

疯狂的摩托(一)





1. 探索新知

1.1

游戏框架

我们编写程序框架模板，设定一个新窗口

```
# 导入pgzero库
import pgzrun

# 设置窗口
WIDTH = 1500 # 窗口的宽度
HEIGHT = 800 # 窗口的高度

# 启动游戏，写在所有程序的最后
pgzrun.go()
```

- pgzrun 就是 pygame zero run
- 窗口大小设定为 1500 * 800



项目类型: PNG 文件
分辨率: 1500 x 800
大小: 27.2 KB

任务一新增代码:

```
# (新增) 导入Pgzero库
import pgzrun

# (新增) 设置窗口
WIDTH = 1500 # 窗口的宽度
HEIGHT = 800 # 窗口的高度

# (新增) 启动游戏
pgzrun.go()
```

1.2

背景滚动

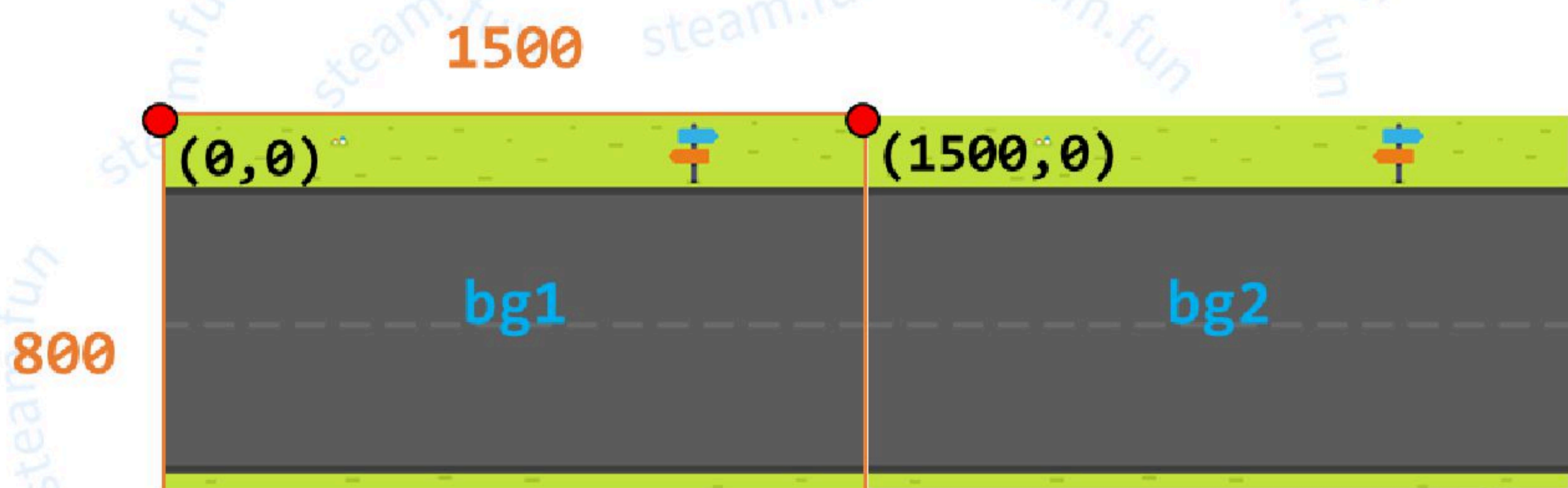
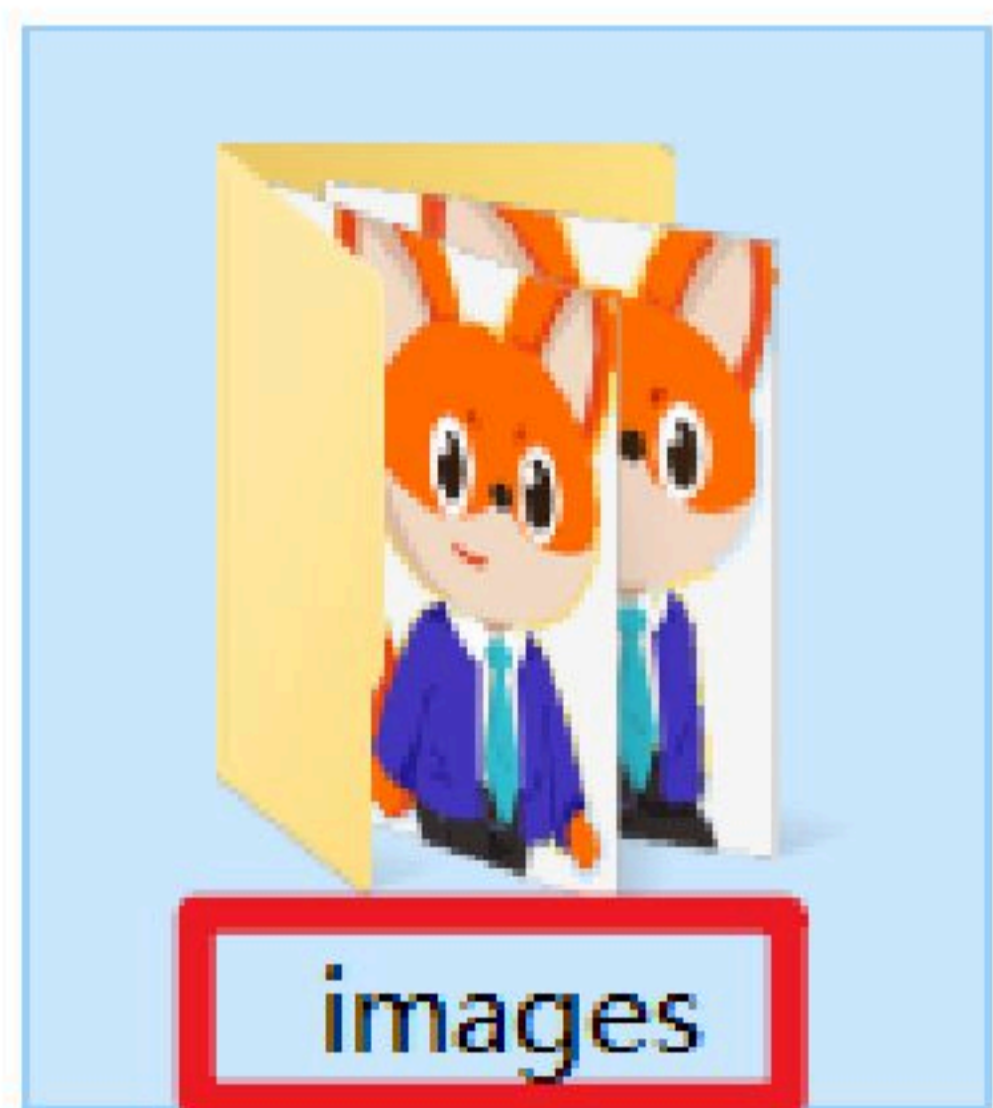
前面几节课，我们是如何绘制背景图像的呢？

- `screen.blit(name, pos)` 绘制静态背景时使用

要实现背景滚动，应该如何实现呢？

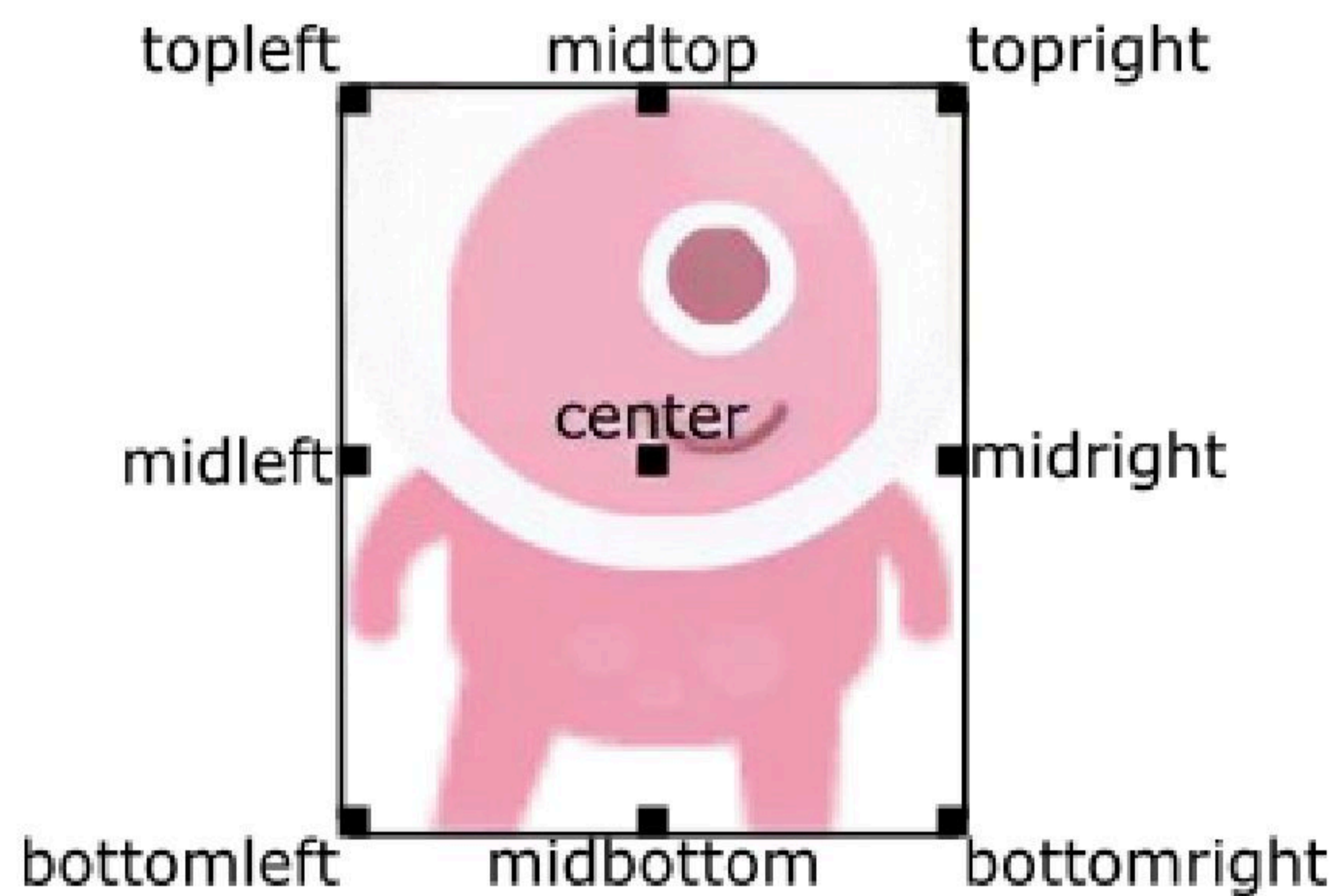
- `Actor(name, pos)` 当需要背景滚动时使用

只需要创建两个背景角色bg1、bg2，一个在舞台中间，一个在舞台右侧。让他们不断向左移动就可以了



Actor('图片名', center = (坐标))

我们可以使用关键字进行参数传递，可以指定角色以下 9 个点的坐标，以此设定角色所在位置，注意默认是center



创建背景角色

```
bg1 = Actor('连续道路', topleft = (0,0))
bg2 = Actor('连续道路', topleft = (1500,0))
def draw():
    # 绘制连续背景
    bg1.draw()
    bg2.draw()
```

1.2

背景滚动

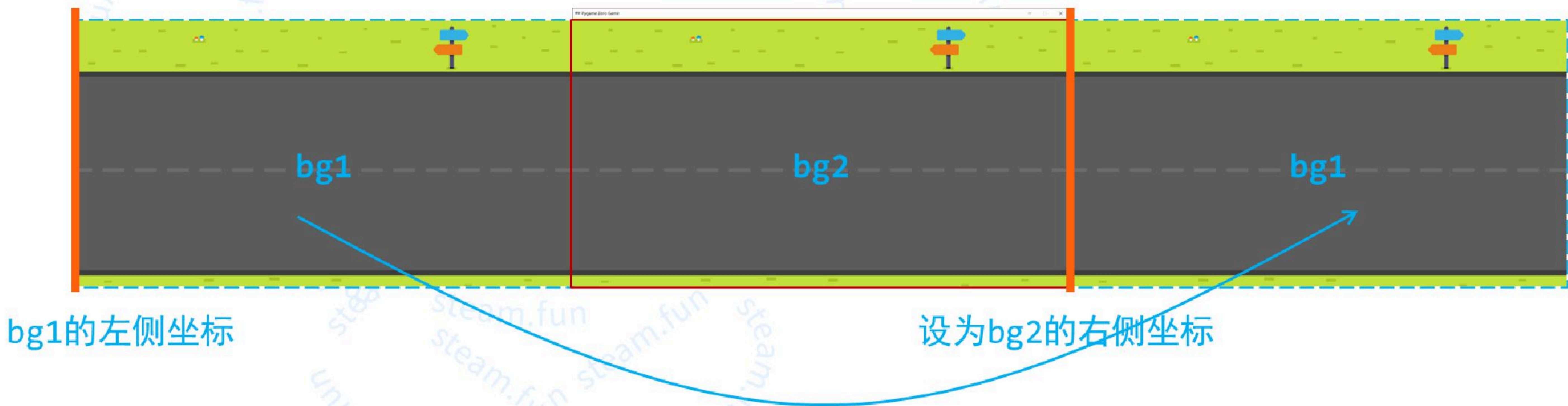
```
# 初始化设置
speed = 15

# 背景移动
def bg_move():
    bg1.x -= speed
    bg2.x -= speed
def update():
    bg_move()
```

游戏中，背景和障碍物等运动速度相同，利用相对运动的原理，营造出摩托前进的效果

将这个速度设定为变量 speed

背景图片移出屏幕左侧后，立马将其移动到舞台右侧，如此反复



```
def bg_move():
    ...
    if bg1.right <= 0:
        bg1.left = bg2.right
    if bg2.right <= 0:
        bg2.left = bg1.right
```

任务二新增代码:

```
import pgzrun
WIDTH = 1500 # 窗口的宽度
HEIGHT = 800 # 窗口的高度

# (新增) 创建背景角色
bg1 = Actor('连续道路', topleft = (0,0))
bg2 = Actor('连续道路', topleft = (1500,0))

# (新增) 初始化设置
speed = 15

# (新增) 绘制
def draw():
    bg1.draw()
    bg2.draw()

# (新增) 背景移动
def bg_move():
    bg1.x -= speed
    bg2.x -= speed
    if bg1.right <= 0:
        bg1.left = bg2.right
    if bg2.right <= 0:
        bg2.left = bg1.right

# (新增) 更新
def update():
    bg_move()

# 启动游戏
pgzrun.go()
```

1.3

障碍移动

首先创建障碍物

这里我创建3个障碍物，在道路区域随机位置出现



```
import random
...
# 创建背景角色
...
# 创建障碍物
obstacles = []
for _ in range(3):
    x = random.randint(WIDTH, WIDTH + 1500)
    y = random.randint(100, HEIGHT - 50)
    obstacles.append(Actor('路障', (x, y)))
```

接着绘制障碍物

```
def draw():
    # 绘制连续背景
    bg1.draw()
    bg2.draw()
    # 绘制障碍物
    for obs in obstacles:
        obs.draw()
```

思考：

运行后为什么看不到障碍物呢？

最后让障碍物随背景同步移动，移出屏幕后，再从右侧进入

```
# 更新障碍物位置
def obs_move():
    for obs in obstacles:
        obs.x -= speed # 每个障碍物都随背景一起移动
        if obs.x < 0:
            obs.x = random.randint(WIDTH, WIDTH + 1500)
            obs.y = random.randint(100, HEIGHT - 50)
def update():
    bg_move()
    obs_move()
```

任务三新增代码：

```
import pgzrun
import random # (新增)
...
# (新增) 创建障碍物
obstacles = []
for _ in range(3):
    x = random.randint(WIDTH, WIDTH + 1500)
    y = random.randint(100, HEIGHT - 50)
    obstacles.append(Actor('路障', (x, y)))

# 初始化设置
speed = 15

def draw():
    ...
    # (新增) 绘制障碍物
    for obs in obstacles:
        obs.draw()
# 背景移动
...
```

1.3

障碍移动

```

# (新增) 更新障碍物位置
def obs_move():
    for obs in obstacles:
        obs.x -= speed # 每个障碍物都随背景一起移动
        if obs.x < 0:
            obs.x = random.randint(WIDTH, WIDTH + 1500)
            obs.y = random.randint(100, HEIGHT - 50)

def update():
    bg_move()
    obs_move() # (新增)

pgzrun.go()

```

1.4

摩托移动

创建摩托角色，确定初始位置

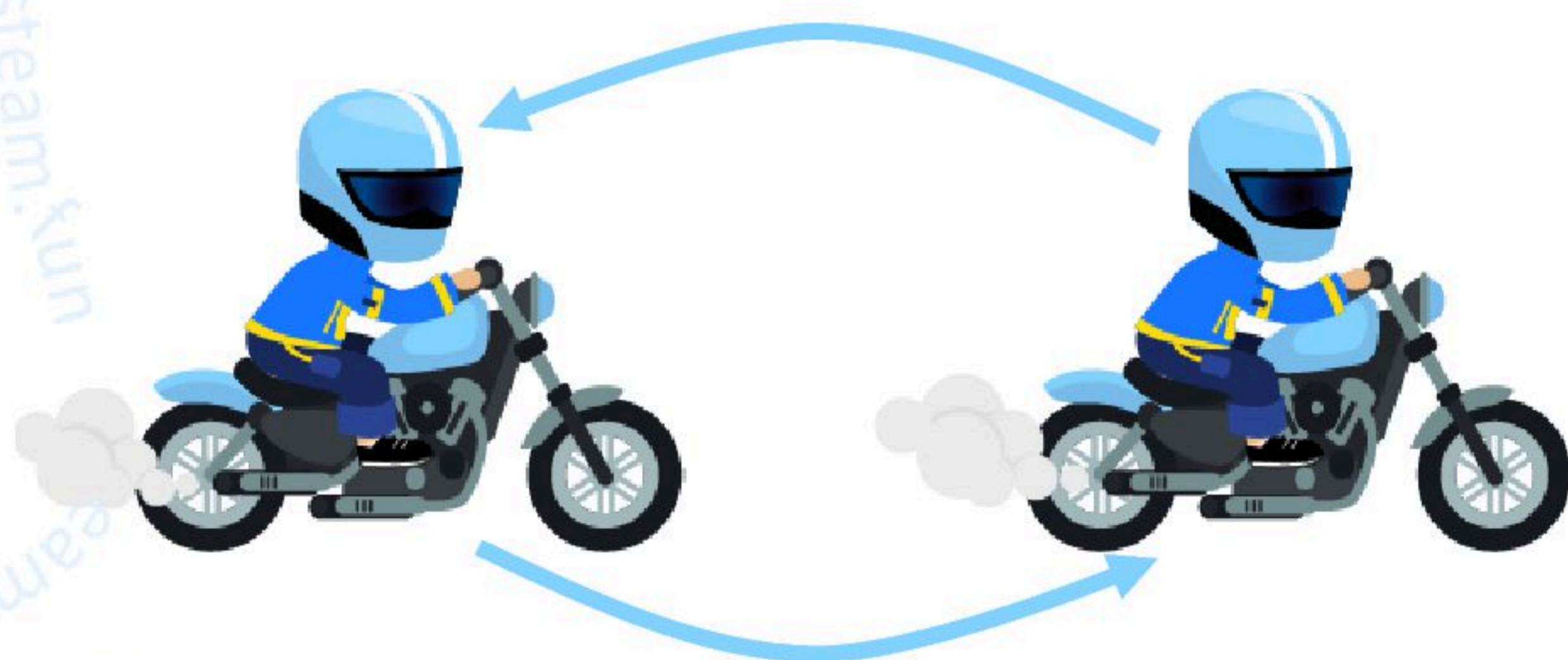
```

# 创建摩托车
motor = Actor('摩托1', (200, HEIGHT // 2))
def draw():
    # 绘制摩托
    motor.draw()

```

思考：

如何制作摩托的动态效果？



1.4

摩托移动

借助pgzhelper库，实现切换造型效果

```
# 导入pgzhelper库
from pgzhelper import *

motor = Actor('摩托1',(200,HEIGHT//2))
# 将摩托造型存放在列表内
motor.images = ['摩托1','摩托2']

# 更新函数
def update():
    ...
    motor.next_image()
```


指令手册

Actor.images

- 设置角色列表，存储actor的多张图片
- 设置后，角色造型默认呈现列表内第一个

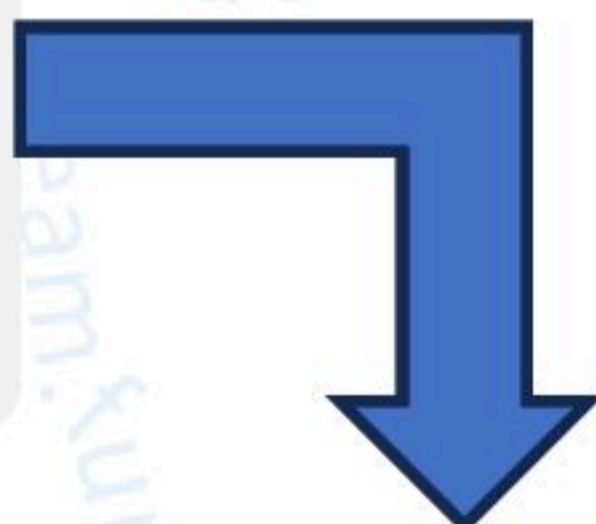
Actor.next_image()

- 将角色切换到列表内下一个造型



如何设定摩托切换造型的频率？

```
def update():
    ...
    motor.next_image()
```

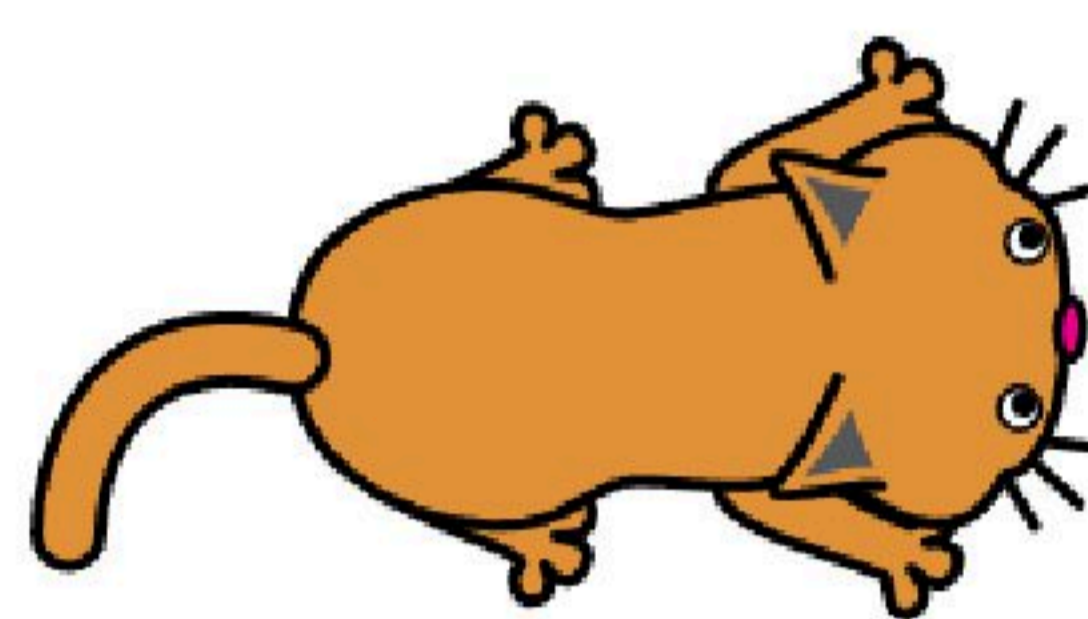


```
def change_image():
    motor.next_image()
clock.schedule_interval(change_image, 0.2)
```

如何用键盘控制角色移动？

先通过小猫，学习键盘上下左右键控制角色移动的方法

```
# 键盘上下左右键控制小猫移动
def update():
    if keyboard.left:
        cat.x -= 10
    elif keyboard.right:
        cat.x += 10
    elif keyboard.up:
        cat.y -= 10
    elif keyboard.down:
        cat.y += 10
```



控制摩托上下移动，躲避障碍物

思考：如何限制摩托的移动范围？

```
def update():
    bg_move()
    obs_move()
    # 控制赛车移动
    if keyboard.up and motor.top > 0:
        motor.y -= 7
    if keyboard.down and motor.bottom < HEIGHT-50:
        motor.y += 7
```

任务四新增代码：

```
...
# (新增) 创建摩托车
motor = Actor('摩托1', (200, HEIGHT // 2))
# (新增) 将摩托造型存放在列表内
motor.images = ['摩托1', '摩托2']

speed = 15

def draw():
    # 绘制连续背景
    ...
    # 绘制障碍物
    ...
    # (新增) 绘制摩托
    motor.draw()
...

# 更新
def update():
    bg_move()
    obs_move()
    # (新增) 控制赛车移动, 限制不能出界
    if keyboard.up and motor.top > 0:
        motor.y -= 7
    if keyboard.down and motor.bottom < HEIGHT-50:
        motor.y += 7

# (新增) 切换造型
def change_image():
    motor.next_image()
clock.schedule_interval(change_image, 0.2)
...
```

完整代码

```
import pgzrun
import random
from pgzhelper import *
# 游戏窗口大小
WIDTH = 1500
HEIGHT = 800

# 创建背景角色
bg1 = Actor('连续道路', topleft = (0,0))
bg2 = Actor('连续道路', topleft = (1500,0))
# 创建障碍物
obstacles = []
for _ in range(3): # 障碍物数量
    x = random.randint(WIDTH, WIDTH + 1500) # 障碍物出现在屏幕右侧外
    y = random.randint(100, HEIGHT - 50)
    obstacles.append(Actor('路障', (x, y)))
# 创建摩托车
motor = Actor('摩托1', (200, HEIGHT // 2))
# 将摩托造型存放在列表内
motor.images = ['摩托1', '摩托2']
# 初始化设置
speed = 15

def draw():
    # 绘制连续背景
    bg1.draw()
    bg2.draw()
    # 绘制障碍物
    for obs in obstacles:
        obs.draw()
    # 绘制摩托
    motor.draw()

# 背景移动
def bg_move():
    bg1.x -= speed
    bg2.x -= speed
    if bg1.right <= 0:
        bg1.left = bg2.right
    if bg2.right <= 0:
        bg2.left = bg1.right
```

完整代码

```
def obs_move(): # 更新障碍物位置
    for obs in obstacles:
        obs.x -= speed # 障碍物随背景一起移动
        if obs.x < 0: # 障碍物继续出现在屏幕外
            obs.x = random.randint(WIDTH, WIDTH + 1500)
            obs.y = random.randint(100, HEIGHT - 50)

def update():
    bg_move()
    obs_move()
    # 控制赛车移动, 限制不能出界
    if keyboard.up and motor.top > 0:
        motor.y -= 7
    if keyboard.down and motor.bottom < HEIGHT-50:
        motor.y += 7

def change_image():
    motor.next_image()
    clock.schedule_interval(change_image, 0.2)

pgzrun.go()
```



2. 强化练习

1. 已知赛车手角色左上角的坐标，请选出能正确设定角色的代码。()

- A. Actor('赛车手', (100, 300))
- B. Actor('赛车手', bottomleft = (100, 300))
- C. Actor('赛车手', topleft = (100, 300))
- D. Actor('赛车手', center = (100, 300))



2. 实现背景滚动效果，bg1向左移出舞台后，出现在舞台的右侧。选出相应代码()

A.

```
if bg1.left <= 0:  
    bg1.left = bg2.right
```

B.

```
if bg1.left >= 0:  
    bg1.left = bg2.right
```

C.

```
if bg1.right <= 0:  
    bg1.left = bg2.right
```

D.

```
if bg1.right >= 0:  
    bg1.left = bg2.right
```

3. 下面哪个选项，背景滚动速度最快？()

A. `bg1.x -= 10`

B. `bg1.x -= 15`

C. `bg1.x -= 20`

D. `bg1.x -= 25`



2. 强化练习

4. 下面哪个指令可以用于检测是否按下右键？（ ）

- A. keyboard.left
- B. keyboard.right
- C. keyboard.up
- D. keyboard.down

5. 下面哪个指令可以切换角色造型？（ ）

- A. motor.next_image()
- B. motor.images
- C. motor += 1
- C. motor += 1

3.术语箱

keyboard

键盘

motorcycle

摩托车

obstacles

障碍物

speed

速度

4.课后挑战

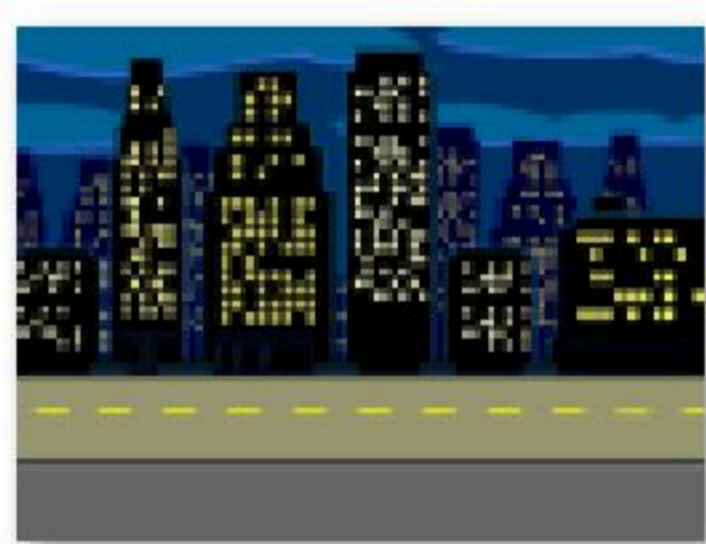
小城漫步

要求如下：

1. ‘白日街景’、‘夜晚街景’两图滚动出现
2. ‘走1’、‘走2’两图，实现博克斯走路动效
3. 按下键盘右键，背景滚动



白日街景.png



夜晚街景.png



走1.png



走2.png