will be in the center of our 4-point body plane*)
(*define the 4-point body plane*)

(*---xyz coordinates for front right body point-----*)
frontRx = j0[[1]] + lbod * Cos[yawangle] * Sin[pitchangle];
frontRy = j0[[2]] + lbod * Sin[yawangle] * Sin[pitchangle];
frontRz = j0[[3]] + lbod * Cos[pitchangle];
frontR = {frontRx, frontRy, frontRz};

(*----xyz coordinates for front left body point----*)
frontLx = frontR[[1]] + wbod * Cos[yawangle + Pi / 2];
frontLy = frontR[[2]] + wbod * Sin[yawangle + Pi / 2];
frontLz = frontR[[3]];
frontL = {frontLx, frontLy, frontLz};

(*----xyz coordinates for rear left body point----*)
rearLx = j0[[1]] + wbod * Cos[yawangle + Pi / 2];
rearLy = j0[[2]] + wbod * Sin[yawangle + Pi / 2];
rearLz = j0[[3]];
rearL = {rearLx, rearLy, rearLz};

rearR = j0; (*origin*)

pointsBODYg = {rearL, rearR, frontR, frontL, rearL};
(*points of the body in the global 'ground' reference frame.  rearL
 appears twice so it makes a closed object for plotting*)

(*this will make more sense later -- we'll define pointsBODY,
but we will later use pointsBODY to represent reorientations of the body frame-
 of-reference i.e. local reference frame *)
pointsBODY = pointsBODYg;
Show[ListPointPlot3D[pointsBODY], Graphics3D[Line[pointsBODY]],
 PlotRange → XYZrange, BoxRatios -> {1, 1, 1}, ImageSize → 300,
 PlotRange → XYZrange, AspectRatio → 1, AxesLabel → {"X", "Y", "ZED"},
 ImagePadding → 50, RotationAction → "Clip"]
```

In[2]:=

```
(*---BODY CONSTANTS----*)
range = 5;(*the maximum +/- range of the plot below*)
XYZrange = {{-range, range}, {-range, range}, {-range, range}};
lbod = 2; (*body length*)
```

```
wbod = 0.5; (*body width*)
yawangle = Pi / 2; (*swivel about z axis*)
pitchangle = Pi / 2;(*up-down pitching about x axis*)

(*----LEG CONSTANTS----*)
l1 = 1; l2 = 1; l3 = .7; l4 = 1.5;
Qi = {0.9, 2.3, -0.8, 1.6, 0.5, 2.43, 1.4, 1.5};
(*initial slider values for angles*)



Manipulate[
 (*---------------LIMB POINTS-------------------*)
 j0 = {0, 0, 0};
 j1 = xyzJoint[j0, q11, q12, l1];
 j2 = xyzJoint[j1, q21, q22, l2];
 j3 = xyzJoint[j2, q31, q32, l3];
 j4 = xyzJoint[j3, q41, q42, l4];
 JOINTS = {{j0}, {j1}, {j2}, {j3}, {j4}};

 JOINTSxy = JOINTS[[All, All, 1 ;; 2]];
 (*points projected in xy plane*)
 JOINTSxz = JOINTS[[All, All, {1, 3}]];
 (*points projected in xz plane*)
 SEGMENTS = Flatten[JOINTS, 1];
 SEGMENTSxy = Flatten[JOINTSxy, 1];

 SEGMENTSxz = Flatten[JOINTSxz, 1];
 jointPlot = ListPointPlot3D[JOINTS]; (*Plot the joints*)
 segmentPlot = Graphics3D[Line[SEGMENTS]];
 (*plot the projections*)
 jointPlotxy = ListPlot[JOINTSxy, PlotRange -> XYZrange[[ ;; 2]]];
 segmentPlotxy = ListLinePlot[SEGMENTSxy];
 jointPlotxz = ListPlot[JOINTSxz, PlotRange -> XYZrange[[ ;; 2]]];
 segmentPlotxz = ListLinePlot[SEGMENTSxz];

 (*-----------------------------------------*)


 (*--------------------BODY POINTS----------*)
 xposition = 0;
 yposition = 0;
 zposition = 0;
 (*the xy position of the animal's body*)
 j0 = {xposition, yposition, zposition};
 (*origin of body and leg as above*)
 (*the body's orientation will be defined
  in terms of pitch (dorso-ventral rocking about x),
 roll (left-right tilting about y) and yaw (left right swiveling about z*)
 (*The body will be 4 points initially in the xy plane*)
 (*we'll define an anterio-posterior axis starting at the origin, 0,0,0*)
 (*the A-P axis will be in the center of our 4-point body plane*)
 (*define the 4-point body plane*)

 (*---xyz coordinates for front right body point-----*)
 frontRx = j0[[1]] + lbod * Cos[yawangle] * Sin[pitchangle];
 frontRy = j0[[2]] + lbod * Sin[yawangle] * Sin[pitchangle];
 frontRz = j0[[3]] + lbod * Cos[pitchangle];
 frontR = {frontRx, frontRy, frontRz};

 (*----xyz coordinates for front left body point----*)
```

```
frontLx = frontR[[1]] + wbod * Cos[yawangle + Pi / 2];
frontLy = frontR[[2]] + wbod * Sin[yawangle + Pi / 2];
frontLz = frontR[[3]];
frontL = {frontLx, frontLy, frontLz};

(*----xyz coordinates for rear left body point----*)
rearLx = j0[[1]] + wbod * Cos[yawangle + Pi / 2];
rearLy = j0[[2]] + wbod * Sin[yawangle + Pi / 2];
rearLz = j0[[3]];
rearL = {rearLx, rearLy, rearLz};

rearR = j0; (*origin*)

pointsBODYg = {rearL, rearR, frontR, frontL, rearL};
(*points of the body in the global 'ground' reference frame.  rearL
 appears twice so it makes a closed object for plotting*)

(*this will make more sense later -- we'll define pointsBODY,
but we will later use pointsBODY to represent reorientations
  of the body frame-of-reference i.e. local reference frame *)
pointsBODY = pointsBODYg;

(*--------------------BODY POINTS----------*)

(*---------------PLOTS---------------------*)
plot1 = Show[jointPlot, segmentPlot, BoxRatios -> {1, 1, 1},
  ImageSize → 300, PlotRange → XYZrange, AspectRatio → 1,
  AxesLabel → {"X", "Y", "ZED"}, ImagePadding → 50, RotationAction → "Clip"];
(*superimpose the plots*)

plot2 = Show[ListPointPlot3D[pointsBODY], Graphics3D[Line[pointsBODY]],
  PlotRange → XYZrange, BoxRatios -> {1, 1, 1}, ImageSize → 300,
  PlotRange → XYZrange, AspectRatio → 1, AxesLabel → {"X", "Y", "ZED"},
  ImagePadding → 50, RotationAction → "Clip"];

Show[plot1, plot2]


, {{q11, Qi[[1]]}, -Pi / 2, Pi / 2}, {{q12, Qi[[2]]}, 0, Pi},
{{q21, Qi[[3]]}, -Pi / 2, Pi / 2}, {{q22, Qi[[4]]}, 0, Pi},
{{q31, Qi[[5]]}, -Pi / 2, Pi / 2}, {{q32, Qi[[6]]}, 0, Pi},
{{q41, Qi[[7]]}, -Pi / 2, Pi / 2}, {{q42, Qi[[8]]}, 0, Pi}]
```

In[8]:=
```
pointsLEG = {j0, j1, j2, j3, j4}
```

Out[8]=
```
{{0, 0, 0}, {0.463538, 0.584131, -0.666276}, {1.15995, -0.132919, -0.695476},
 {1.56112, 0.0862398, -1.2256}, {1.81543, 1.56071, -1.1195}}
```