**Unity3D脚本：导弹追踪代码**

Posted on 2013年02月20日 by U3d / [Unity3D脚本/插件](http://www.unitymanual.com/category/script)/被围观 453 次

[**Unity3D**](http://www.unitymanual.com)导弹追踪，方法一简单，但效果一般，建议采用方法二。

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

|  |  |
| --- | --- |
| 001 | **using** UnityEngine; |
| 002 |  |
| 003 | **using** System.Collections; |
| 004 |  |
| 005 | **public** **class** AIFollow : MonoBehaviour { |
| 006 |  |
| 007 | **public** GameObject man; |
| 008 |  |
| 009 | **public** GameObject missile; |
| 010 |  |
| 011 | **public** **float** manSpeed = 6f; |
| 012 |  |
| 013 | **public** **float** missileSpeed = 4f; |
| 014 |  |
| 015 | **public** **float** missileRotateSpeed = 2f; |
| 016 |  |
| 017 | **bool** whehterShooted = **false**; |
| 018 |  |
| 019 | **float** distance; |
| 020 |  |
| 021 | **float** collisionDistance; |
| 022 |  |
| 023 | *// Use this for initialization* |
| 024 |  |
| 025 | **void** Start () { |
| 026 |  |
| 027 | **if** (man!=**null**&&missile!=**null**) |
| 028 |  |
| 029 | { |
| 030 |  |
| 031 | **float** manWidth = man.GetComponent().mesh.bounds.size.x\*man.transform.localScale.x; |
| 032 |  |
| 033 | **float** missileLength = missile.GetComponent().mesh.bounds.size.z \* missile.transform.localScale.z; |
| 034 |  |
| 035 | print("manWidth:"+manWidth.ToString()+",missileLength:"+missileLength.ToString()); |
| 036 |  |
| 037 | collisionDistance = manWidth / 2 + missileLength / 2; |
| 038 |  |
| 039 | } |
| 040 |  |
| 041 | } |
| 042 |  |
| 043 | *// Update is called once per frame* |
| 044 |  |
| 045 | **void** Update () { |
| 046 |  |
| 047 | **if** (Input.GetKey (KeyCode.LeftArrow)) |
| 048 |  |
| 049 | { |
| 050 |  |
| 051 | man.transform.Translate(Vector3.left\*Time.deltaTime\*manSpeed); |
| 052 |  |
| 053 | } |
| 054 |  |
| 055 | **if** (Input.GetKey(KeyCode.RightArrow)) |
| 056 |  |
| 057 | { |
| 058 |  |
| 059 | man.transform.Translate(-Vector3.left \* Time.deltaTime \* manSpeed); |
| 060 |  |
| 061 | } |
| 062 |  |
| 063 | **if** (Input.GetKey(KeyCode.UpArrow)) |
| 064 |  |
| 065 | { |
| 066 |  |
| 067 | man.transform.Translate(Vector3.forward \* Time.deltaTime \* manSpeed); |
| 068 |  |
| 069 | } |
| 070 |  |
| 071 | **if** (Input.GetKey(KeyCode.DownArrow)) |
| 072 |  |
| 073 | { |
| 074 |  |
| 075 | man.transform.Translate(-Vector3.forward \* Time.deltaTime \* manSpeed); |
| 076 |  |
| 077 | } |
| 078 |  |
| 079 | **if** (Input.GetKeyDown(KeyCode.S))*//按下S 发射导弹* |
| 080 |  |
| 081 | { |
| 082 |  |
| 083 | whehterShooted = **true**; |
| 084 |  |
| 085 | } |
| 086 |  |
| 087 | **if** (whehterShooted&&missile!=**null**) |
| 088 |  |
| 089 | { |
| 090 |  |
| 091 | distance = Vector3.Distance(man.transform.position, missile.transform.position); |
| 092 |  |
| 093 | *////导弹朝向人 法一* |
| 094 |  |
| 095 | *//missile.transform.LookAt(man.transform);* |
| 096 |  |
| 097 | *//导弹朝向人 法二* |
| 098 |  |
| 099 | Quaternion missileRotation = Quaternion.LookRotation(man.transform.position - missile.transform.position, Vector3.up); |
| 100 |  |
| 101 | *//missile.transform.rotation = Quaternion.Slerp(missile.transform.rotation, missileRotation, Time.deltaTime \* missileRotateSpeed);* |
| 102 |  |
| 103 | missile.transform.rotation = missileRotation; |