

Sprite Bonus · Visual Polish

Five tiny pygame programs that swap plain shapes for cartoon sprites. Use these to make your Galaxy Defender look like a real arcade game.

How sprites work in pygame

A "sprite" is just a **shape made of smaller shapes**. Pygame doesn't load picture files in Trinket easily, but we can compose any cartoon character out of `pygame.draw.polygon`, `pygame.draw.circle`, and `pygame.draw.ellipse`. We wrap each character in a tiny helper function so the game loop stays tidy.

Each program below introduces ONE sprite helper. Copy the code into the Trinket editor on the matching lesson page (or any pygame Trinket), press **Run**, then play with the colours and numbers.

Sprite 1 · The Cartoon Ship (intro in Lesson 5)

What it does

Replace the green rectangle from earlier lessons with a green spaceship - hull (polygon), cyan cockpit (circle), and an orange engine flame underneath. Steer with LEFT / RIGHT.

```
import pygame

pygame.init()
WIDTH, HEIGHT = 800, 600
screen = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption("Sprite Ship - LEFT/RIGHT to fly")
clock = pygame.time.Clock()

SPACE = (0, 0, 50)

def draw_ship(s, x, y):
    pygame.draw.polygon(s, (0, 230, 100),
        [(x+30,y),(x+60,y+40),(x+40,y+30),(x+20,y+30),(x,y+40)]) # hull
    pygame.draw.circle(s, (100, 200, 255), (x+30, y+18), 6) # cockpit
    pygame.draw.polygon(s, (255, 100, 0),
        [(x+25,y+30),(x+35,y+30),(x+30,y+42)]) # flame

player = pygame.Rect(370, 520, 60, 40)

while True:
    for e in pygame.event.get():
        if e.type == pygame.QUIT: pygame.quit(); exit()
    keys = pygame.key.get_pressed()
    if keys[pygame.K_LEFT] and player.x > 0: player.x -= 5
    if keys[pygame.K_RIGHT] and player.x < WIDTH - 60: player.x += 5

    screen.fill(SPACE)
    draw_ship(screen, player.x, player.y)
    pygame.display.flip()
    clock.tick(60)
```

Try these one by one

Ship 1 Change the hull to red (255, 80, 80).

Ship 2 Make the cockpit yellow.

```
pygame.draw.circle(s, (255, 220, 0), (x+30, y+18), 6)
```

Ship 3 Add a second flame for a wider exhaust.

Sprite 2 · Fiery Lasers (intro in Lesson 6)

What it does

Press SPACE - lasers fire upward. Each one is two stacked ellipses: an orange glow around a yellow core. They look like real arcade plasma bolts.

```
import pygame

pygame.init()
WIDTH, HEIGHT = 800, 600
screen = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption("Lasers - SPACE to fire")
clock = pygame.time.Clock()

SPACE = (0, 0, 50)

def draw_ship(s, x, y):
    pygame.draw.polygon(s, (0, 230, 100),
        [(x+30,y), (x+60,y+40), (x+40,y+30), (x+20,y+30), (x,y+40)])
    pygame.draw.circle(s, (100, 200, 255), (x+30, y+18), 6)
    pygame.draw.polygon(s, (255, 100, 0), [(x+25,y+30), (x+35,y+30), (x+30,y+42)])

def draw_laser(s, b):
    pygame.draw.ellipse(s, (255, 100, 0), (b.x-3, b.y-5, b.width+6, b.height+10))
    pygame.draw.ellipse(s, (255, 220, 0), b)

player = pygame.Rect(370, 520, 60, 40)
bullets = []

while True:
    for e in pygame.event.get():
        if e.type == pygame.QUIT: pygame.quit(); exit()
        if e.type == pygame.KEYDOWN and e.key == pygame.K_SPACE:
            bullets.append(pygame.Rect(player.x + 27, player.y, 6, 15))
    keys = pygame.key.get_pressed()
    if keys[pygame.K_LEFT] and player.x > 0: player.x -= 5
    if keys[pygame.K_RIGHT] and player.x < WIDTH - 60: player.x += 5
    for b in bullets[:]:
        b.y -= 10
        if b.y < 0: bullets.remove(b)

    screen.fill(SPACE)
    draw_ship(screen, player.x, player.y)
    for b in bullets: draw_laser(screen, b)
    pygame.display.flip()
    clock.tick(60)
```

Try these one by one

Laser 1 Make lasers BLUE - swap orange for (50, 130, 230) and yellow for (180, 220, 255).

Laser 2 Bigger lasers: spawn bullets that are 10 wide instead of 6.

```
bullets.append(pygame.Rect(player.x + 25, player.y, 10, 18))
```

Laser 3 Twice as fast: b.y -= 20.

Sprite 3 · Cartoon Aliens (intro in Lesson 7)

What it does

Five purple aliens drift from above. Each one is a small dome (head) on a big dome (body), with two white eyes and black pupils. Way more menacing than a red square.

```
import pygame
import random

pygame.init()
WIDTH, HEIGHT = 800, 600
screen = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption("Alien Field")
clock = pygame.time.Clock()

SPACE = (0, 0, 50)

def draw_alien(s, e):
    cx, cy = e.x + 25, e.y + 25
    pygame.draw.ellipse(s, (200, 50, 200), (e.x, e.y+10, 50, 30))
    pygame.draw.ellipse(s, (200, 50, 200), (e.x+10, e.y, 30, 20))
    pygame.draw.circle(s, (255,255,255), (cx-7, cy-8), 5)
    pygame.draw.circle(s, (255,255,255), (cx+7, cy-8), 5)
    pygame.draw.circle(s, (0, 0, 0), (cx-7, cy-8), 2)
    pygame.draw.circle(s, (0, 0, 0), (cx+7, cy-8), 2)

enemies = [pygame.Rect(random.randint(0, WIDTH-50),
                        random.randint(-300, 0), 50, 50) for _ in range(5)]

while True:
    for ev in pygame.event.get():
        if ev.type == pygame.QUIT: pygame.quit(); exit()
    for e in enemies:
        e.y += 2
        if e.y > HEIGHT:
            e.y, e.x = random.randint(-200, -50), random.randint(0, WIDTH-50)
    screen.fill(SPACE)
    for e in enemies: draw_alien(screen, e)
    pygame.display.flip()
    clock.tick(60)
```

Try these one by one

- Alien 1** Green aliens - swap (200, 50, 200) for (50, 220, 100).
- Alien 2** Angry red eyes - change the pupil colour from (0,0,0) to (255, 0, 0).
- Alien 3** Spawn 15 aliens for a serious wave.

```
enemies = [pygame.Rect(... ) for _ in range(15)]
```

Sprite 4 · Starfield Backdrop (intro in Lesson 8)

What it does

Generate 80 random white dots once at startup. Paint them every frame. Now the space-blue background looks like real outer space.

```
import pygame
import random

pygame.init()
WIDTH, HEIGHT = 800, 600
screen = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption("Starfield Backdrop")
clock = pygame.time.Clock()

SPACE = (0, 0, 50)

# generate once, paint every frame
stars = [(random.randint(0, WIDTH), random.randint(0, HEIGHT),
            random.randint(1, 2)) for _ in range(80)]
def draw_stars(s):
    for sx, sy, sz in stars:
        pygame.draw.circle(s, (255, 255, 255), (sx, sy), sz)

while True:
    for e in pygame.event.get():
        if e.type == pygame.QUIT: pygame.quit(); exit()
    screen.fill(SPACE)
    draw_stars(screen)
    pygame.display.flip()
    clock.tick(60)
```

Try these one by one

- Stars 1** More stars: range(200) for a richer sky.
- Stars 2** Some stars yellow, some white: pick (random.choice([(255,255,255), (255, 220, 100)])).
- Stars 3** Bigger stars: randint(2, 4) for radius.

Sprite 5 · All Sprites Together

The full combo - starry background, cartoon ship, fiery lasers, alien wave. A complete arcade-looking shooter in ~60 lines. This is what your Galaxy Defender looks like once all four sprite helpers are in.

Part 1 of 2 - the four sprite helpers:

```
import pygame
import random

pygame.init()
WIDTH, HEIGHT = 800, 600
screen = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption("All Sprites Together")
clock = pygame.time.Clock()
SPACE = (0, 0, 50)

def draw_ship(s, x, y):
    pygame.draw.polygon(s, (0, 230, 100),
        [(x+30,y), (x+60,y+40), (x+40,y+30), (x+20,y+30), (x,y+40)])
    pygame.draw.circle(s, (100, 200, 255), (x+30, y+18), 6)
    pygame.draw.polygon(s, (255, 100, 0),
        [(x+25,y+30), (x+35,y+30), (x+30,y+42)])

def draw_laser(s, b):
    pygame.draw.ellipse(s, (255, 100, 0),
        (b.x-3, b.y-5, b.width+6, b.height+10))
    pygame.draw.ellipse(s, (255, 220, 0), b)

def draw_alien(s, e):
    cx, cy = e.x + 25, e.y + 25
    pygame.draw.ellipse(s, (200, 50, 200), (e.x, e.y+10, 50, 30))
    pygame.draw.ellipse(s, (200, 50, 200), (e.x+10, e.y, 30, 20))
    for dx in (-7, 7):
        pygame.draw.circle(s, (255,255,255), (cx+dx, cy-8), 5)
        pygame.draw.circle(s, (0,0,0), (cx+dx, cy-8), 2)

stars = [(random.randint(0, WIDTH), random.randint(0, HEIGHT),
    random.randint(1, 2)) for _ in range(80)]
def draw_stars(s):
    for sx, sy, sz in stars:
        pygame.draw.circle(s, (255, 255, 255), (sx, sy), sz)

# ■■ the main loop continues on the next page ■■
```

Sprite 5 · All Sprites Together (continued)

Part 2 of 2 - the main loop using all four helpers:

```
# ■■ ...continued from the previous page ■■

player = pygame.Rect(370, 520, 60, 40)
bullets = []
enemies = [pygame.Rect(random.randint(0, WIDTH-50),
                        random.randint(-300, 0), 50, 50) for _ in range(5)]

while True:
    for e in pygame.event.get():
        if e.type == pygame.QUIT: pygame.quit(); exit()
        if e.type == pygame.KEYDOWN and e.key == pygame.K_SPACE:
            bullets.append(pygame.Rect(player.x + 27, player.y, 6, 15))
    keys = pygame.key.get_pressed()
    if keys[pygame.K_LEFT] and player.x > 0: player.x -= 5
    if keys[pygame.K_RIGHT] and player.x < WIDTH - 60: player.x += 5
    for b in bullets[:]:
        b.y -= 10
        if b.y < 0: bullets.remove(b)
    for e in enemies:
        e.y += 2
        if e.y > HEIGHT:
            e.y, e.x = random.randint(-200, -50), random.randint(0, WIDTH-50)

    screen.fill(SPACE)
    draw_stars(screen)
    draw_ship(screen, player.x, player.y)
    for b in bullets: draw_laser(screen, b)
    for e in enemies: draw_alien(screen, e)
    pygame.display.flip()
    clock.tick(60)
```

Try these one by one

- All 1** Add collision detection from Lesson 7 - bullets remove aliens on contact.
- All 2** Add a score variable + draw_text helper from Lesson 7.
- All 3** Change EVERY sprite to your own colour palette. Make it YOUR arcade.

Where to plug these in

Each sprite helper drops straight into your main Galaxy Defender code (the cumulative game on the lesson pages). The lesson HTML pages from Lesson 5 onwards already use them - this PDF is the reference if you want to remix colours or design your own sprites.

Design your own sprite: pick a character (a robot? a turtle? a snowman?). Sketch it on paper as 3-4 stacked shapes. Then write a draw_yourname(s, x, y) helper using polygon / circle / ellipse. Drop it in.