

復興知2021 Processing-解答例

練習問題

1. 日本の国旗を描いてください

```
1 size(900, 600)
2 background(255)
3 noStroke()
4 fill(255, 0, 0)
5 ellipse(width / 2, height / 2, 350, 350)
```

2. オランダの国旗を描いてください

```
1 size(900, 600)
2 background(255)
3 noStroke()
4 fill(255, 0, 0)
5 rect(0, 0, width, height / 3)
6 fill(0, 0, 255)
7 rect(0, height * 2 / 3, width, height)
```

3. スイスの国旗を描いてください

```
1 size(600, 600)
2 background(255, 0, 0)
3 noStroke()
4 fill(255, 255, 255)
5 rect(width // 5, 2 * height // 5, 3 * width // 5, height // 5)
6 rect(2 * width // 5, height // 5, width // 5, 3 * height // 5)
```

アニメーションプログラム

- Step 0 | ウィンドウを設定

```
1 def setup():
2   size(800, 800) # ウィンドウのサイズ
3
4 def draw():
5   background(255) # 背景色を決める
```

- Step 1 | ハエ叩きをマウスの位置で変更

```
1 def setup():
2     size(800, 800) # ウィンドウのサイズ
3
4 def draw():
5     background(255)
6     sWidth = 30      # ハエ叩きの幅
7     sHeight = 30     # ハエ叩きの高さ
8     fill(198, 204, 255) # ハエ叩きの色
9     # ハエ叩きを描画する
10    rect(mouseX - sWidth / 2, mouseY - sHeight / 2, sWidth, sHeight)
```

- Step 2 | ハエを動かす

```
1 fX = 50 # ハエのx座標
2 fY = 50 # ハエのy座標
3 fVx = 9 # ハエのx方向に対するの更新量
4 fVy = 2 # ハエのy方向に対するの更新量
5
6 def setup():
7     size(800, 800) # ウィンドウのサイズ
8
9 def draw():
10    global fX, fY, fVx, fVy # グローバル変数を変更する
11
12    background(255)
13    sWidth = 30      # ハエ叩きの幅
14    sHeight = 30     # ハエ叩きの高さ
15    fill(198, 204, 255) # ハエ叩きの色
16    # ハエ叩きを描画する
17    rect(mouseX - sWidth / 2, mouseY - sHeight / 2, sWidth, sHeight)
18
19    fill(0, 0, 0)      # ハエの色
20    ellipse(fX, fY, 10, 10) # ハエを描画する
21    fX += fVx          # ハエのx座標を更新する
22    fY += fVy          # ハエのy座標を更新する
```

- Step 3 | ハエを壁で反射させる

```
1 fX = 50 # ハエのx座標
2 fY = 50 # ハエのy座標
3 fVx = 9 # ハエのx方向に対するの更新量
4 fVy = 2 # ハエのy方向に対するの更新量
5
```

```

6 def setup():
7     size(800, 800) # ウィンドウのサイズ
8
9 def draw():
10    global fX, fY, fVx, fVy # グローバル変数を変更する
11    background(255)
12    sWidth = 30           # ハエ叩きの幅
13    sHeight = 30         # ハエ叩きの高さ
14    fill(198, 204, 255) # ハエ叩きの色
15    # ハエ叩きを描画する
16    rect(mouseX - sWidth / 2, mouseY - sHeight / 2, sWidth, sHeight)
17    fill(0, 0, 0)        # ハエの色
18    ellipse(fX, fY, 10, 10) # ハエを描画する
19    fX += fVx             # ハエのx座標を更新する
20    fY += fVy             # ハエのy座標を更新する
21
22    if fY < 0 or fY > height: # 上下端衝突
23        fVy *= -1
24    if fX < 0 or fX > width: # 左右端衝突
25        fVx *= -1

```

- Step 4 | ハエ叩きとハエの衝突判定

```

1 fX = 50 # ハエのx座標
2 fY = 50 # ハエのy座標
3 fVx = 9 # ハエのx方向に対するの更新量
4 fVy = 2 # ハエのy方向に対するの更新量
5
6 def setup():
7     size(800, 800) # ウィンドウのサイズ
8
9 def draw():
10    global fX, fY, fVx, fVy, count # グローバル変数を変更する
11    background(255)
12    sWidth = 30           # ハエ叩きの幅
13    sHeight = 30         # ハエ叩きの高さ
14    fill(198, 204, 255) # ハエ叩きの色
15    # ハエ叩きを描画する
16    rect(mouseX - sWidth / 2, mouseY - sHeight / 2, sWidth, sHeight)
17    fill(0, 0, 0)        # ハエの色
18    ellipse(fX, fY, 10, 10) # ハエを描画する
19    fX += fVx             # ハエのx座標を更新する
20    fY += fVy             # ハエのy座標を更新する
21
22    if fY < 0 or fY > height: # 上下端衝突
23        fVy *= -1
24    if fX < 0 or fX > width: # 左右端衝突
25        fVx *= -1

```

```

26
27 sLeft = mouseX - sWidth / 2 # ハエ叩きの左端座標
28 sRight = mouseX + sWidth / 2 # ハエ叩きの右端座標
29 sTop = mouseY - sHeight / 2 # ハエ叩きの上端座標
30 sBottom = mouseY + sHeight / 2 # ハエ叩きの下端座標
31
32 # ハエ叩きとハエの衝突 (衝突後、ハエの位置と速度を変更する)
33 if sLeft <= fX and sRight >= fX and sTop <= fY and sBottom >= fY:
34     fX = int(random(width + 1))
35     fY = int(random(height + 1))
36     fVx = int(random(10)) + 1
37     fVy = int(random(10)) + 1

```

- Step 5 | 文字列によるインタラクション

```

1 fX = 50 # ハエのx座標
2 fY = 50 # ハエのy座標
3 fVx = 9 # ハエのx方向に対するの更新量
4 fVy = 2 # ハエのy方向に対するの更新量
5 count = 0 # 衝突回数
6
7 def setup():
8     size(800, 800) # ウィンドウのサイズ
9
10 def draw():
11     global fX, fY, fVx, fVy, count # グローバル変数を変更する
12     background(255)
13     sWidth = 30 # ハエ叩きの幅
14     sHeight = 30 # ハエ叩きの高さ
15     fill(198, 204, 255) # ハエ叩きの色
16     # ハエ叩きを描画する
17     rect(mouseX - sWidth / 2, mouseY - sHeight / 2, sWidth, sHeight)
18     fill(0, 0, 0) # ハエの色
19     ellipse(fX, fY, 10, 10) # ハエを描画する
20     fX += fVx # ハエのx座標を更新する
21     fY += fVy # ハエのy座標を更新する
22
23     if fY < 0 or fY > height: # 上下端衝突
24         fVy *= -1
25     if fX < 0 or fX > width: # 左右端衝突
26         fVx *= -1
27
28     sLeft = mouseX - sWidth / 2 # ハエ叩きの左端座標
29     sRight = mouseX + sWidth / 2 # ハエ叩きの右端座標
30     sTop = mouseY - sHeight / 2 # ハエ叩きの上端座標
31     sBottom = mouseY + sHeight / 2 # ハエ叩きの下端座標
32
33     # ハエ叩きとハエの衝突 (衝突後、ハエの位置と速度を変更する)
34     if sLeft <= fX and sRight >= fX and sTop <= fY and sBottom >= fY:

```

```
35     fX = int(random(width + 1))
36     fY = int(random(height + 1))
37     fVx = int(random(10)) + 1
38     fVy = int(random(10)) + 1
39     count += 1 # 衝突数を増やす
40
41     # 衝突数を表示する
42     textSize(16)
43     text(str(count), 20, 20) # 文字列にキャストする
```