

파이썬 거북이, 빙글빙글 돌아가는 육각형 꽃 만들기! 🐢🌸🎨 (리스트 설명 보강)

등장인물:

- **슬기 선생님:** 친절하고 유머러스한 코딩 선생님
- **지혜:** 똑똑하고 질문이 많은 학생
- **코딩이:** 코딩을 처음 배우지만 호기심 많은 학생

슬기 선생님: 여러분, 안녕하세요! 이중 for문으로 멋진 별 잔치를 열어봤죠? 오늘은 그 이중 for문을 이용해서 더욱 아름다운 패턴을 만들어 볼 거예요. 바로 여러 가지 색깔의 정육각형들이 빙글빙글 돌아가며 예쁜 꽃 모양을 만드는 마법입니다!

코딩이: 정육각형이요? 변이 6개인 도형 맞죠? 그게 막 돌아가면서 꽃이 된다고요? 우와! king 거북이가 오늘은 정원사가 되려나 봐요!

슬기 선생님: (웃음) 맞아요, 코딩이 학생! king 거북이가 아름다운 육각형 꽃을 피울 거예요. 먼저, 작업장을 준비해 볼까요? 이번엔 배경을 깜깜한 밤하늘처럼 검은색으로 해볼게요.

```
clearscreen()
```

```
screen = Screen()
```

```
screen.bgcolor(0, 0, 0) # RGB 값으로 검은색 배경! (빨강0, 초록0, 파랑0)
```

```
king = Turtle()
```

```
king.speed(10)          # king 거북이는 오늘도 빠르게!
```

```
king.pensize(10)        # 펜 굵기는 10으로 아주 두껍게!
```

지혜: screen.bgcolor(0, 0, 0) 이렇게 숫자로 배경색을 정할 수도 있군요! 저번에 배운 RGB 색상 맞죠? 세 숫자가 다 0이면 검은색이 되는 거네요!

슬기 선생님: 정확해요! 빛이 하나도 없는 상태가 검은색이니깐요. 자, 이제 이 깜깜한 밤하늘에 피어날 육각형 꽃잎의 색깔들을 미리 준비해 둘게요. 이번에도 리스트를 사용할 거예요.

```
colors = ["white", "green", "gray", "yellow", "blue", "red"] # 6가지 색깔 이름 준비!
```

코딩이: colors 리스트에 하얀색, 초록색, 회색, 노란색, 파란색, 빨간색 이름이 들어있어요! 선생님, 근데 어떤 친구가 색깔 정할 때 kitty.colors = [(240,255,255),(224,255,255),...] 이런 식으로 쓰거나, 아예 kitty.colors([(240,255,255),(224,255,255),...]) 이렇게 괄호 안에 바로 리스트를 넣기도 하던데요? 그렇게 해도 되는 거예요?

슬기 선생님: 아, 코딩이 학생! 정말 중요한 실제 사례를 가져왔네요! 많은 친구들이 헷갈릴 수 있는 부분이에요. 아주 잘 짚어줬어요! 먼저, 리스트에 RGB 색상 값(튜플)들을 담는 것 자체는 아주 좋은 생각이예요. 예를 들어, 이렇게 여러 RGB 색깔들을 리스트로 만들 수 있죠.

```
# 올바른 RGB 색상 리스트 만들기 예시
```

```
my_rgb_colors = [(240,255,255), (224,255,255), (175,238,238), (0,255,255), (32,178,170), (95,158,160)]
# 이 리스트에는 하늘색 계열의 RGB 튜플들이 6개 들어있어요.
```

이렇게 my_rgb_colors 라는 이름으로 리스트를 만들어 두면, 나중에 for 반복문에서 이렇게 사용할 수 있어요.
for one_rgb_color in my_rgb_colors: king.color(one_rgb_color) # ... 나머지 육각형 그리는 코드 ... 이렇게 하면 one_rgb_color 변수에 (240,255,255) 같은 튜플이 쏙 들어가고, king.color() 함수는 이 튜플을 받아서 해당 RGB 색으로 거북이 색을 바꿔준답니다.

자, 그럼 친구가 썼던 방식들을 살펴볼까요?

1. kitty.colors = [(240,255,255), ...] 이걸 kitty라는 거북이 객체에 colors라는 새로운 속성을 만들어서 거기에 리스트를 저장하려는 시도처럼 보여요. 파이썬에서는 이렇게 객체에 속성을 직접 만들 수도 있지만, 우리가 for문에서 색깔 목록을 순서대로 꺼내 쓰기 위한 일반적인 리스트를 만들 때는 보통 colors = [...] 나 my_rgb_colors = [...] 처럼 **별도의 변수에 리스트를 저장**하는 방식을 사용해요. 그게 코드를 읽고 이해하기 더 쉽고, 거북이 객체의 원래 기능과 섞이지 않아서 더 안전하답니다.
2. kitty.colors([(240,255,255), ...]) 이걸 kitty 거북이의 colors라는 기능을 괄호 () 안의 리스트를 가지고 호출하려는 것처럼 보여요. 하지만 거북이에게는 colors라는 이름의 기능(메서드)은 없어요. 색깔을 정하는 기능은 color() 이죠? 그리고 king.color() 함수는 현재 거북이의 선 색이나 채우기 색을 **설정**하는 역할을 하지, 여러 색깔의 목록을 저장해두었다가 for문에서 꺼내 쓰도록 하는 기능은 아닙니다. king.color("red") 처럼 색 이름을 주거나, king.color(255, 0, 0) 처럼 RGB 숫자를 직접 주거나, king.color((255,0,0)) 처럼 RGB 튜플을 주는 방식으로 **현재 색을 지정**하는 거죠.

그래서, 여러 색깔을 목록으로 만들어 사용하고 싶을 때는 오늘 우리가 한 것처럼 colors = ["white", "green", ...] 이렇게 문자열 리스트를 만들거나, 방금 예시로 보여준 my_rgb_colors = [(240,255,255), ...] 처럼 RGB 튜플 리스트를 만들어서, for 반복문으로 하나씩 꺼내 쓰는 것이 정확한 방법입니다!

지혜: 아하! 리스트는 colors = [...] 처럼 따로 만들고, for문에서 i in colors 이렇게 꺼내 쓰는 거군요! 그리고 RGB 값으로 리스트를 만들 수도 있고요! 친구가 썼던 방식은 조금 오해가 있었네요!

슬기 선생님: 맞아요. 처음 배울 때는 여러 가지를 시도해 보면서 헷갈릴 수 있어요. 중요한 건 왜 그렇게 되는지 원리를 이해하는 거랍니다! 자, 그럼 오늘은 우리가 준비한 색깔 이름 리스트 colors를 사용해서 이중 for문으로 육각형 꽃을 계속 피워볼까요?

```
for i in colors:      # 바깥쪽 for문: colors 리스트에서 색깔을 하나씩 꺼내 i에 담아요.
    king.color(i)      # 꺼내온 색깔 i로 king 거북이 변신!
    king.right(360/6)  # 다음 육각형을 그리기 전에 살짝 방향을 틀어요! (60도)
```

```
    for j in range(6): # 안쪽 for문: 정육각형의 변 6개를 그릴 거예요. (변수명을 j로 바꿔서 명확하게!)
        king.forward(100)
        king.right(360/6) # 정육각형의 한 각을 만들기 위해 방향을 틀어요! (60도)
```

(선생님이 안쪽 for문의 반복 변수를 j로 수정하여 설명한다)

슬기 선생님: 방금 제가 안쪽 for문의 반복 변수를 i에서 j로 살짝 바꿨어요. 이렇게 하면 바깥쪽 for문의

i(색깔 이름)와 안쪽 for문의 j(0부터 5까지의 숫자)가 서로 다른 이름을 갖게 돼서 우리가 코드를 읽을 때 덜 헷갈리겠죠? 내용은 똑같이 동작한답니다.

코딩이: 네! j로 하니까 더 알기 쉬운 것 같아요! 안쪽 for j in range(6): 이걸 육각형 하나를 그리는 거죠? 6번 반복하면서 앞으로 갔다가 꺾고! 그때도 king.right(360/6)으로 60도씩 꺾네요! 왜 똑같이 60도예요?

슬기 선생님: 아주 중요한 질문이에요! 정다각형(모든 변의 길이와 모든 각의 크기가 같은 다각형)을 그릴 때, 거북이가 꺾어야 하는 외각의 크기는 항상 $360 / \text{변의_개수}$ 랍니다. 정육각형은 변이 6개니까, 한 꼭짓점에서 다음 변으로 넘어갈 때 거북이는 $360 / 6 = 60$ 도 만큼 방향을 틀어야 반듯한 정육각형을 그릴 수 있어요. 그래서 안쪽 for문의 king.right(60)은 **하나의 정육각형을 완성하기 위해** 각 꼭짓점에서 꺾는 각도이고, 바깥쪽 for문의 king.color(i) 바로 다음에 있는 king.right(60)은 **하나의 육각형을 그린 후, 다음 색깔의 육각형을 그리기 전에 전체적인 그림의 구도를 잡기 위해** 거북이의 시작 방향을 트는 각도입니다. 이 두 번의 60도 회전 덕분에 육각형들이 서로 다른 방향으로 예쁘게 펼쳐지며 꽃 모양을 만들게 되는 거예요.

(king 거북이가 검은 배경 위로 다양한 색깔의 굵은 선으로 된 육각형들이 회전하며 겹쳐져 아름다운 꽃과 같은 패턴을 그린다)

지혜: 아! 안쪽 60도는 육각형 모양을 만들고, 바깥쪽 60도는 꽃잎처럼 펼쳐지게 하는 거군요! 그리고 색깔 리스트 만드는 법도 이제 확실히 알겠어요!

코딩이: 와! 진짜 예쁜 육각형 꽃이 만들어졌어요! 색깔도 계속 바뀌고, 방향도 빙글빙글 돌고! pen-size(10)으로 하니까 선도 엄청 굵어서 멋있어요!

슬기 선생님: 그렇죠? 오늘 우리는 이중 for문을 사용하고, 각도 계산을 통해 정육각형을 그리고, 그것들을 회전시키면서 아름다운 패턴을 만들어봤어요. 리스트에 색깔 정보를 담는 올바른 방법, 그리고 RGB 값으로도 색을 표현할 수 있다는 것도 다시 한번 확인했구요. 반복 변수 이름을 어떻게 지으면 좋을지에 대한 고민도 해봤네요!

지혜 & 코딩이: 네! 선생님! 오늘 정말 신기하고 재미있는 마법을 배운 것 같아요! 이제 저희도 멋진 패턴 아티스트가 될 수 있겠어요!