

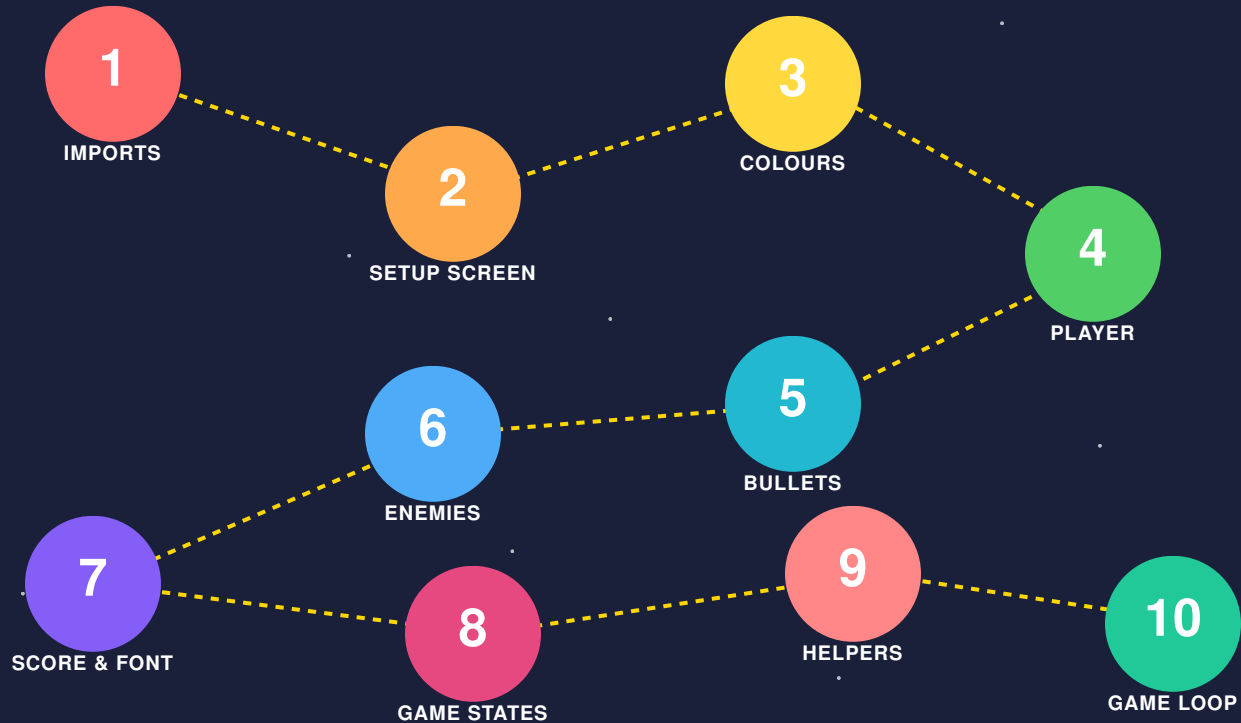
BUILD-A-GAME ROCKET

Every Python game has these 4 BIG parts



GALAXY MAP OF CODE

10 planets you visit to build Galaxy Defender



SECRET MISSION CODE — REMEMBER THIS!

**"I See Cool Pilots Blast Enemies,
Scoring Galactic Hero GOALS!"**

I
I

Imports

S
See

Setup Screen

C
Cool

Colours

P
Pilots

Player

B
Blast

Bullets

E
Enemies

Enemies

S
Scoring

Score & Font

G
Galactic

Game States

H
Hero

Helpers

G
GOALS

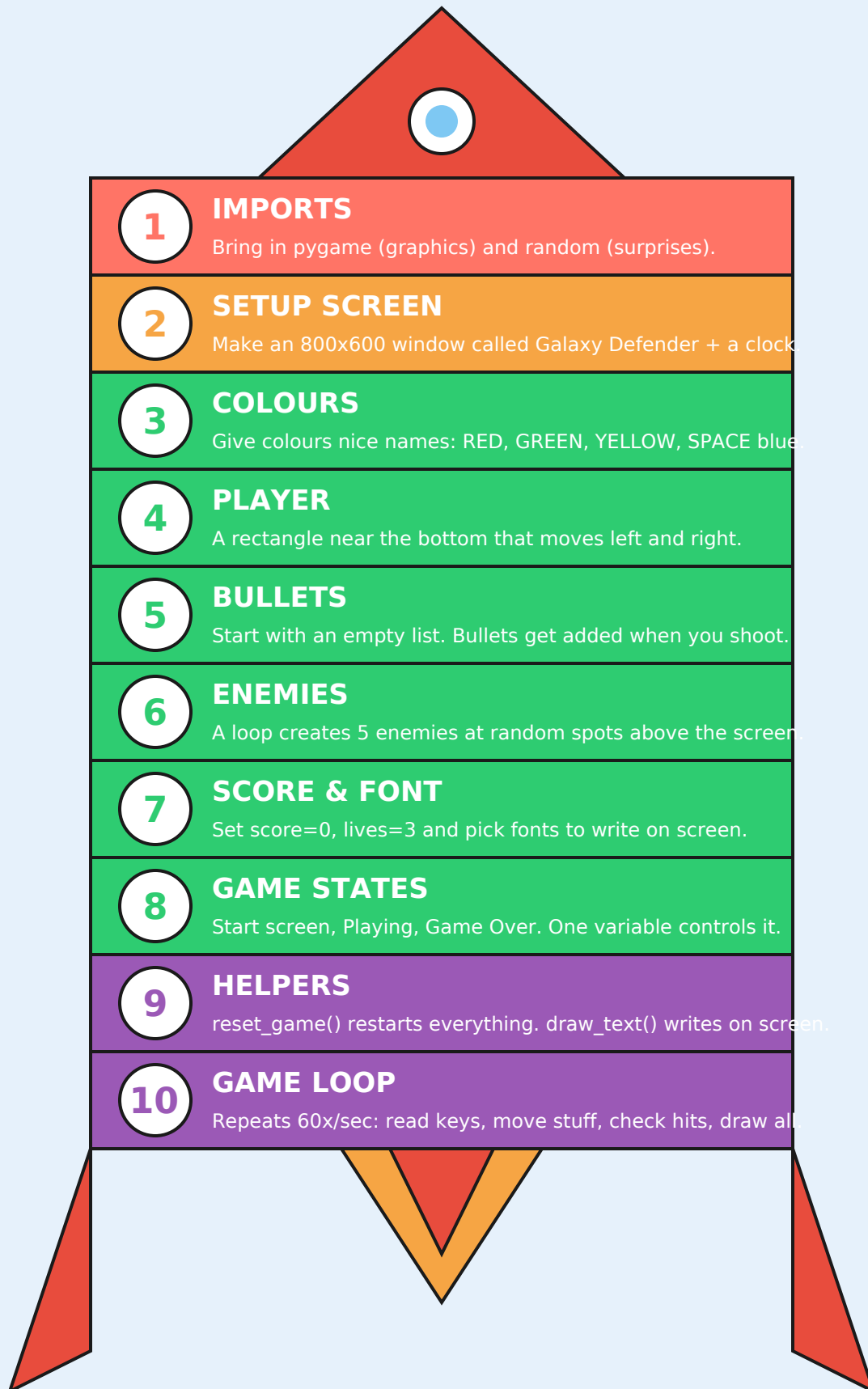
Game Loop

MISSION LOG:

- 1. **IMPORTS** Grab your toolbox
- 2. **SETUP SCREEN** Open the game window
- 3. **COLOURS** Mix your paint
- 4. **PLAYER** Build the spaceship
- 5. **BULLETS** Empty bullet box
- 6. **ENEMIES** Spawn 5 aliens
- 7. **SCORE & FONT** Keep score
- 8. **GAME STATES** Three modes
- 9. **HELPERS** Tiny tool functions
- 10. **GAME LOOP** The heartbeat

ROCKET LADDER

Stack 10 stages to launch Galaxy Defender



Import

Setup

THE BODY (stages 3-8)

THE END (stages 9-10)

Matches Big Picture (file 1)

CODE COMIC: 10 EPIC PANELS

The story of how Galaxy Defender comes alive

1 IMPORTS

"Grab your toolbox!"

Bring in pygame (for graphics) and random (for surprises).

2 SETUP SCREEN

"Open the game window!"

Make an 800x600 window called Galaxy Defender + a clock.

3 COLOURS

"Mix your paint!"

Give colours nice names: RED, GREEN, YELLOW, SPACE blue.

4 PLAYER

"Build the spaceship!"

A rectangle near the bottom that moves left and right.

5 BULLETS

"Empty bullet box!"

Start with an empty list. Bullets get added when you shoot.

6 ENEMIES

"Spawn 5 aliens!"

A loop creates 5 enemies at random spots above the screen.

7 SCORE & FONT

"Keep score!"

Set score=0, lives=3 and pick fonts to write on screen.

8 GAME STATES

"Three modes!"

Start screen, Playing, Game Over. One variable controls it.

9 HELPERS

"Tiny tool functions!"

reset_game() restarts everything. draw_text() writes on screen.

10 GAME LOOP

"The heartbeat!"

Repeats 60x/sec: read keys, move stuff, check hits, draw it all!

PIXEL CHEAT SHEET

All 10 parts + the exact code you type

1 IMPORTS

Grab your toolbox

```
import pygame
import random
```

2 SETUP SCREEN

Open the game window

```
screen = pygame.display.
    set_mode((800,600))
```

3 COLOURS

Mix your paint

```
RED = (255, 0, 0)
GREEN = (0, 255, 0)
```

4 PLAYER

Build the spaceship

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

5 BULLETS

Empty bullet box

```
bullets = []
bullet_speed = 10
```

6 ENEMIES

Spawn 5 aliens

```
for i in range(5):
    enemies.append(...)
```

7 SCORE & FONT

Keep score

```
score = 0
lives = 3
```

8 GAME STATES

Three modes

```
game_state = "start"
```

9 HELPERS

Tiny tool functions

```
def reset_game():
    ...
```

10 GAME LOOP

The heartbeat

```
while running:
    clock.tick(60)
```

10 CODE SUPERHEROES

Each part has its own power. Collect them all!

IMPORTS

#1

Power: Tool-Bringer



Bring in pygame (for graphics) and random (for surprises).

XP LEVEL 1/10

SETUP SCREEN

#2

Power: Window-Maker



Make an 800x600 window called Galaxy Defender + a clock.

XP LEVEL 2/10

COLOURS

#3

Power: Colour Wizard



Give colours nice names: RED, GREEN, YELLOW, SPACE blue.

XP LEVEL 3/10

PLAYER

#4

Power: Ship-Builder



A rectangle near the bottom that moves left and right.

XP LEVEL 4/10

BULLETS

#5

Power: Spark Launcher



Start with an empty list. Bullets get added when you shoot.

XP LEVEL 5/10

ENEMIES

#6

Power: Alien Spawner



A loop creates 5 enemies at random spots above the screen.

XP LEVEL 6/10

SCORE & FONT

#7

Power: Score Keeper



Set score=0, lives=3 and pick fonts to write on screen.

XP LEVEL 7/10

GAME STATES

#8

Power: Mode Switcher



Start screen, Playing, Game Over. One variable controls it.

XP LEVEL 8/10

HELPERS

#9

Power: Quick Fixer



reset_game() restarts everything. draw_text() writes on screen.

XP LEVEL 9/10

GAME LOOP

#10

Power: Heart-Beat



Repeats 60x/sec: read keys, move stuff, check hits, draw it all!

XP LEVEL 10/10

RECIPE: HOW TO COOK A GAME

Galaxy Defender | Serves: 1 happy coder | Time: ~1 hour

INGREDIENTS (set things up)

- | | | |
|---|---------------------|--|
| 1 | IMPORTS | - Bring in pygame (for graphics) and random (for surprises). |
| 2 | SETUP SCREEN | - Make an 800x600 window called Galaxy Defender + a clock. |
| 3 | COLOURS | - Give colours nice names: RED, GREEN, YELLOW, SPACE blue. |

METHOD (stir it together)

- | | | |
|---------|-------------------------|--|
| Step 4 | PLAYER | A rectangle near the bottom that moves left and right. |
| Step 5 | BULLETS | Start with an empty list. Bullets get added when you shoot. |
| Step 6 | ENEMIES | A loop creates 5 enemies at random spots above the screen. |
| Step 7 | SCORE & FONT | Set score=0, lives=3 and pick fonts to write on screen. |
| Step 8 | GAME STATES | Start screen, Playing, Game Over. One variable controls it. |
| Step 9 | HELPERS | reset_game() restarts everything. draw_text() writes on screen. |
| Step 10 | GAME LOOP | Repeats 60x/sec: read keys, move stuff, check hits, draw it all! |

Chef's tip: type the parts in order. Every recipe tastes better when you don't skip steps!

```
1  import turtle
2
3  # --- Set up the canvas ---
4  screen = turtle.Screen()
5  screen.bgcolor("black")
6  screen.title("Hello, Pixel World!")
7
8  # --- Create our turtle ---
9  t = turtle.Turtle()
10 t.color("cyan")
11 t.pensize(4)
12 t.speed(0)
13
14 # --- Draw a colourful square spiral ---
15 colours = ["cyan", "magenta", "yellow", "lime"]
16 for i in range(40):
17     t.color(colours[i % 4])
18     t.forward(i * 5)
19     t.right(90)
20
21 # --- Write your name ---
22 t.penup()
23 t.goto(0, -100)
24 t.color("white")
25 t.write("Coded by YOUR NAME", ...)
26
27 screen.mainloop()
```

- 1

Import the toolbox
Brings in Python's drawing tools. Without this line nothing draws — the turtle doesn't even exist yet.
- 2

Set up the canvas
Creates the window, paints the background, names the title bar. This is your blank piece of paper — size, colour, and title all set before you draw.
- 3

Create your turtle
Makes a turtle called t and sets its pen colour, line thickness, and drawing speed. speed(0) is the fastest setting — instant drawing.
- 4

Draw the spiral (the magic part)
A list of 4 colours + a for-loop that runs 40 times. Each loop: pick the next colour, draw a longer line, turn right 90°. Lines grow longer each time → the square becomes a colourful spiral instead.
- 5

Sign your art
penup() lifts the pen so the turtle can teleport without drawing. goto(0, -100) jumps to the spot below the spiral. write() drops your name on screen like a signature on a painting.
- 6

Keep the window open
Without this, the window vanishes the moment the drawing finishes. mainloop() says "stay open."

Teach this pattern: every Python drawing program has these 6 parts. Once kids see them once, they spot them in every program afterwards.