

Android 2D

* إنشاء كلاس (class) جديد يمتد من View subclass في الكود
 * توضع الـ code في الكلاس الجديد (View) في ملف activity الخاص به

* إنشاء كلاس جديد Animation يمتد من الكلاس Thread في الكود
 * يتم تحديث الـ update في الكلاس الجديد، ويتم تحديث الكلاس.

الخطوات → 1) create a new Java class.

2) extend the ~~the~~ View class

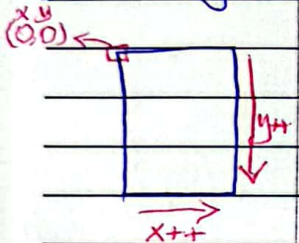
3) handle the error using the super constructor → (context, AttributeSet)

1) method

2) constrains

4) override the onDraw() method :
 it takes a Canvas Object.

الخطوات



5) we use the canvasObj to draw any shape canvasObj.draw() (~).

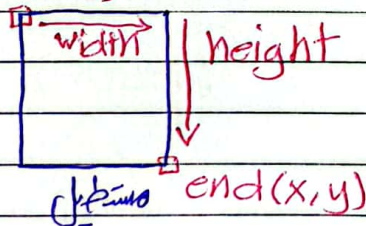
Shapes :

1) Rectangle Shape :

C.drawRect(left, top, right, bottom, Paint)
C.drawRect(new RectF(~), PaintObj)

الخطوات
 في الكلاس الجديد
 في الكلاس الجديد

start(x,y)



الخطوات في الكلاس الجديد
 (150, 200) كلاس

height = 500 و width = 300

: endpoint في الكلاس الجديد

$x = 150 + 300 = 450$

$y = 200 + 500 = 700$

في الكلاس الجديد

C.drawRect(150, 200, 450, 700,

start
 point

new Paint() ;

في الكلاس الجديد

end
 point

Android 2D

- * عشان نرسم الشكل في رساله ونضيف على ال layout جوا ال XML بنسا على tag زي هيك :

<PackageName, className />

- * Paint class : هو عبارة عن قلم رسم الشكل والاساسي فيه انه يكون كونه اسود ويحتوي fill للشكل كمان باللون الاسود

Paint P = new Paint();

- P.setColor (Color, "anyColor");

- P.setARGB (a, r, g, b); اللون (التي بيختر كلهم) 255
اللون الاسود كلهم 0

لا تنسى فوكه تفتح وتفتح الالوان وما في انت على فوكا .

- P.setStyle (Paint, style, ...); FILL
STROKE.

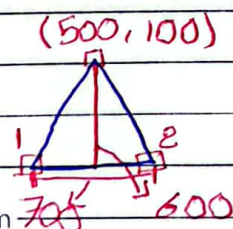
2) Circle shape:

C.drawCircle (x, y, radius, PaintObj);

3) Oval Shape:

C.drawOval (new RectF (L, T, R, B), PaintObj);
C.drawOval (Left, Top, Right, Bottom, PaintObj);

4) Triangle shape:



500 هو x
100 هو y
ارتفاع هو 600
طول القاعدة هو 700

الزاوية الاولى (x, y) :
x = 500 + 350 = 850
y = 100 + 600 = 700
الزاوية الثانية (x, y) :
x = 500 - 350 = 150
y = 100 + 600 = 700

Android 2D

- to draw a Triangle we need to use the path class. → `Canvas` class method `drawPath()`

1) create a path Obj

`Path pt = new Path();`

2) draw the head point using `moveTo()`

`pt.moveTo(500, 100);`

3) draw the two other point using `lineTo()`

`pt.lineTo(150, 700);`

`pt.lineTo(850, 700);`

4) to get back to the head point use `close()`

`pt.close();`

or just use the `close()` method

`pt.close();`

5) Line Shape:

`C.drawLine(startPointX, startPointY, endPointX, endPointY, Paintobj);`

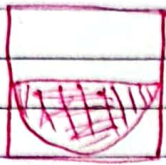
6) Point Shape:

`C.drawPoint(x, y, Paint)`

Android 2D

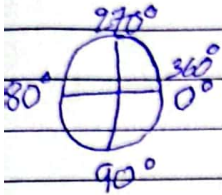
7) Arc shape:

عشان نرسم
قوس
بنا الشكل
من الأشياء



rectangle

RectF, startangle, Sweepangle
centerUsage, PaintObj.



- RectF → جوانه القوس عشان انا نرسمها
- StartAngle → زاوية بداية رسم القوس
- SweepAngle → حجم زاوية رسم القوس
- centerUsage → "true" في الدوران

من الأشياء
عشان نرسم
قوس
بنا الشكل
من الأشياء

8) Text shape

c.drawText("msg", x, y, PaintObj);

PaintObj.setTextSize(100);
~~PaintObj~~

- لتغيير حجم النص في text:

1) create a Typeface obj or use anonimous

2) use this method:

PaintObj.setTypeface(Typeface.create());

Typeface.FontFamily,

Typeface.Style → BOLD, ~~ITALIC~~ ITALIC

3}

Android 2D

9) Image shape: class 11

1) add any Image to drawable file

2) use the method drawBitmap()

C. drawBitmap(BitmapObj, X, y, PaintObj);

3) we need to create a BitmapObj to attach the image to it.

Bitmap B = BitmapFactory.decodeResource(getResources(), R.drawable.imagename);

Animation

1) عرف الـ Animation كلاس في مستوي الـ class.

2) الـ Animation كلاس في فئة من فئة الـ View كلاس في مستوي الـ class.

3) في فئة الـ Animation كلاس في فئة الـ View كلاس في مستوي الـ class. الـ Animation كلاس في فئة الـ View كلاس في مستوي الـ class. الـ Animation كلاس في فئة الـ View كلاس في مستوي الـ class.

invalidate(): method

الـ invalidate() method في فئة الـ View كلاس في مستوي الـ class. الـ invalidate() method في فئة الـ View كلاس في مستوي الـ class. الـ invalidate() method في فئة الـ View كلاس في مستوي الـ class.

Thread.sleep()

الـ Thread.sleep() method في فئة الـ Thread كلاس في مستوي الـ class. الـ Thread.sleep() method في فئة الـ Thread كلاس في مستوي الـ class. الـ Thread.sleep() method في فئة الـ Thread كلاس في مستوي الـ class.

exception في الـ Thread.sleep() method. InterruptedExp. call

Android 2D Animation

یہ سب سے پہلی صورت ہے جس میں ایک `Canvas` کے ساتھ ایک `View` کی شکل دیکھی جاتی ہے۔

یہ دونوں `Canvas` کے ساتھ ہیں۔
`C.getWidth()`؛
`C.getHeight()`؛

Android 2D Animation control

یہ سب سے پہلی صورت ہے جس میں ایک `Canvas` کے ساتھ ایک `View` کی شکل دیکھی جاتی ہے۔

1) control animation with Buttons:

1) create a `LinearLayout` above your drawing Panel.

Buttons (1)

2) create your Buttons with (id).

Keyboard (2)

3) create an `actionListener` in the `MainActivity` class to each Button.

~~Buttons (3)~~

4) to edit the shape direction we need to create a method `setDir()`, or you can set the direction var as a static var and change it by the class name.

یہ سب سے پہلی صورت ہے جس میں ایک `Canvas` کے ساتھ ایک `View` کی شکل دیکھی جاتی ہے۔

2) control animation with Keyboard:

1) override the `onKeyDown()` method

یہ سب سے پہلی صورت ہے جس میں ایک `Canvas` کے ساتھ ایک `View` کی شکل دیکھی جاتی ہے۔

Android ED Animation control

codes:

UP = 19

down = 20

right = 22

$$\underline{L_{\text{est}} = 21}$$

- في أوقات تعلم مع method ما في ايد ما بتسحب والاصل ايد زمني ال focus على زمني ما يكون على focus ايد canvas تبعنا و آخر نقطة

- 1) go to the xml file and give our draw (id)
- 2) go to the main and create an obj of our drawing class using the id
- 3) Use the method `drawObj.setFocassable(true)`
- 4) use an if-statement checking the `KeyCode` and changing the directions Based on it. ^{u can use}
or we can use `(KeyEvent.KEYCODE_DPAD_...)` ^{all}
all of these are constants of the `KeyCode`s.

3) control Animation with screen Touching:

الرسالة
خلق الصفوة

- 1) override the onTouchEvent method

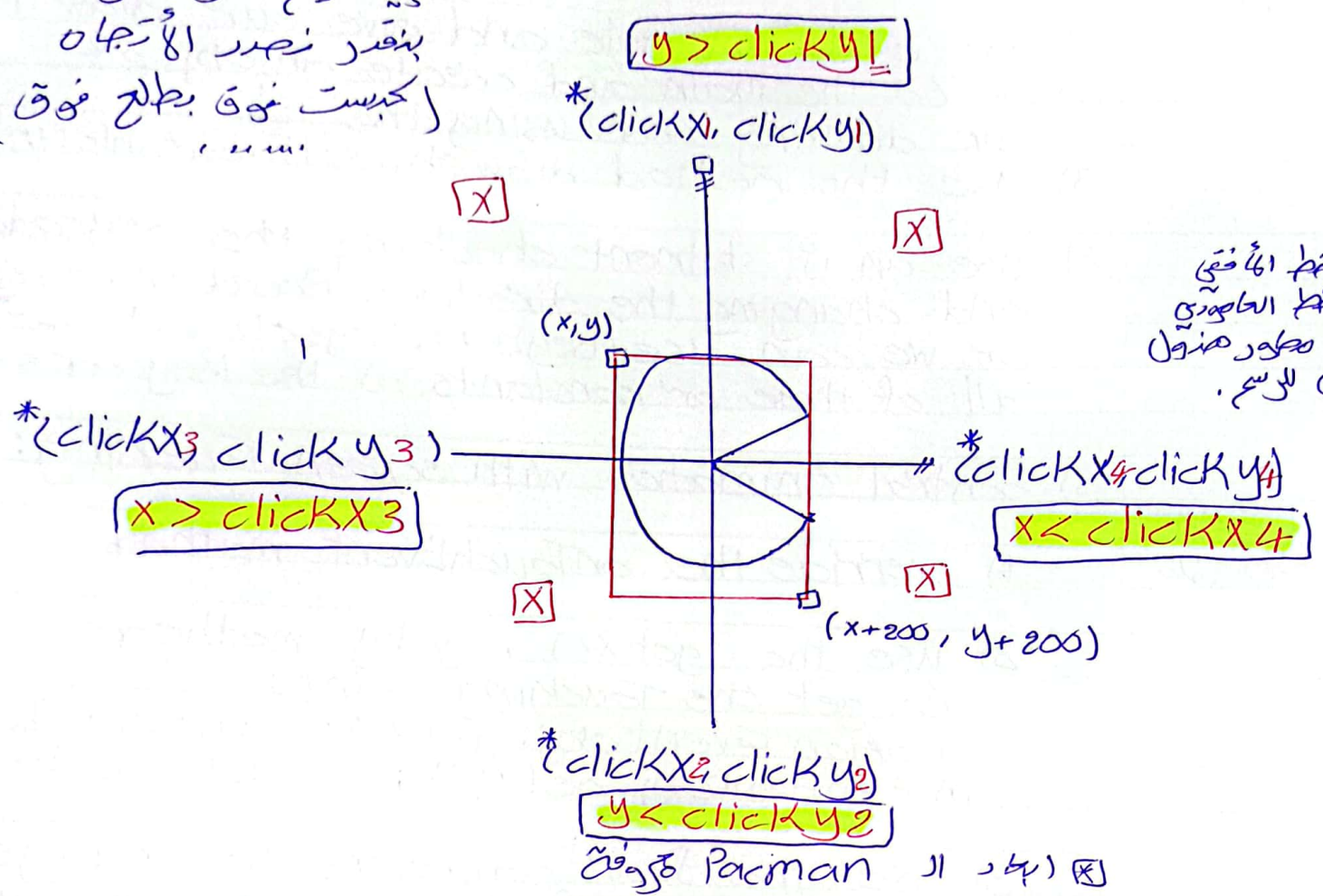
- 8) use the `getX()`, `getY` methods to get the Touching points

MotionEvent obj. getX(); These methods
MotionEvent obj. getY(); returns float.

- 3) use the four conditions to change the directions of the shape that makes.

ابسط صه هو انه نقارن قيمته
 x اليه مع x او y اليه مع y .

من خلال ار 4 شوط
 اليه عليه highlight
 بنقدر نحدد الاتجاه
 (كجست فوق بطلع فوق)



* هائي نقاطه كس الشاشة بلأبعاد المعروفة
 يعني اشي بسيطه مثنى كل الماكس

النقاط الأربعه الحياه دالي لو كجستنا فيهم
 نخلصن بدنا نفكي ب 3 اشياء : 1) ار x
 2) ار y
 3) الاتجاه نيج. ار Pacman