**实现鼠标拖拽旋转与滚轮缩放**

Posted on 2013年05月10日 by U3d / [Unity3D脚本/插件](http://www.unitymanual.com/category/script)/被围观 100 次

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | **var** target : Transform; |
| 02 | **var** xSpeed = 250.0; |
| 03 | **var** ySpeed = 120.0; |
| 04 |  |
| 05 | **var** yMinLimit = -20; |
| 06 | **var** yMaxLimit = 80; |
| 07 |  |
| 08 | **var** initDis = 20; |
| 09 | **var** minDis = 3.0; |
| 10 | **var** maxDis = 50.0; |
| 11 |  |
| 12 | **var** wheelSpeed = 5; |
| 13 |  |
| 14 | **static** **var** x = 0.0; |
| 15 | **static** **var** y = 0.0; |
| 16 |  |
| 17 | **static** **var** distance; |
| 18 |  |
| 19 | **private** **var** position; |
| 20 | **private** **var** rotation; |
| 21 |  |
| 22 | function Start () { |
| 23 |  |
| 24 | x = 130; |
| 25 | y = 30;&nbsp; Unity3D教程手册 |
| 26 |  |
| 27 |  |
| 28 |  |
| 29 |  |
| 30 | transform.rotation = Quaternion.Euler(y, x, 0);; |
| 31 | transform.position = Quaternion.Euler(y, x, 0) \* Vector3(0.0, 0.0, -initDis) + target.position; |
| 32 |  |
| 33 | *// Make the rigid body not change rotation* |
| 34 | **if** (rigidbody) |
| 35 | rigidbody.freezeRotation = **true**; |
| 36 |  |
| 37 | } |
| 38 |  |
| 39 | function Update () { |
| 40 | **if** (target) { |
| 41 | distance = Vector3.Distance(target.position,transform.position); |
| 42 | **if**(Input.GetMouseButton(1)){ |
| 43 | x += Input.GetAxis("Mouse X") \* xSpeed \* 0.02; |
| 44 | y -= Input.GetAxis("Mouse Y") \* ySpeed \* 0.02; |
| 45 |  |
| 46 | y = ClampAngle(y, yMinLimit, yMaxLimit); |
| 47 | } |
| 48 |  |
| 49 | distance-= Input.GetAxis("Mouse ScrollWheel")\*wheelSpeed;*//获取鼠标中建响应* |
| 50 | distance = Mathf.Clamp(distance,minDis,maxDis);*//距离取最大值和最小值* |
| 51 |  |
| 52 | rotation = Quaternion.Euler(y, x, 0); |
| 53 | position = rotation \* Vector3(0.0, 0.0, -distance) + target.position; |
| 54 |  |
| 55 | transform.rotation = rotation; |
| 56 | transform.position = position; |
| 57 |  |
| 58 | } |
| 59 | } |
| 60 |  |
| 61 | **static** function ClampAngle (angle : **float**, min : **float**, max : **float**) { |
| 62 | **if** (angle < -360) |
| 63 | angle += 360; |
| 64 | **if** (angle > 360) |
| 65 | angle -= 360; |
| 66 | **return** Mathf.Clamp (angle, min, max); |
| 67 | } |
| 68 |  |
| 69 |  |