```python
import pygame as pg
import random
import sys


class Screen:
    def __init__(self, title, wh, img_path):
        pg.display.set_caption(title)
        self.sfc = pg.display.set_mode(wh)
        self.rct = self.sfc.get_rect()
        self.bgi_sfc = pg.image.load(img_path)
        self.bgi_rct = self.bgi_sfc.get_rect()

    def blit(self):
        self.sfc.blit(self.bgi_sfc, self.bgi_rct)


class Shot: #こうかとんの攻撃手段　（動かない）

    key_shot = {pg.K_SPACE}
    def __init__(self, color, rad, vxy, scr:Screen):
        self.sfc = pg.Surface((2*rad, 2*rad)) # 正方形の空のSurface
        self.sfc.set_colorkey((0, 0, 0))
        pg.draw.circle(self.sfc, (100,100,100), (rad, rad), rad) #(100,100,100)色の丸を描画
        self.rct = self.sfc.get_rect()
        self.rct.centerx = random.randint(0, scr.rct.width)
        self.rct.centery = random.randint(0, scr.rct.height)
        self.vx, self.vy = vxy

    def blit(self, scr:Screen):
        scr.sfc.blit(self.sfc, self.rct)

    def update(self, scr:Screen):
        self.rct.move_ip(self.vx, self.vy)
        yoko, tate = check_bound(self.rct, scr.rct)
        self.vx *= yoko
        self.vy *= tate
        self.blit(scr)

class Bird:
    key_delta = {
        pg.K_UP:    [0, -1],
        pg.K_DOWN:  [0, +1],
        pg.K_LEFT:  [-1, 0],
        pg.K_RIGHT: [+1, 0],
    }

    def __init__(self, img_path, ratio, xy):
        self.sfc = pg.image.load(img_path)
        self.sfc = pg.transform.rotozoom(self.sfc, 0, ratio)
        self.rct = self.sfc.get_rect()
        self.rct.center = xy

    def blit(self, scr:Screen):
        scr.sfc.blit(self.sfc, self.rct)
```

```python

    def update(self, scr:Screen):
        key_dct = pg.key.get_pressed()
        for key, delta in Bird.key_delta.items():
            if key_dct[key]:
                self.rct.centerx += delta[0]
                self.rct.centery += delta[1]
            if check_bound(self.rct, scr.rct) != (+1, +1):
                self.rct.centerx -= delta[0]
                self.rct.centery -= delta[1]
        self.blit(scr)


class Bomb:
    def __init__(self, color, rad, vxy, scr:Screen):
        self.sfc = pg.Surface((2*rad, 2*rad)) # 正方形の空のSurface
        self.sfc.set_colorkey((0, 0, 0))
        pg.draw.circle(self.sfc, color, (rad, rad), rad)
        self.rct = self.sfc.get_rect()
        self.rct.centerx = random.randint(0, scr.rct.width)
        self.rct.centery = random.randint(0, scr.rct.height)
        self.vx, self.vy = vxy

    def blit(self, scr:Screen):
        scr.sfc.blit(self.sfc, self.rct)

    def update(self, scr:Screen):
        self.rct.move_ip(self.vx, self.vy)
        yoko, tate = check_bound(self.rct, scr.rct)
        self.vx *= yoko
        self.vy *= tate
        self.blit(scr)


def check_bound(obj_rct, scr_rct):
    """
    第1引数：こうかとんrectまたは爆弾rect
    第2引数：スクリーンrect
    範囲内：+1／範囲外：-1
    """
    yoko, tate = +1, +1
    if obj_rct.left < scr_rct.left or scr_rct.right < obj_rct.right:
        yoko = -1
    if obj_rct.top < scr_rct.top or scr_rct.bottom < obj_rct.bottom:
        tate = -1
    return yoko, tate


def main():
    clock =pg.time.Clock()

    # 練習１
    scr = Screen("負けるな！こうかとん", (1600,900), "fig/pg_bg.jpg")

    # 練習３
    kkt = Bird("fig/6.png", 2.0, (900,400))
    kkt.update(scr)

    # 練習５
    bombs = []
    colors=["red", "green", "blue", "yellow", "magenta","greenyellow"] #爆弾を6個にする
    for i in range(6):
        color = colors[i]
        vx = random.choice([-1,+1])
        vy = random.choice([-1,+1])
        bombs.append(Bomb(color,10,(vx,vy,),scr))


    # 練習２

    while True:
        scr.blit()

        for event in pg.event.get():
```

```
130             if event.type == pg.QUIT:
131                 return
132
133
134
135
136         kkt.update(scr)
137         for bomb in bombs:
138             bomb.update(scr)
139             if kkt.rct.colliderect(bomb.rct):
140                 return
141
142         pg.display.update()
143         clock.tick(1000)
144
145
146 if __name__ == "__main__":
147     pg.init()
148     main()
149     pg.quit()
150     sys.exit()
```