```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class Matrix4By4
6  {
7
8      public Matrix4By4(MyVector4 column1, MyVector4 column2, MyVector4
         column3, MyVector4 column4)
9      {
10         values = new float[4, 4];
11
12         //column1
13         values[0, 0] = column1.x;
14         values[1, 0] = column1.y;
15         values[2, 0] = column1.z;
16         values[3, 0] = column1.w;
17         //column2
18         values[0, 1] = column2.x;
19         values[1, 1] = column2.y;
20         values[2, 1] = column2.z;
21         values[3, 1] = column2.w;
22         //column3
23         values[0, 2] = column3.x;
24         values[1, 2] = column3.y;
25         values[2, 2] = column3.z;
26         values[3, 2] = column3.w;
27         //column4
28         values[0, 3] = column4.x;
29         values[1, 3] = column4.y;
30         values[2, 3] = column4.z;
31         values[3, 3] = column4.w;
32     }
33     public Matrix4By4(MyVector3 column1, MyVector3 column2, MyVector3
         column3, MyVector3 column4)
34     {
35         values = new float[4, 4];
36
37         //column1
38         values[0, 0] = column1.x;
39         values[1, 0] = column1.y;
40         values[2, 0] = column1.z;
41         values[3, 0] = 0;
42         //column2
43         values[0, 1] = column2.x;
44         values[1, 1] = column2.y;
45         values[2, 1] = column2.z;
46         values[3, 1] = 0;
47         //column3
48         values[0, 2] = column3.x;
49         values[1, 2] = column3.y;
50         values[2, 2] = column3.z;
51         values[3, 2] = 0;
```

```
52          //column4
53          values[0, 3] = column4.x;
54          values[1, 3] = column4.y;
55          values[2, 3] = column4.z;
56          values[3, 3] = 1;
57      }
58      public float[,] values;
59      //public static Vector4 operator *(Matrix4By4 lhs, Vector4 rhs)
60      //{
61      //    Vector4 rv = new Vector4(0, 0, 0, 0);
62
63      //    rv.x = lhs.values[0, 0] * rhs.x + lhs.values[0, 1] * rhs.y +
          lhs.values[0, 2] + rhs.z + lhs.values[0, 3] * rhs.w;
64      //    rv.y = lhs.values[0, 0] * rhs.x + lhs.values[0, 1] * rhs.y +
          lhs.values[0, 2] + rhs.z + lhs.values[0, 3] * rhs.w;
65      //    rv.z = lhs.values[0, 0] * rhs.x + lhs.values[0, 1] * rhs.y +
          lhs.values[0, 2] + rhs.z + lhs.values[0, 3] * rhs.w;
66      //    rv.w = lhs.values[0, 0] * rhs.x + lhs.values[0, 1] * rhs.y +
          lhs.values[0, 2] + rhs.z + lhs.values[0, 3] * rhs.w;
67      //    return rv;
68      //}
69      //public static Vector4 operator *(Matrix4By4 lhs, Vector4 rhs)
70      //{
71      //    Vector4 rv = new Vector4(0, 0, 0);
72
73      //    rv.x = lhs.values[0, 0] * rhs.x + lhs.values[0, 1] * rhs.y +
          lhs.values[0, 2] * rhs.z + lhs.values[0, 3] * 1;
74      //    rv.y = lhs.values[1, 0] * rhs.x + lhs.values[1, 1] * rhs.y +
          lhs.values[1, 2] * rhs.z + lhs.values[1, 3] * 1;
75      //    rv.z = lhs.values[2, 0] * rhs.x + lhs.values[2, 1] * rhs.y +
          lhs.values[2, 2] * rhs.z + lhs.values[2, 3] * 1;
76      //    rv.w = lhs.values[3, 0] * rhs.x + lhs.values[3, 1] * rhs.y +
          lhs.values[3, 2] * rhs.z + lhs.values[3, 3] * 1;
77      //    //Debug.Log(lhs.values[0, 0]+"-"+ lhs.values[1, 1]+"-"+ lhs.values
          [2, 2]);
78      //    return rv;
79      //}
80      public static Vector4 operator *(Matrix4By4 lhs, Vector4 rhs)
81      {
82          Vector4 rv = new Vector4(0, 0, 0,0);
83
84          rv.x = lhs.values[0, 0] * rhs.x + lhs.values[0, 1] * rhs.y +
              lhs.values[0, 2] * rhs.z + lhs.values[0, 3] * 1;
85          rv.y = lhs.values[1, 0] * rhs.x + lhs.values[1, 1] * rhs.y +
              lhs.values[1, 2] * rhs.z + lhs.values[1, 3] * 1;
86          rv.z = lhs.values[2, 0] * rhs.x + lhs.values[2, 1] * rhs.y +
              lhs.values[2, 2] * rhs.z + lhs.values[2, 3] * 1;
87          rv.w = lhs.values[3, 0] * rhs.x + lhs.values[3, 1] * rhs.y +
              lhs.values[3, 2] * rhs.z + lhs.values[3, 3] * 1;
88          //Debug.Log(lhs.values[0, 0]+"-"+ lhs.values[1, 1]+"-"+ lhs.values
              [2, 2]);
89          return rv;
90      }
```

```
91      public static Vector4 Multiply(Matrix4By4 lhs1, Matrix4By4 lhs2, Vector4
           rhs)
92      {
93          Vector4 rv = new Vector4(0, 0, 0);
94
95          rv.x = lhs1.values[0, 0] * rhs.x + lhs1.values[0, 1] * rhs.y +
              lhs1.values[0, 2] * rhs.z + lhs1.values[0, 3] * 1;
96          rv.y = lhs1.values[1, 0] * rhs.x + lhs1.values[1, 1] * rhs.y +
              lhs1.values[1, 2] * rhs.z + lhs1.values[1, 3] * 1;
97          rv.z = lhs1.values[2, 0] * rhs.x + lhs1.values[2, 1] * rhs.y +
              lhs1.values[2, 2] * rhs.z + lhs1.values[2, 3] * 1;
98          rv.w = lhs1.values[3, 0] * rhs.x + lhs1.values[3, 1] * rhs.y +
              lhs1.values[3, 2] * rhs.z + lhs1.values[3, 3] * 1;
99          //Debug.Log(lhs.values[0, 0]+"-"+ lhs.values[1, 1]+"-"+ lhs.values
              [2, 2]);
100         return rv;
101     }
102     //public static Matrix4By4 operator *(Matrix4By4 lhs1, Matrix4By4 lhs2)
103     //{
104     //    Matrix4By4 rv;
105     //    rv = Matrix4By4.Identity;
106
107     //    rv.values[0, 0] = (lhs1.values[0, 0] * lhs2.values[0, 0]) +
          (lhs1.values[1, 0] * lhs2.values[0, 1]) + (lhs1.values[2, 0] *
          lhs2.values[0, 2]) + (lhs1.values[3, 0] * lhs2.values[0, 3]);
108     //    rv.values[1, 0] = (lhs1.values[0, 0] * lhs2.values[1, 0]) +
          (lhs1.values[1, 0] * lhs2.values[1, 1]) + (lhs1.values[2, 0] *
          lhs2.values[1, 2]) + (lhs1.values[3, 0] * lhs2.values[1, 3]);
109     //    rv.values[2, 0] = (lhs1.values[0, 0] * lhs2.values[2, 0]) +
          (lhs1.values[1, 0] * lhs2.values[2, 1]) + (lhs1.values[2, 0] *
          lhs2.values[2, 2]) + (lhs1.values[3, 0] * lhs2.values[2, 3]);
110     //    rv.values[3, 0] = (lhs1.values[0, 0] * lhs2.values[3, 0]) +
          (lhs1.values[1, 0] * lhs2.values[3, 1]) + (lhs1.values[2, 0] *
          lhs2.values[3, 2]) + (lhs1.values[3, 0] * lhs2.values[3, 3]);
111
112     //    rv.values[0, 1] = (lhs1.values[0, 1] * lhs2.values[0, 0]) +
          (lhs1.values[1, 1] * lhs2.values[0, 1]) + (lhs1.values[2, 1] *
          lhs2.values[0, 2]) + (lhs1.values[3, 1] * lhs2.values[0, 3]);
113     //    rv.values[1, 1] = (lhs1.values[0, 1] * lhs2.values[1, 0]) +
          (lhs1.values[1, 1] * lhs2.values[1, 1]) + (lhs1.values[2, 1] *
          lhs2.values[1, 2]) + (lhs1.values[3, 1] * lhs2.values[1, 3]);
114     //    rv.values[2, 1] = (lhs1.values[0, 1] * lhs2.values[2, 0]) +
          (lhs1.values[1, 1] * lhs2.values[2, 1]) + (lhs1.values[2, 1] *
          lhs2.values[2, 2]) + (lhs1.values[3, 1] * lhs2.values[2, 3]);
115     //    rv.values[3, 1] = (lhs1.values[0, 1] * lhs2.values[3, 0]) +
          (lhs1.values[1, 1] * lhs2.values[3, 1]) + (lhs1.values[2, 1] *
          lhs2.values[3, 2]) + (lhs1.values[3, 1] * lhs2.values[3, 3]);
116
117     //    rv.values[0, 2] = (lhs1.values[0, 2] * lhs2.values[0, 0]) +
          (lhs1.values[1, 2] * lhs2.values[0, 1]) + (lhs1.values[2, 2] *
          lhs2.values[0, 2]) + (lhs1.values[3, 2] * lhs2.values[0, 3]);
118     //    rv.values[1, 2] = (lhs1.values[0, 2] * lhs2.values[1, 0]) +
          (lhs1.values[1, 2] * lhs2.values[1, 1]) + (lhs1.values[2, 2] *
```

```
        lhs2.values[1, 2]) + (lhs1.values[3, 2] * lhs2.values[1, 3]);
119     //    rv.values[2, 2] = (lhs1.values[0, 2] * lhs2.values[2, 0]) +
        (lhs1.values[1, 2] * lhs2.values[2, 1]) + (lhs1.values[2, 2] *
        lhs2.values[2, 2]) + (lhs1.values[3, 2] * lhs2.values[2, 3]);
120     //    rv.values[3, 2] = (lhs1.values[0, 2] * lhs2.values[3, 0]) +
        (lhs1.values[1, 2] * lhs2.values[3, 1]) + (lhs1.values[2, 2] *
        lhs2.values[3, 2]) + (lhs1.values[3, 2] * lhs2.values[3, 3]);
121
122     //    rv.values[0, 3] = (lhs1.values[0, 3] * lhs2.values[0, 0]) +
        (lhs1.values[1, 3] * lhs2.values[0, 1]) + (lhs1.values[2, 3] *
        lhs2.values[0, 2]) + (lhs1.values[3, 3] * lhs2.values[0, 3]);
123     //    rv.values[1, 3] = (lhs1.values[0, 3] * lhs2.values[1, 0]) +
        (lhs1.values[1, 3] * lhs2.values[1, 1]) + (lhs1.values[2, 3] *
        lhs2.values[1, 2]) + (lhs1.values[3, 3] * lhs2.values[1, 3]);
124     //    rv.values[2, 3] = (lhs1.values[0, 3] * lhs2.values[2, 0]) +
        (lhs1.values[1, 3] * lhs2.values[2, 1]) + (lhs1.values[2, 3] *
        lhs2.values[2, 2]) + (lhs1.values[3, 3] * lhs2.values[2, 3]);
125     //    rv.values[3, 3] = (lhs1.values[0, 3] * lhs2.values[3, 0]) +
        (lhs1.values[1, 3] * lhs2.values[3, 1]) + (lhs1.values[2, 3] *
        lhs2.values[3, 2]) + (lhs1.values[3, 3] * lhs2.values[3, 3]);
126     //    //Debug.Log(lhs.values[0, 0]+"-"+ lhs.values[1, 1]+"-"+ lhs.values
        [2, 2]);
127     //    return rv;
128     //}
129
130     public static Matrix4By4 operator *(Matrix4By4 lhs1, Matrix4By4 lhs2)
131     {
132         Matrix4By4 rv;
133         rv = Matrix4By4.Identity;
134
135
136         for (int v = 0; v < 4; v++)
137         {
138             for (int i = 0; i < 4; i++)
139             {
140                 rv.values[i, v] = (lhs1.values[0, v] * lhs2.values[i, 0]) +
                    (lhs1.values[1, v] * lhs2.values[i, 1]) + (lhs1.values[2,
                    v] * lhs2.values[i, 2]) + (lhs1.values[3, v] * lhs2.values
                    [i, 3]);
141             }
142         }
143
144         //Debug.Log(lhs.values[0, 0]+"-"+ lhs.values[1, 1]+"-"+ lhs.values
            [2, 2]);
145         return rv;
146     }
147
148     public static Matrix4By4 Identity
149     {
150         get
151         {
152          return new Matrix4By4(new MyVector4(1, 0, 0, 0),
153                                new MyVector4(0, 1, 0, 0),
```

```
154                                 new MyVector4(0, 0, 1, 0),
155                                 new MyVector4(0, 0, 0, 1));
156         }
157     }
158
159     //public static Matrix4By4 GetRow(float row)
160     //{
161     //    Matrix4By4 rv = matrix;
162
163
164
165     //}
166 }
167
168 public class MyVector4
169 {
170     public float x, y, z, w;
171
172     public MyVector4(float x, float y, float z,float w)
173     {
174         this.x = x;
175         this.y = y;
176         this.z = z;
177         this.w = w;
178     }
179
180     public Vector4 ToUnityVector()
181     {
182         //Converts class MyVector3 to the Vector3 unity
183         Vector4 rv = new Vector4(x, y, z,w);
184
185
186         return rv;
187     }
188     public static MyVector4 Zero()
189     {
190         //Converts class MyVector3 to the Vector3 unity
191         MyVector4 rv = new MyVector4(0,0,0,0);
192
193
194         return rv;
195     }
196
197     public MyVector4 GetAxisAngle()
198     {
199         MyVector4 rv = new MyVector4(0,0,0,0);
200
201         float halfAngle = Mathf.Acos(w);
202         rv.w = halfAngle * 2;
203
204         rv.x = x / Mathf.Sin(halfAngle);
205         rv.y = y / Mathf.Sin(halfAngle);
206         rv.z = z / Mathf.Sin(halfAngle);
```

```
207
208         return rv;
209     }
210
211     public static MyVector4 QuaternionToVector4(MyQuaternion myquat)
212     {
213         MyVector4 rv = new MyVector4(0,0,0,0);
214
215         rv.w = myquat.w;
216         rv.x = myquat.x;
217         rv.y = myquat.y;
218         rv.z = myquat.z;
219
220         return rv;
221     }
222 }
```