

파이썬 거북이와 정삼각형 그리기! 📐🐢

등장인물:

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

슬기 선생님: 자, 여러분! 지난 시간에 우리 baby 거북이로 선도 긋고, 방향도 바꿔봤죠? 오늘은 kitty라는 새 거북이 친구를 불러서 멋진 도형을 그려 볼 거예요. 바로 반듯반듯 정삼각형이랍니다!

코딩이: 와! 정삼각형이요? 세모 모양 맞죠? kitty는 고양이예요? 고양이 거북이인가? 똥굴똥굴~?

슬기 선생님: (웃음) 이름은 kitty라고 붙여봤지만, 역시 그림을 그려주는 특별한 거북이 친구예요. 자, 먼저 도화지를 깨끗하게 해볼까요? 이젠 이제 다들 알죠?

```
clearscreen()
```

지혜: 네! clearscreen() 하고 괄호 ()를 꼭 붙여서 “지금 바로 화면을 지워줘!” 라고 명령하는 거죠!

슬기 선생님: 훌륭해요! 이제 오늘의 주인공 kitty 거북이를 만들어 볼게요.

```
kitty = Turtle()
```

슬기 선생님: 여기서 Turtle()은 마치 ‘거북이 만들기 설계도’ 같은 거예요. 이 설계도를 ****클래스(Class)****라고 부르는데, 우리가 “사람”이라고 말하는 것과 비슷해요. “사람”은 키가 큰 사람, 작은 사람, 안경 쓴 사람 등 다양하지만, ‘두 팔, 두 다리, 머리 하나’ 같은 공통된 특징을 가진 추상적인 개념이죠?

지혜: 아! 그럼 Turtle 클래스는 ‘그림을 그릴 수 있는 거북이’라는 특징을 가진 설계도 같은 거네요!

슬기 선생님: 맞아요! 그리고 kitty = Turtle() 이렇게 명령하면, 그 Turtle 설계도대로 만들어진 진짜 거북이 한 마리가 탄생하는 거예요. 이 진짜 거북이, kitty를 **객체(Object)** 또는 ****인스턴스(Instance)****라고 불러요. “사람”이라는 클래스에서 “유재석”이라는 구체적인 한 사람이 있는 것처럼, Turtle이라는 클래스에서 kitty라는 구체적인 거북이 한 마리가 만들어진 거죠. baby도 Turtle()로 만들었으니 baby도 객체고요!

코딩이: 오오! 그럼 Turtle() 설계도로 doggy = Turtle(), hero = Turtle() 이렇게 만들면 doggy 거북이, hero 거북이도 생기는 거예요?

슬기 선생님: 정확해요, 코딩이 학생! 각자 다른 이름을 가진, 하지만 모두 Turtle 설계도로 만들어진 거북이 객체들이 생기는 거랍니다. 자, 이제 kitty 거북이에게 정삼각형을 그리도록 명령해 볼까요? 정삼각형은 세 변의 길이가 같고, 세 각의 크기도 같아요. 먼저 앞으로 쪽 가볼게요.

```
kitty.forward(200)
```

지혜: kitty라는 객체(우리 거북이)가 가지고 있는 forward라는 능력을 사용하는 거죠! 괄호 안의 200은 200만큼 앞으로 가라는 거고요! 이 forward() 같은 명령어를 **함수(Function)** 또는 ****메서드(Method)****라고 부른다고 하셨죠? 어떤 일을 하도록 미리 만들어 놓은 명령어 묶음이요!

슬기 선생님: 지혜 학생, 정말 잘 이해하고 있네요! `forward()`는 kitty 거북이가 앞으로 이동하도록 미리 만들어진 기능이에요. 이처럼 특정 객체(kitty)가 가지고 있는 함수를 특별히 **메서드**라고 부르기도 합니다. 자, 이제 한 변을 그렸으니 방향을 꺾어야겠죠? 정삼각형은 안쪽 각이 60도인데, 거북이는 바깥쪽으로 돌아야 하니까 몇 도를 돌아야 할까요?

코딩이: 음... 60도요!

슬기 선생님: 아하! 거북이가 보고 있는 방향에서 바깥쪽으로 꺾어야 다음 변을 그릴 수 있어요. 직선은 180도니까, 180도에서 안쪽 각 60도를 빼면?

지혜: 120도요!

슬기 선생님: 맞아요! 그래서 오른쪽으로 120도 돌 거예요.

```
kitty.right(120)
```

코딩이: `kitty.right(120)` ! 아, 선생님! 제가 만약에 `kitty.right` 하고 괄호랑 숫자를 안 쓰면 어떻게 돼요?

슬기 선생님: 좋은 질문이에요! `kitty.right` 까지만 쓰면, 컴퓨터는 “음, kitty의 오른쪽으로 도는 기능이 있다는 건 알겠는데... 그래서 뭘 어찌라는 거지? 몇 도를 돌아야 할지 모르겠어!” 하고 아무것도 하지 않거나 오류를 낼 수 있어요. 괄호 ()는 “이 기능을 지금 바로 실행해!” 라는 뜻이고, 괄호 안의 숫자 120은 “120도 만큼!” 이라고 구체적인 값을 알려주는 거예요. 그래서 `kitty.right(120)` 이라고 정확히 써줘야 합니다.

슬기 선생님: 자, 이제 두 번째 변을 그려볼까요?

```
kitty.forward(200)
```

(kitty 거북이가 120도 회전 후 200만큼 앞으로 이동한다)

지혜: 와! 정말 정삼각형 모양이 되어가고 있어요! 그럼 또 120도 돌고, 200만큼 가면 완성되겠네요!

슬기 선생님: 그렇죠! 마지막으로 한 번 더!

```
kitty.right(120)
```

```
kitty.forward(200)
```

(kitty 거북이가 다시 120도 회전 후 200만큼 앞으로 이동하여 정삼각형을 완성한다)

코딩이: 와! 예쁜 정삼각형이 만들어졌어요! 제가 만든 거예요! 선생님, 근데 만약에요, 제가 `kitty = Turtle()` 이렇게 잘 만들어놓고, 다음에 `kitty = forward(200)` 이렇게 쓰면 어떻게 돼요? 점 찍는 거 깜빡하고요.

슬기 선생님: 아주 중요한 질문이에요, 코딩이 학생! 만약 `kitty = forward(200)` 이라고 쓰면, 컴퓨터는 이렇게 생각해요. “`forward(200)`이라는 새로운 값을 kitty라는 이름표에 붙여야겠다!” 이렇게 되면 kitty는 더 이상 우리가 만든 거북이가 아니라, `forward(200)`이 실행된 결과 (아마도 아무것도 없다는 뜻의 None일 거예요)를 가리키게 돼요. 그래서 그 다음부터 `kitty.right(120)` 같은 명령을 내리면 “어? kitty는 거북이가 아닌데 어떻게 오른쪽으로 돌아?” 하면서 오류가 난답니다. 거북이 kitty의 능력을 사용할 때는 반드시 `kitty.forward(200)`처럼 점(.)을 찍어서 “kitty야, 네 능력 forward를 써줘!” 라고 알려줘야 해요.

지혜: 아! 그럼 만약에 `cat = Turtle()` 해놓고 거북이 모양을 바꾸고 싶어서 `cat.shape("turtle")` 이렇게 해야 하는데, 실수로 `cat = shape("turtle")` 이라고 하면, cat이 더 이상 거북이가 아니게 되는 거네요?

슬기 선생님: 정확해요! cat이라는 이름표가 거북이를 가리키고 있다가, shape("turtle")이라는 (아마도 존재하지 않는) 함수의 결과를 가리키게 되면서 거북이를 잃어버리는 거죠. 항상 객체 cat의 모양을 바꿀 거니까 cat.shape("turtle") 이라고 점을 찍어야 해요.

코딩이: 선생님! shape("turtle") 할 때 turtle에 왜 큰따옴표를 써야 해요? 그리고 대문자 Turtle이라고 쓰면 안 돼요?

슬기 선생님: 좋은 질문이 두 개나 나왔네요! 첫째, shape() 함수에게 “거북이 모양으로 바꿔줘!” 라고 알려줄 때, 'turtle'이라는 글자 자체를 전달해야 해요. 만약 따옴표 없이 shape(turtle)이라고 쓰면, 파이썬은 'turtle'이라는 이름의 변수(값을 담는 상자)를 찾으려고 할 거예요. 하지만 우리는 'turtle'이라는 글자, 즉 문자열을 전달하고 싶은 거니까 "turtle" 또는 'turtle' 이렇게 따옴표로 감싸줘야 “이건 글자 그대로의 turtle이야!”라고 파이썬이 알아들을 수 있답니다. 둘째, shape("Turtle")처럼 대문자 'T'를 쓰면 안 되는 이유는, 이 shape 함수가 알고 있는 거북이 모양의 이름이 소문자 'turtle'로 정해져 있기 때문이에요. 마치 우리가 친구 이름을 부를 때 정확한 이름을 불러야 알아듣는 것처럼요. shape 함수는 “classic”, “arrow”, “circle” 그리고 “turtle” 같은 몇 가지 정해진 모양 이름만 알고 있답니다.

슬기 선생님: 오늘 정삼각형도 그려보고, 클래스와 객체라는 중요한 개념도 배웠어요. 그리고 함수(메서드)를 사용할 때 괄호 ()와 점.을 왜 써야 하는지, 문자열을 쓸 때 왜 따옴표 ""가 필요한지도 알게 됐죠? 코딩 실수가 줄어들면 더 재미있게 그림을 그릴 수 있을 거예요!

지혜 & 코딩이: 네! 선생님! 이제 헛갈리지 않고 잘 할 수 있을 것 같아요!