# Module **Objects**

▶ EXPAND SOURCE CODE

# **Functions**

def load image(name, colorkey=None)

▶ EXPAND SOURCE CODE

# Classes

```
class Mob (*group)
```

simple base class for visible game objects

pygame.sprite.Sprite(\*groups): return Sprite

The base class for visible game objects. Derived classes will want to override the Sprite.update() method and assign Sprite.image and Sprite.rect attributes. The initializer can accept any number of Group instances that the Sprite will become a member of.

When subclassing the Sprite class, be sure to call the base initializer before adding the Sprite to Groups.

▶ EXPAND SOURCE CODE

### **Ancestors**

pygame.sprite.Sprite

# Class variables

```
var image_run
var image_run1
```

Methods
def again(self)
<pre>def check_fall(self)</pre>
<pre>def fall(self, hero, shoting, pos)</pre>
<pre>def get_coords(self)</pre>
def move(self)

▶ EXPAND SOURCE CODE

```
def sound(self)
```

▶ EXPAND SOURCE CODE

```
def update(self)
```

method to control sprite behavior

Sprite.update(args, \*kwargs):

The default implementation of this method does nothing; it's just a convenient "hook" that you can override. This method is called by Group.update() with whatever arguments you give it.

There is no need to use this method if not using the convenience method by the same name in the Group class.

► EXPAND SOURCE CODE

```
class MobBonus (x, y, *groups)
```

simple base class for visible game objects

pygame.sprite.Sprite(\*groups): return Sprite

The base class for visible game objects. Derived classes will want to override the Sprite.update() method and assign Sprite.image and Sprite.rect attributes. The initializer can accept any number of Group instances that the Sprite will become a member of.

When subclassing the Sprite class, be sure to call the base initializer before adding the Sprite to Groups.

▶ EXPAND SOURCE CODE

#### **Ancestors**

pygame.sprite.Sprite

#### Methods

def again(self)

def check\_fall(self)

def fall(self, hero, shoting, \*args)

def get\_coords(self)

def move(self)

def sound(self)

► EXPAND SOURCE CODE

► EXPAND SOURCE CODE

► EXPAND SOURCE CODE

► EXPAND SOURCE CODE

▶ EXPAND SOURCE CODE

► EXPAND SOURCE CODE

def update(self)

method to control sprite behavior

Sprite.update(args, \*kwargs):

The default implementation of this method does nothing; it's just a convenient "hook" that you can override. This method is called by Group.update() with whatever arguments you give it.

There is no need to use this method if not using the convenience method by the same name in the Group class.

▶ EXPAND SOURCE CODE

```
class MobGumba (x, y, *groups)
```

simple base class for visible game objects

pygame.sprite.Sprite(\*groups): return Sprite

The base class for visible game objects. Derived classes will want to override the Sprite.update() method and assign Sprite.image and Sprite.rect attributes. The initializer can accept any number of Group instances that the Sprite will become a member of.

When subclassing the Sprite class, be sure to call the base initializer before adding the Sprite to Groups.

► EXPAND SOURCE CODE

#### **Ancestors**

pygame.sprite.Sprite

#### Methods

method to control sprite behavior

Sprite.update(args, \*kwargs):

The default implementation of this method does nothing; it's just a convenient "hook" that you can override. This method is called by Group.update() with whatever arguments you give it.

There is no need to use this method if not using the convenience method by the same name in the Group class.

```
class MobMushroom (x, y, *groups)
```

simple base class for visible game objects

pygame.sprite.Sprite(\*groups): return Sprite

The base class for visible game objects. Derived classes will want to override the Sprite.update() method and assign Sprite.image and Sprite.rect attributes. The initializer can accept any number of Group instances that the Sprite will become a member of.

When subclassing the Sprite class, be sure to call the base initializer before adding the Sprite to Groups.

▶ EXPAND SOURCE CODE

#### **Ancestors**

pygame.sprite.Sprite

#### Class variables

```
var image_run
 var kill
Methods
 def again(self)
                                                                             ▶ EXPAND SOURCE CODE
 def check_fall(self)
                                                                             ▶ EXPAND SOURCE CODE
 def fall(self, hero, shoting, pos)
                                                                             ▶ EXPAND SOURCE CODE
 def get_coords(self)
                                                                             ► EXPAND SOURCE CODE
 def move(self)
                                                                             ► EXPAND SOURCE CODE
 def sound(self)
                                                                             ▶ EXPAND SOURCE CODE
 def update(self)
```

method to control sprite behavior

Sprite.update(args, \*kwargs):

The default implementation of this method does nothing; it's just a convenient "hook" that you can override. This method is called by Group.update() with whatever arguments you give it.

There is no need to use this method if not using the convenience method by the same name in the Group class.

# Index

# **Functions**

load\_image

# Classes

# Mob

again

check\_fall

fall

get\_coords

image\_run

image\_run1

move

sound

update

#### **MobBonus**

again

check\_fall

fall

get\_coords

move

sound

update

# **MobGumba**

again

check\_fall

fall

get\_coords

move

sound

update

# **MobMushroom**

again

check\_fall

fall

get\_coords

image\_run

kill

move

sound

update

Generated by pdoc 0.10.0.