

Python 游戏设计教程

编辑：唐宏

2024 年 3 月 14 日

目录

第一章 神奇的游戏循环：弹跳小球	5
1.1 实现小球反弹	5
1.2 实现 2 个小球反弹	6
1.3 通过列表实现多个小球反弹	7
1.4 把 Python 游戏打包成 exe 文件	8
1.4.1 基础环境配置	8
1.4.2 打包流程	8

第一章 神奇的游戏循环：弹跳小球

本章我们学习 Mu 编辑器的基本操作，借助 Pgzzero 库来创建游戏窗口，绘制背景及图像。

通过编程让小球移动，依靠游戏循环，实现小球反弹效果，让小球围绕窗口的四条边界来回弹跳。

1.1 实现小球反弹

```
WIDTH = 800
HEIGHT = 600
ball = Actor("alien", (200, 100))
ball.dx = 2
ball.dy = 2

def update():
    ball.x += ball.dx
    ball.y += ball.dy
    if ball.right > WIDTH or ball.left < 0:
        ball.dx = -ball.dx
    if ball.bottom > HEIGHT or ball.top < 0:
        ball.dy = -ball.dy

def draw():
    screen.fill((255, 255, 255))
    ball.draw()
```

1.2 实现 2 个小球反弹

```
WIDTH = 800
HEIGHT = 600
ball = Actor("alien", (200, 100))
ball.dx = 2
ball.dy = 2
ball2 = Actor("alien", (100, 200))
ball2.dx = 2
ball2.dy = 2

def update():
    ball.x += ball.dx
    ball.y += ball.dy
    if ball.right > WIDTH or ball.left < 0:
        ball.dx = -ball.dx
    if ball.bottom > HEIGHT or ball.top < 0:
        ball.dy = -ball.dy
    ball2.x += ball2.dx
    ball2.y += ball2.dy
    if ball2.right > WIDTH or ball2.left < 0:
        ball2.dx = -ball2.dx
    if ball2.bottom > HEIGHT or ball2.top < 0:
        ball2.dy = -ball2.dy

def draw():
    screen.fill((255, 255, 255))
    ball.draw()
    ball2.draw()
```

随着小球数量的增加，代码会变得越来越长。上面两个小球角色除了名称和初始位置不同之外，其他的操作几乎是一模一样的。倘若要在程序中添加大量的小球，例如几十个甚至上百个小球，那么意味着相同的操作也要重复几十遍甚至上百遍，而且代码的长度将变得不可想象。

1.3 通过列表实现多个小球反弹

```
WIDTH = 800
HEIGHT = 600
NUM = 10
balls = []
for i in range(NUM):
    ball = Actor("alien")
    ball.x = 50 * i + 100
    ball.y = 100
    ball.dx = 2 + i
    ball.dy = 2 + i
    balls.append(ball)

def update():
    for ball in balls:
        ball.x += ball.dx
        ball.y += ball.dy
        if ball.right > WIDTH or ball.left < 0:
            ball.dx = -ball.dx
        if ball.bottom > HEIGHT or ball.top < 0:
            ball.dy = -ball.dy

def draw():
    screen.fill((255, 255, 255))
    for ball in balls:
        ball.draw()
```

1.4 把 Python 游戏打包成 exe 文件

1.4.1 基础环境配置

不要用商店安装的 python，在官网下载安装。首次打包 exe 程序需要以 5 步操作。

1. 将 Python 源程序开头加上 `import pgzrun`，结尾加上 `pgzrun.go()`。

```
import pgzrun
# Python 代码
...
pgzrun.go()
```

2. 检查命令行中能否运行 `py` 或者 `python`，`pip`，若不行则检查环境变量，`pip` 的环境变量。

```
C:\Users\finance\AppData\Local\Programs\Python\
Python312\Scripts
```

3. 通过 `pip` 命令安装游戏运行环境 `pgzero`，安装打包程序 `pyinstaller`。

```
pip install pgzero
pip install pyinstaller
```

4. 在 DOS 命令行中通过 `py game.py` 运行游戏。

1.4.2 打包流程

1. 首次编译，生成 `test.spec` 配置文件。

```
pyinstaller test.py --collect-all pgzero -F -w
```

2. 修改 `test.spec` 配置文件。

```
datas = [('images', 'images'), ('music', 'music')]
```


3. 删除打包生成的 build、dist 文件夹，重新编译 test.spec 配置文件。

```
pyinstaller test.spec
```

生成的 exe 文件在 dist 文件夹中。