

Module Mark

► [EXPAND SOURCE CODE](#)

Functions

```
def load_image(name, colorkey=None)
```

► [EXPAND SOURCE CODE](#)

Classes

```
class Mario (x, y, *gr)
```

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_damgel
```

```
var image_damgel1
```

```
var image_damger
```

```
var image_damger1
```

```
var image_dethl
```

```
var image_dethr
```

```
var image_jump_l
```

```
var image_jump_r
```

```
var image_run1_l
```

```
var image_run1_r
```

```
var image_run2_l
```

```
var image_run2_r
```

```
var image_start
```

```
var image_stay_l
```

```
var image_stay_r
```

Methods

```
def check_fall(self, mob)
```

► [EXPAND SOURCE CODE](#)

```
def damage_mario(self)
```

► [EXPAND SOURCE CODE](#)

```
def get_coords(self)
```

► [EXPAND SOURCE CODE](#)

```
def get_dash(self)
```

► [EXPAND SOURCE CODE](#)

```
def if_kill(self)
```

► [EXPAND SOURCE CODE](#)

```
def move_x(self, x)
```

► [EXPAND SOURCE CODE](#)

```
def return_shot(self)
```

► [EXPAND SOURCE CODE](#)

```
def set_group(self, gr)
```

► [EXPAND SOURCE CODE](#)

```
def set_lives(self, live)
```

► [EXPAND SOURCE CODE](#)

```
def set_moving(self)
```

► [EXPAND SOURCE CODE](#)

```
def set_walls(self, gr)
```

► [EXPAND SOURCE CODE](#)

```
def start_jump(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](#)

```
def update_lives(self)
```

► [EXPAND SOURCE CODE](#)

Index

Functions

`load_image`

Classes

Mario

`check_fall`
`damage_mario`
`get_coords`
`get_dash`
`if_kill`
`image_damgel`
`image_damgel1`
`image_damger`
`image_damger1`
`image_dethl`
`image_dethr`
`image_jump_l`
`image_jump_r`
`image_run1_l`
`image_run1_r`
`image_run2_l`
`image_run2_r`
`image_start`
`image_stay_l`
`image_stay_r`
`move_x`
`return_shot`
`set_group`
`set_lives`
`set_moving`
`set_walls`
`start_jump`
`update`
`update_lives`

