

## MAIN

```
import pygame
from pygame.locals import *
from configs import *
from Frog import *
from Soldier import *

pygame.init()

screen = pygame.display.set_mode([Window.WIDTH, Window.HEIGHT])
pygame.display.set_caption(Window.TITLE)
pygame.display.set_icon(Window.ICON)

Sound.AMBIENCE.play(loops = -1)
Sound.AMBIENCE.set_volume(0.4)

frog = Frog(30, 150, 5)
soldier1 = Soldier("top")
soldier2 = Soldier("bottom")

clock = pygame.time.Clock()

run = True
while run:
    dt = clock.tick(60)

    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            run = False
            break

    screen.blit(Img.BACKGROUND, [0, 0])

    key = pygame.key.get_pressed()
    if key[pygame.K_LEFT]:
        frog.move(Direction.LEFT)
    elif key[pygame.K_RIGHT]:
        frog.move(Direction.RIGHT)

    if key[pygame.K_UP]:
        frog.move(Direction.UP)
    elif key[pygame.K_DOWN]:
        frog.move(Direction.DOWN)

    frog.draw(screen)

    soldier1.move()
    soldier1.draw(screen)
    soldier2.move()
    soldier2.draw(screen)

    pygame.display.update()

pygame.quit()
```

## Configs.py

```
import pygame
pygame.init()

class Window:
    WIDTH = 552
    HEIGHT = 360
    MARGIN = {
        'top': 85,
        'bottom': 225,
        'left': 15,
        'right': 490
    }
    TITLE = "Frog Special Operation"
    ICON = pygame.image.load("imgs/frog.png")

class Img:
    BACKGROUND = pygame.image.load("imgs/background.png")
    FROG_RIGHT = pygame.image.load("imgs/frog.png")
    FROG_LEFT = pygame.transform.flip(FROG_RIGHT, True, False)
    FROG_UP = pygame.transform.rotate(FROG_RIGHT, 90)
    FROG_DOWN = pygame.transform.rotate(FROG_RIGHT, 270)
    SOLDIER = pygame.image.load("imgs/soldier.png")
    SOLDIER_TOP = pygame.transform.flip(SOLDIER, False, True)

class Sound:
    AMBIENCE = pygame.mixer.Sound("sound/BGSound.ogg")
```

## Direction.py

```
class Direction:
    UP = 0
    DOWN = 1
    LEFT = 2
    RIGHT = 3
```

## Soldier.py

```
import random
from configs import *
from Direction import *

class Soldier:
    def __init__(self, position):
        self.__x = 262
        self.__speed = 2

        if position == "top":
            self.__img = Img.SOLDIER_TOP
            self.__y = 5
            self.__direction = Direction.LEFT
        elif position == "bottom":
            self.__img = Img.SOLDIER
            self.__y = 300
            self.__direction = Direction.RIGHT

    def draw(self, surface):
        surface.blit(self.__img, [self.__x, self.__y])

    def bounce_boundaries(self):
        if self.__x < 10:
            self.__direction = Direction.RIGHT
        elif self.__x > Window.WIDTH - self.__img.get_width() - 10:
            self.__direction = Direction.LEFT

    def move(self):
        if self.__direction == Direction.LEFT:
            self.__x -= self.__speed
        elif self.__direction == Direction.RIGHT:
            self.__x += self.__speed
        elif self.__direction == Direction.UP:
            self.__y -= self.__speed
        elif self.__direction == Direction.DOWN:
            self.__y += self.__speed

        self.bounce_boundaries()
```

Frog.py

```
import pygame
from configs import *
from Direction import Direction

class Frog:
    def __init__(self, x, y, speed):
        self.__x = x
        self.__y = y
        self.__speed = speed
        self.__img = Img.FROG_RIGHT

    def draw(self, surface):
        surface.blit(self.__img, [self.__x, self.__y])

    def move(self, direction):
        if direction == Direction.LEFT:
            self.__x -= self.__speed
            self.__img = Img.FROG_LEFT
        elif direction == Direction.RIGHT:
            self.__x += self.__speed
        elif direction == Direction.UP:
            self.__y -= self.__speed
            self.__img = Img.FROG_UP
        elif direction == Direction.DOWN:
            self.__y += self.__speed
            self.__img = Img.FROG_DOWN

        self.limit_boundaries()

    def limit_boundaries(self):
        if self.__x < Window.MARGIN['left']:
            self.__x = Window.MARGIN['left']
        elif self.__x > Window.MARGIN['right']:
            self.__x = Window.MARGIN['right']
        elif self.__y < Window.MARGIN['top']:
            self.__y = Window.MARGIN['top']
        elif self.__y > Window.MARGIN['bottom']:
            self.__y = Window.MARGIN['bottom']
```