

背包界面设计

一：CocoStudio设计

二：数据设计

三：代码设计

一：CocoStudio设计

在游戏里面，我们经常会遇到装备背包界面，里面的元素有的是一行一行的，有的是一行里面有多个，不管怎么设计，大体都会遵循一个思路，就是ListView里面存放每一个单元格，单元格里面控制具体元素的布局。

下面我们从CocoStudio的角度来进行背包界面的设计。

版本说明：quick3.5 final搭配CocoStudio2.2.1，其他的版本会出现错误。

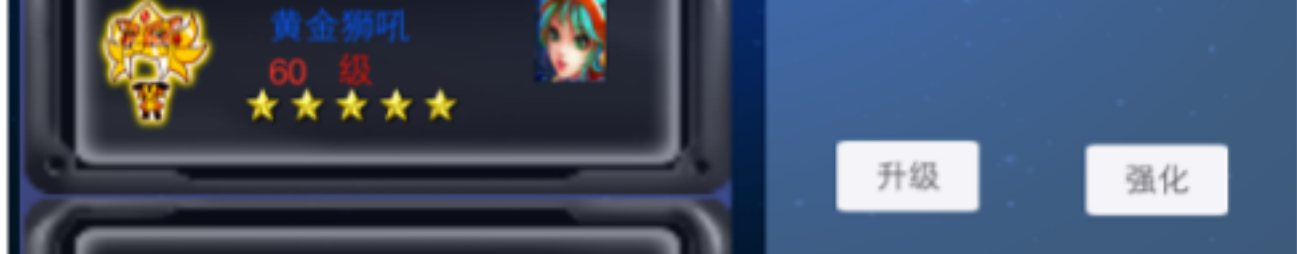
步骤如下

(1) 新建一个CocoStudio工程，命名为PackageDemo，同时倒入我们上课提供的图片资源。

(2) 新建一个层，命名为EQCell，同时大小设置为400*150，为了匹配ListView的大小。

(3) 根据项目的UI要求，添加ImageView, Text等控件，并且进行规范的命名。图片如下所示：





页码: 2/3

www.neworigin.net

(4) 新建一个PackageLayer, 用来进行ListView以及右侧详细界面的布局, 这里主要有两个元素, ListView以及基础Panel, 右侧的具体信息全部布局在基础面板之上, 之后我们获得子节点的同时就需要根据这个来获取。在此注意每个控件的命名, 之后我们要使用它。

(5) 设计完成之后, 发布资源, 并且将其拖动我们项目的res文件夹下。注: 这里我们采用真实目录

二: 数据设计

游戏的数据无非就是几种格式, xml, json, excel, csv, table等, 在Cocos里面我们经常使用json, 在lua里面我们最常用的就是table, 将一些基础的装备信息存放在table里面, 当我们使用的时候, 通过table数据的获取方法, 很简单的就能取得数据。我们在此使用的数据结构如下所示

```
local EQData = {  
    [1] = {name = "小熊战袍", kind="仙器", eq_kind = "盔甲", lv = 50, star = 4, hero = 1, add = {health = 100, atk = 0, def = 100}},  
    [2] = {name = "紫狼战甲", kind="绝世", eq_kind = "盔甲", lv = 60, star = 5, hero = 2, add = {health = 200, atk = 100, def = 200}},  
    [3] = {name = "黄金狮吼", kind="绝世", eq_kind = "盔甲", lv = 60, star = 5, hero = 2, add = {health = 200, atk = 100, def = 200}},  
    [4] = {name = "黄金圣衣", kind="绝世", eq_kind = "盔甲", lv
```

```

[4] = {name = "黄金圣衣",kind="绝世",eq_kind = "盔甲",lv =50,star
=60 ,star =5 ,hero =3 ,add = {health = 200 ,atk =100 ,def =
200 }},
[5] = {name = "龙骑战衣",kind="仙器",eq_kind = "盔甲",lv =50,star
=4 ,hero =1,add = {health = 100 ,atk =0 ,def =100 }},
[6] = {name = "木簪",kind="凡器" ,eq_kind = "头饰" ,lv =1 ,star
=0 ,hero = 0,add = {health =50 ,atk =0 ,def =50 }},
[7] = {name = "金霞棒",kind="名器" ,eq_kind = "武器" ,lv = 1,star
= 2,hero = 0,add = {health =0 ,atk = 100,def = 0}},

```

页码: 3/4

www.neworigin.net

```

[8] = {name = "玄奘法杖",kind="仙器" ,eq_kind = "武器" ,lv
=2 ,star =3 ,hero =0 ,add = {health =100 ,atk =100 ,def =0 }},

[9] = {name = "金手臂",kind="名器" ,eq_kind = "护腕" ,lv =
3,star =3 ,hero =0 ,add = {health =100 ,atk =0 ,def =0 }},
[10] = {name = "紫凤霞衣",kind="仙器",eq_kind = "盔甲",lv =
70,star = 3,hero =0 ,add = {health =100 ,atk =0 ,def =100 }},
}

```

三：代码设计

(1) 我们首先需要创建ListView每一个选项对应的cell，创建代码如下所示，主要就是一个class，继承自ccui.Widget，这样我们才能正确的添加进入ListView。创建代码如下所示：

```

local EQCell = class("EQCell", function()
    return ccui.Widget:create()
end)

function EQCell:ctor(eq_id)
    self.id = eq_id
    self.cell = cc.ui.loader:load("EQCell.csb")
    self:addChild(self.cell)
    self:setContentSize(self.cell:getContentSize())
    self:initUI()

```

```

end

function EQCell:initUI()
    self.icon = self.cell:getChildByName("img_icon")
    self.lb_lv = self.cell:getChildByName("lb_lv")
    self.lb_name = self.cell:getChildByName("lb_name")
    self.star = {}
    for i=1,5 do
        self.star[i] = self.cell:getChildByName("star_"..i)
    end
    self.hero = self.cell:getChildByName("img_hero")

    self:refreshUI()
end

```

页码: 4/5

www.neworigin.net

```

end

function EQCell:refreshUI()
    local data = EQData[self.id]
    self.icon:loadTexture("eq_"..self.id.."png")
    if data.hero ~= 0 then
        local hero_img = "hero_"..data.hero.."jpg"

        print(hero_img)
        self.hero:loadTexture(hero_img)
    end

    self.lb_lv:setString(tostring(data.lv))
    self.lb_name:setString(data.name)
    for i=1,data.star do
        self.star[i]:loadTexture("star_on.png")
    end
end
end

```

综合上述代码，主要思想就是创建一个id，用来标识每一个cell，然后根据id从table里面读取对应的元素，进行界面的展示即可。

(2) 详细信息界面的代码设计

此界面要求我们点击单元格的时候，能够从里面获取

此界面要求我们点击单元格的时候，能够从里面获取点击的下标，并且展示具体的数据。具体代码如下所示：

```
function MainScene:ctor()
    self.selected_index = 0
    self:initUI()
end

function MainScene:initUI()
    -- self.layer = cc.CSLoader:createNode("PackageLayer.csb")
    self.layer = cc.uiloader:load("PackageLayer.csb")
    self.layer:setAnchorPoint(cc.p(0, 0))
    self.layer:pos(0, 0)
    self:addChild(self.layer)

    self.listView = self.layer:getChildByName("EQL")

    self.eqdes = self.layer:getChildByName("Des")
```

页码: 5/6

www.neworigin.net

```
self.btn_lv_up = self.eqdes:getChildByName("btn_up")
self.btn_star_up = self.eqdes:getChildByName("btn_star_up")
self.eqdes_icon = self.eqdes:getChildByName("des_icon")
self.eqdes_kind = self.eqdes:getChildByName("des_kind")
self.star = {}
for i=1,5 do
    self.star[i] = self.eqdes:getChildByName("star_"..i)
end
self.eqdes_name = self.eqdes:getChildByName("des_name")
self.lb_kind = self.eqdes:getChildByName("lb_kind")
self.lb_lv = self.eqdes:getChildByName("lb_lv")
self.lb_des = self.eqdes:getChildByName("lb_des")
self:refreshListView()
end

function MainScene:refreshListView()
    self.cell = {}
    for i=1,#EQData do
        self.cell[i] = EQCell.new(i)
        self.listView:pushBackCustomItem(self.cell[i])
        self.cell[i]:setTouchEnabled(true)
    end
end
```

```

        self.listView:addEventListener(
            function ( sender,event)
                print("*****")
                if event==1 then
                    dump(event)

self.selected_index=sender:getCurSelectedIndex()
                print(self.selected_index)
                self:refreshDes()

            end
        end)

    end
    self:refreshDes()
end

function MainScene:refreshDes()
    local item = self.listView:getItem(self.selected_index)
    local data = EQData[item.id]
    self.eqdes_icon:loadTexture("eq_"..item.id..".png")
    self.eqdes_kind:setString(data.kind)

    self.eqdes_name:setString(data.name)

```

页码: 6/7

www.neworigin.net

```

    self.lb_kind:setString(data.eq_kind)
    self.lb_lv:setString(tostring(data.lv))
    local lb_des = ""
    if data.add.health ~=0 then
        lb_des = lb_des.." + "..data.add.health.." 生命值\r\n"
    end
    if data.add.atk ~=0 then
        lb_des = lb_des.." + "..data.add.atk.." 攻击力\r\n"
    end
    if data.add.def ~=0 then
        lb_des = lb_des.." + "..data.add.def.." 防御力\r\n"
    end
    self.lb_des:setString(lb_des)
    for i=1,data.star do
        self.star[i]:loadTexture("star_on.png")
    end
    for i=data.star+1,5 do

```