



ANDROID STUDIO -BITMAP DRAW-

SOULSEEK

BITMAP DRAW

단말기 해상도 구하기

- 해상도를 구하고 그것을 기준으로 그림의 크기를 비율적으로 그려주는게 좋기 때문에 해상도를 먼저 알아야한다.

```
Display display = ((WindowManager) getSystemService(Context.WINDOW_SERVICE)).getDefaultDisplay();
```

```
Width = display.getWidth();
```

```
Height = display.getHeight();
```

그림 처리하기

- 해상도 기준으로 리소스를 **Bitmap**으로 읽어 들이고 그려준다.

```
Bitmap spaceship;
```

```
//리소스에 있는 그림을 BITMAP으로 만들어준다.
```

```
Spaceship = BitmapFactory.decodeResource(getResources(), R.drawable.spaceship);
```

```
int x = Width/8
```

```
int y = Height/11;
```

```
//이미지의 크기를 해상도 기준으로 맞춰준다.
```

```
Spaceship = Bitmap.createScaledBitmap(spaceship, x, y, true);
```

View클래스를 만들어서 **layout**대신 **View**클래스를 보여준다.

```
class MyView extends View{
```

```
    MyView(Context context){
```

```
        super(context); //부모 클래스의 생성자를 호출한다.
```

```
        .....
```

```
    }
```

```
    @Override
```

```
    public void onDraw(Canvas canvas){
```

```
        ..... //화면에 나타낼 그림이나 문자를 처리한다.
```

```
    }
```

```
    public Boolean onTouchEvent(MotionEvent event){
```

```
        ..... //화면을 터치했을 경우 처리하는 부분이다.
```

```
    }
```

BITMAP DRAW

Canvas 클래스 & Paint 클래스

- **drawBitmap()**, **drawText()**를 사용해서 그릴 수 있는 함수를 지원한다.
- **Paint** 도구를 이용해 그릴 때 특수한 효과를 준다.

```
Public void onDraw(Canvas canvas){
```

```
    Paint p1 = new Paint();  
    p1.setColor(Color.RED);  
    p1.setTextSize(50);
```

```
    canvas.drawText("hello", 0, 200, p1);  
    canvas.drawBitmap(spaceship, spaceshipX, spaceshipY, p1);
```

```
}
```

OnTouchEvent

```
public Boolean onTouchEvent(MotionEvent event){
```

```
    int x = 0, int y = 0;
```

```
    //화면을 터치하거나 움직이는 상태를 체크한다.
```

```
    if(event.getAction() == MotionEvent.ACTION_DOWN
```

```
        || event.getAction() == MotionEvent.ACTION_MOVE){
```

```
        //터치 했을때의 좌표를 알아온다.
```

```
        x = (int)event.getX();
```

```
        y = (int)event.getY();
```

```
    }
```

```
    //버튼을 터치하는 걸 판단해서 좌표 이동을 해준다.
```

```
    if((x > leftKeyX) && (x < leftKeyX + button_width) && (y > leftKeyY) && (y < leftKeyY + button_width)){
```

```
        spaceshipX -= 20;
```

```
    }
```

```
    return 0;
```

```
}
```

BITMAP DRAW

```
public class MainActivity extends AppCompatActivity {
```

```
    Bitmap screen; //배경 이미지  
    Bitmap spaceship; //우주선 이미지  
    Bitmap leftKey, rightKey; //왼쪽, 오른쪽 버튼 이미지  
    int spaceshipPosX, spaceshipPosY; //우주선 위치  
    int leftKeyPosX, leftKeyPosY; //왼쪽 버튼 위치  
    int rightKeyPosX, rightKeyPosY; //오른쪽 버튼 위치  
    int Width, Height; //해상도  
    int button_width; //왼쪽 버튼의 크기  
    int touchPosX, touchPosY; // 터치 위치
```

```
@Override
```

```
protected void onCreate(Bundle savedInstanceState) {
```

```
    super.onCreate(savedInstanceState);
```

```
    //직접 만든 activity를 보여준다.
```

```
    MyView myview = new MyView(this);
```

```
    setContentView(myview);
```

```
    //사용기기의 해상도를 알아온다.
```

```
    Display display
```

```
        = ((WindowManager) getSystemService(Context.WINDOW_SERVICE)).getDefaultDisplay();
```

```
    Width = display.getWidth();
```

```
    Height = display.getHeight();
```

```
    spaceship = BitmapFactory.decodeResource(getResources(), R.drawable.spaceship);
```

BITMAP DRAW

```
int x = Width/8;
int y = Height/11;
spaceship = Bitmap.createScaledBitmap(spaceship, x, y, true);

spaceshipPosX = Width * 1/9;
spaceshipPosY = Height * 6/9;

leftKey = BitmapFactory.decodeResource(getResources(), R.drawable.leftkey);

leftKeyPosX = Width * 5/9;
leftKeyPosY = Height * 7/9;
button_width = Width / 6;

leftKey = Bitmap.createScaledBitmap(leftKey, button_width, button_width, true);

rightKey = BitmapFactory.decodeResource(getResources(), R.drawable.rightkey);

rightKeyPosX = Width * 7/9;
rightKeyPosY = Height * 7/9;

rightKey = Bitmap.createScaledBitmap(rightKey, button_width, button_width, true);

screen = BitmapFactory.decodeResource(getResources(), R.drawable.screen);
screen = Bitmap.createScaledBitmap(screen, Width, Height, true);
}
```

BITMAP DRAW

```
class MyView extends View {  
    MyView(Context context){  
        super(context); //View클래스의 생성자를 호출  
        setBackgroundColor(Color.BLUE); //배경색을 푸른색으로 전환  
    }  
  
    @Override  
    public void onDraw(Canvas canvas){  
        Paint p1 = new Paint();  
        p1.setColor(Color.RED);  
        p1.setTextSize(50);  
  
        //게임배경을 그린다.  
        canvas.drawBitmap(screen, 0, 0, p1);  
        //우주선을 그린다.  
        canvas.drawBitmap(spaceship, spaceshipPosX, spaceshipPosY, p1);  
        //버튼을 그린다.  
        canvas.drawBitmap(leftKey, leftKeyPosX, leftKeyPosY, p1);  
        canvas.drawBitmap(rightKey, rightKeyPosX, rightKeyPosY, p1);  
    }  
}
```

BITMAP DRAW

@Override

public boolean onTouchEvent(MotionEvent event){

**if(event.getAction() == MotionEvent.ACTION_DOWN
|| event.getAction() == MotionEvent.ACTION_MOVE){**

**touchPosX = (int) event.getX();
touchPosY = (int) event.getY();**

}

//왼쪽키를 터치했을 경우

**if((touchPosX > leftKeyPosX) && (touchPosX < leftKeyPosX + button_width)
&& (touchPosY > leftKeyPosY) && (touchPosY < leftKeyPosY + button_width)){**

spaceshipPosX -= 20;

}

//오른쪽 키를 터치했을 경우

**if((touchPosX > rightKeyPosX) && (touchPosX < rightKeyPosX + button_width)
&& (touchPosY > rightKeyPosY) && (touchPosY < rightKeyPosY + button_width)){**

spaceshipPosX += 20;

}

invalidate();

return true; //제대로 처리 되었을 경우 true를 반환

}