**Unity3D脚本：人物控制脚本的实现**

Posted on 2013年02月01日 by U3d / [Unity3D脚本/插件](http://www.unitymanual.com/category/script)/被围观 421 次

对D3D的空间几何知识是必要的，而这里要注意的是U3D提供了一个方法来判断是否与地面相碰撞，\_grounded = (\_flags & CollisionFlags.CollidedBelow) != 0;

其中CollisionFlags.CollidedBelow要想获得一个值，玩家的移动方向必须有朝向地面的方向向量分量。\_grounded = (\_flags & CollisionFlags.CollidedBelow) != 0;

为了在跳跃过程中更具有真实性，加入了重力加速度的概念，注意\_gravity表示的是“重力速度”，也就是向下的一个向量分量。

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

|  |  |
| --- | --- |
| 001 | *// Require a character controller to be attached to the same game object* |
| 002 |  |
| 003 | @script RequireComponent(CharacterController) |
| 004 |  |
| 005 | **var** \_speed : **float**; |
| 006 |  |
| 007 | **var** \_jumpspeed : **float**; |
| 008 |  |
| 009 | **var** \_rotatespeed : **float**; |
| 010 |  |
| 011 | **var** \_originalgravity : **float**; |
| 012 |  |
| 013 | **var** \_gravity : **float**; |
| 014 |  |
| 015 | **var** \_addgravity : **float**; |
| 016 |  |
| 017 | **var** \_moveDirection: Vector3; |
| 018 |  |
| 019 | **var** \_lastDirection: Vector3; |
| 020 |  |
| 021 | **var** \_originalRotation : Quaternion; |
| 022 |  |
| 023 | **var** \_flags; |
| 024 |  |
| 025 | **var** \_grounded; |
| 026 |  |
| 027 | **var** \_canjump; |
| 028 |  |
| 029 | **var** \_animation; |
| 030 |  |
| 031 | **var** \_run : AnimationClip; |
| 032 |  |
| 033 | **var** \_idle : AnimationClip; |
| 034 |  |
| 035 | **var** \_attack : AnimationClip; |
| 036 |  |
| 037 | **var** \_jump : AnimationClip; |
| 038 |  |
| 039 | **private** **var** \_controller : CharacterController; |
| 040 |  |
| 041 | **enum** PlayerState |
| 042 |  |
| 043 | { |
| 044 |  |
| 045 | STATE\_READY = 0, |
| 046 |  |
| 047 | STATE\_RUN = 1, |
| 048 |  |
| 049 | STATE\_IDLE = 2, |
| 050 |  |
| 051 | STATE\_ATTACK = 3, |
| 052 |  |
| 053 | STATE\_JUMP = 4, |
| 054 |  |
| 055 | } |
| 056 |  |
| 057 | **private** **var** \_playerstate : PlayerState; |
| 058 |  |
| 059 | function Start(){ |
| 060 |  |
| 061 | \_speed = 50F; |
| 062 |  |
| 063 | \_jumpspeed = 40F; |
| 064 |  |
| 065 | \_rotatespeed = 10F; |
| 066 |  |
| 067 | \_originalgravity = 40; |
| 068 |  |
| 069 | \_gravity = 40; |
| 070 |  |
| 071 | \_addgravity = 3; |
| 072 |  |
| 073 | \_canjump = **true**; |
| 074 |  |
| 075 | \_moveDirection = Vector3.zero; |
| 076 |  |
| 077 | \_lastDirection = Vector3.zero; |
| 078 |  |
| 079 | \_originalRotation = new Quaternion(0f, 0f, 0f, 1f); |
| 080 |  |
| 081 | \_grounded = **true**; |
| 082 |  |
| 083 | \_animation = GetComponent(Animation); |
| 084 |  |
| 085 | \_controller = GetComponent(CharacterController); |
| 086 |  |
| 087 | \_playerstate = PlayerState.STATE\_READY; |
| 088 |  |
| 089 | \_moveDirection.y = -1; *//给Y轴方向一个初始值，以使其可以与地面正确相撞* |
| 090 |  |
| 091 | } |
| 092 |  |
| 093 | function Update () { |
| 094 |  |
| 095 | *//模型的旋转（临时）* |
| 096 |  |
| 097 | **if**(Input.GetKey("a")){ |
| 098 |  |
| 099 | transform.Rotate(-Vector3.up\*2); |
| 100 |  |
| 101 | } |
| 102 |  |
| 103 | **if**(Input.GetKey("d")){ |
| 104 |  |
| 105 | transform.Rotate(Vector3.up\*2); |
| 106 |  |
| 107 | } |
| 108 |  |
| 109 | **if**(\_grounded) |
| 110 |  |
| 111 | { |
| 112 |  |
| 113 | *//本地空间水平移动方向向量* |
| 114 |  |
| 115 | \_moveDirection = Vector3( 0 , -1 ,Input.GetAxis("Vertical")); |
| 116 |  |
| 117 | *//本地空间移动方向转为世界空间的移动方向向量* |
| 118 |  |
| 119 | \_moveDirection = transform.TransformDirection(\_moveDirection); |
| 120 |  |
| 121 | \_moveDirection \*= \_speed; |
| 122 |  |
| 123 | *//重力速度还原* |
| 124 |  |
| 125 | \_gravity = \_originalgravity; |
| 126 |  |
| 127 | } |
| 128 |  |
| 129 | *//跳跃* |
| 130 |  |
| 131 | **if** (Input.GetButton("Jump") ) |
| 132 |  |
| 133 | { |
| 134 |  |
| 135 | \_playerstate = PlayerState.STATE\_JUMP; |
| 136 |  |
| 137 | *//获得初速度* |
| 138 |  |
| 139 | \_moveDirection.y = \_jumpspeed; |
| 140 |  |
| 141 | \_animation.Play("jump"); |
| 142 |  |
| 143 | } |
| 144 |  |
| 145 | *//如果当前处于跳跃中，就算一下Y向量* |
| 146 |  |
| 147 | **if**(\_playerstate == PlayerState.STATE\_JUMP) |
| 148 |  |
| 149 | { |
| 150 |  |
| 151 | \_gravity += \_addgravity; |
| 152 |  |
| 153 | \_moveDirection.y -= \_gravity \* Time.deltaTime; |
| 154 |  |
| 155 | } |
| 156 |  |
| 157 | *//移动，并获得碰撞标示* |
| 158 |  |
| 159 | **var** \_flags : CollisionFlags =\_controller.Move(\_moveDirection \*Time.deltaTime); |
| 160 |  |
| 161 | *//是否已经着地* |
| 162 |  |
| 163 | \_grounded = (\_flags & CollisionFlags.CollidedBelow) != 0; |
| 164 |  |
| 165 | print(\_grounded); |
| 166 |  |
| 167 | *//=======================================================================* |
| 168 |  |
| 169 | *//攻击* |
| 170 |  |
| 171 | **if**(Input.GetKeyUp("f") && \_grounded &&\_playerstate == PlayerState.STATE\_IDLE) |
| 172 |  |
| 173 | { |
| 174 |  |
| 175 | \_playerstate = PlayerState.STATE\_ATTACK; |
| 176 |  |
| 177 | \_animation.Play("attack"); |
| 178 |  |
| 179 | } |
| 180 |  |
| 181 | *//休闲* |
| 182 |  |
| 183 | **if**(\_moveDirection.x == 0 && \_moveDirection.z == 0&& \_grounded && \_playerstate != PlayerState.STATE\_ATTACK) |
| 184 |  |
| 185 | **if**(\_moveDirection != Vector3(0,0,0)) |
| 186 |  |
| 187 | { |
| 188 |  |
| 189 | **if**(\_playerstate != PlayerState.STATE\_ATTACK) |
| 190 |  |
| 191 | { |
| 192 |  |
| 193 | \_playerstate = PlayerState.STATE\_IDLE; |
| 194 |  |
| 195 | \_animation.Play("idle"); |
| 196 |  |
| 197 | } |
| 198 |  |
| 199 | } |
| 200 |  |
| 201 | *//跑动* |
| 202 |  |
| 203 | **if**((\_moveDirection.x != 0 || \_moveDirection.z != 0) &&\_grounded ) |
| 204 |  |
| 205 | { |
| 206 |  |
| 207 | \_playerstate = PlayerState.STATE\_RUN; |
| 208 |  |
| 209 | \_animation.Play("run"); |
| 210 |  |
| 211 | } |
| 212 |  |
| 213 | } |
| 214 |  |
| 215 | *//供外部UI调用的接口，调用攻击动作* |
| 216 |  |
| 217 | function PlayAnimation\_Attack() |
| 218 |  |
| 219 | { |
| 220 |  |
| 221 | **if**(\_grounded && (\_playerstate ==PlayerState.STATE\_IDLE||\_playerstate ==PlayerState.STATE\_ATTACK)) |
| 222 |  |
| 223 | { |
| 224 |  |
| 225 | \_playerstate = PlayerState.STATE\_ATTACK; |
| 226 |  |
| 227 | \_animation.Play("attack"); |
| 228 |  |
| 229 | } |
| 230 |  |
| 231 | } |