< 커스텀 뷰 생성 방법 >

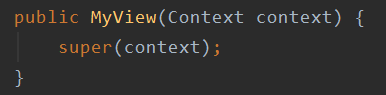
1. View 클래스를 상속

-> 화면에 붙여지는 요소들은 반드시 View클래스를 상속해야함

C:\Users\i\AppData\Local\Microsoft\Windows\INetCache\Content.Word\캡처.png

2. 부모클래스의 생성자 호출

-> 부모, 자식 생성자의 인자의 타입은 반드시 ***Context***타입이어야함

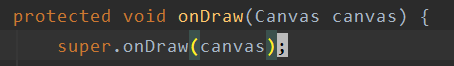


***※ Context란?*** 어떤 특정 요소가 실행되고 있을 당시의 모든 상황들을 의미

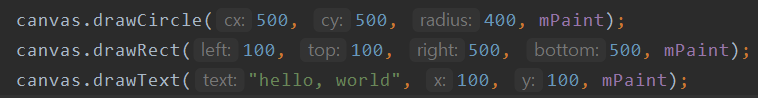
-> 뷰가 앱에 붙여질 때 무작정 붙여지는 것이 아니라 화면의 크기, 테마의 색과 같은 다양한 주변 정보들을 읽어들여서 적절히 초기화한 뒤 화면에 붙여짐. 이런 초기화 정보는 Activity객체에 들어있음. 이 Acitivity객체를 Context타입의 인자에 넣어주는 것임.(Activity클래스는 Context인터페이스를 구현하므로, Context타입의 인자에 넣어줄 수 있음)

3. onDraw메서드를 오버라이딩

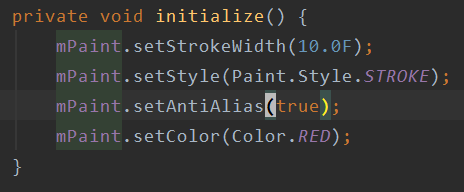
-> 화면에 그려지는 객체들은 모두 자신이 어떻게 그려져야 할 지를 알고 있음. 그리고 이에 대한 정보는 onDraw메서드에 정의되어 있음. onDraw메서드는 안드로이드에 의해 자동으로 콜백되므로 사용자가 직접 호출하는 경우는 거의 없음.

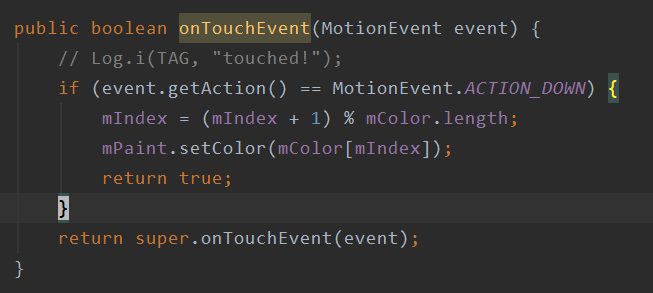


-> onDraw메서드에는 인자로 Canvas객체가 들어옴. Canvas객체에는 다양한 그리기 알고리즘이 제공됨.



-> Paint객체를 선언한 뒤 속성을 지정할 수 있음.





※ 위의 코드는 왜 동작하지 않는가?

위의 코드에서는 터치하면 색은 바뀌지만 바뀌어진 색을 다시 그려야 화면의 색이 바뀜. 즉 onDraw메서드를 다시 호출해야하는데 이 메서드는 사용자가 직접 호출할 수 없음. 따라서 안드로이드에게 다시 그려달라고 요청을 해야함. 그 요청 메서드가 invalidate라는 메서드임. 즉 invalidate메서드는 안드로이드에게 onDraw메서드를 다시 호출해달라고 요청하는 메서드임.

MyView3 예제 잘 봐라

커스텀 뷰를 화면에 붙이는 방법

1. 코드상에서 붙이는 방법

2. xml에서 붙이는 방법

< listView >

listView는 하나 이상의 itemView를 포함할 수 있음.

listView를 만들기 위해서는

1. listView

2. container

3. adapter

가 필요함

Adapter는 컨테이너에서 요소를 뽑아와서 아이템뷰로 만든뒤 리스트뷰에 하나씩 붙여넣음

View의 길이값의 단위는 dp

View에는 가중치라는 개념이 있음. 만약 화면의 해상도 300\*800이라고 가정해보자. 사용자가 버튼 3개를 배치시킨다고 생각해보자. 버튼은 100\*50이다. 근데 만약 단말기가 400\*800이라면 공간이 남음. 따라서 부모의 너비를 100이라보고 각 버튼에 33.34씩 가중치를 주는 방식으로 한다.

가중치를 적용하는 방법: 높이 혹은 넓이를 0dp를 준다. 그리고 weight=1을 준다.

0dp를 준다는 것은 길이가 0으로 하겠다는게 아니라 가중치를 주겠다는 것임.