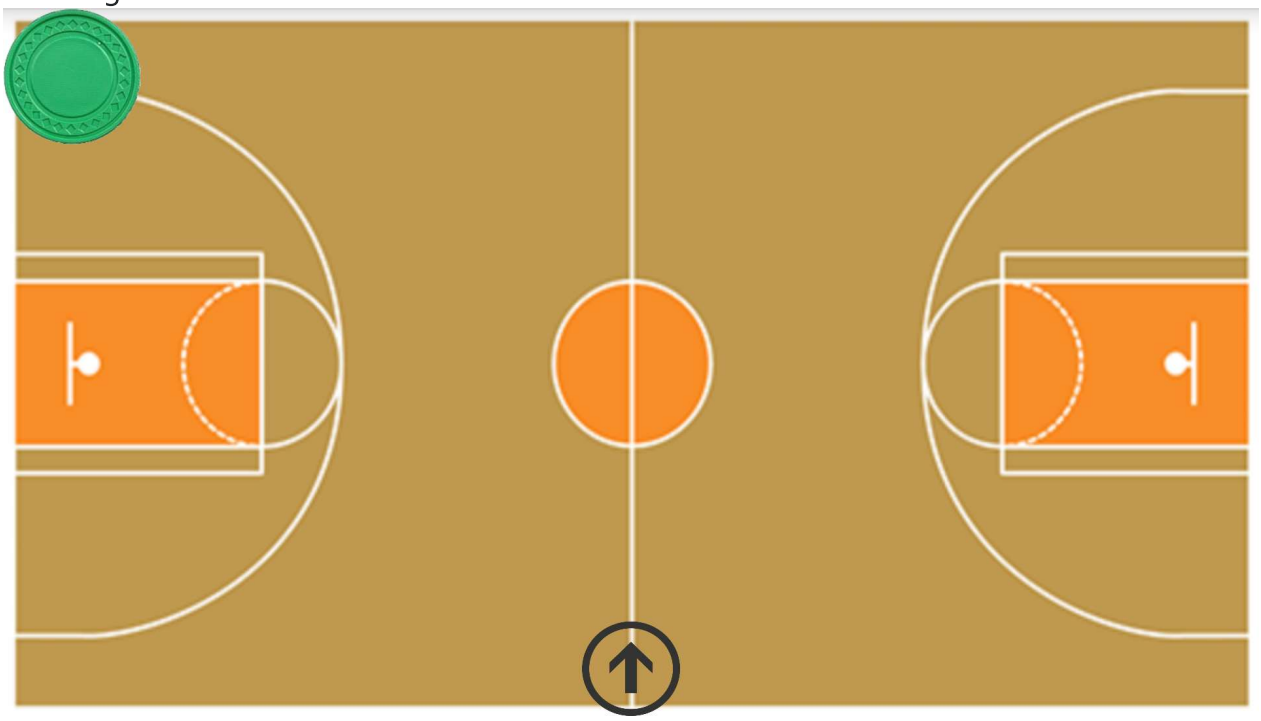
How to Draw Line in Android by Dragging

Asked 7 years, 8 months agoModified 3 years, 5 months agoViewed 2k times

0

I am creating a coaching board. I aim to draw a line between two points. Initial position is in the image below:



If I drag the chip away from the initial position, it should draw a line. When I move the chip, it should always draw a line.

This is what I tried:

```
case MotionEvent.ACTION_UP:
        Toast.makeText(this, "here", Toast.LENGTH_SHORT).show();
        Bitmap bitmap = Bitmap.createBitmap((int) getWindowManager()
                .getDefaultDisplay().getWidth(), (int) getWindowManager()
                .getDefaultDisplay().getHeight(), Bitmap.Config.ARGB_8888);
        Canvas canvas = new Canvas(bitmap);
        Paint paint = new Paint();
        paint.setColor(Color.BLACK);
        paint.setStrokeWidth(10);
        int startx = 50;
        int starty = 100;
        int endx = 150;
        int endy = 210;
        canvas.drawLine(startx, starty, endx, endy, paint);
        break;
```

I put that code inside the ontouchlistener of the chip. You can download the project here: https://www.dropbox.com/s/ggfbsbkaokj9vxi/CoachingBoard.rar?dl=0

android

Share Improve this question Follow edited Feb 3, 2015 at 9:25 asked Feb 3, 2015 at 5:28

Jayson Tamayo
2,623 3 45 76

What a straight line? from the ACTION_DOWN point to the ACTION_UP point? – weston Feb 3, 2015 at 9:32

About the toast: Do you know about Log class and the logcat? – weston Feb 3, 2015 at 9:32

@weston Yes a straight line. I know the Log class and I'm very much aware of the Logcat. But it is easier to see in a toast if it executes a certain block of code rather than scrolling down thru the logcat list. – Jayson Tamayo Feb 3, 2015 at 9:35

Sorted by:

2 Answers

Highest score (default)

1

+200

After further examining your code, I believe I have achieved what you'd like.

We're going to go to DrawingView and define a getter for drawCanvas, so we can access our canvas outside of the DrawingView class.

Next we're going to head to Basketball and do the following:

```
float startX;
float startY;
public boolean onTouch(View view, MotionEvent event) {
    switch(event.getAction()) {
        case MotionEvent.ACTION_DOWN:
            startX = event.getRawX();
            startY = event.getRawY();
            ...
            break;

        case MotionEvent.ACTION_UP:
            Paint paint = new Paint(); //set this as a field in drawView with another getter to avoid garbage collection penalties
            paint.setStrokeWidth(15f);
            paint.setColor(Color.BLACK);
            drawView.getCanvas().drawLine(startX, startY, event.getRawX(), event.getRawY(), paint);
            drawView.invalidate();
            break;
    }
}
```

What this does: when you pick up the chip it will save the starting coordinates, and when you drop the chip it will draw a line in your drawView canvas from start to end.

You can even draw the lines as a continuous Path so the lines always touch, but that is outside the context of this answer.

ShareImprove this answerFollow edited Oct 18, 2016 at 23:25 answered Feb 12, 2015 at 23:39

MeetTitan
3,208 1 12 25

Thanks! It works well. But why does the lines are gone after moving other chips? – Jayson Tamayo Feb 13, 2015 at 0:52

@JaysonTamayo, I have no idea, I cannot replicate that in my environment. Lines are not erased. If you feel I've helped you, consider clicking the check mark on the left :-) – MeetTitan Feb 13, 2015 at 1:02

The lines are not erased in your side? – Jayson Tamayo Feb 13, 2015 at 1:07

No, the lines stay regardless or which or how many chips I move. – MeetTitan Feb 13, 2015 at 1:08

Can you send me your project file please? :) Maybe I've done something wrong. – Jayson Tamayo Feb 13, 2015 at 1:09

0

the best solution to draw a line by dragging is to get x and y positions from ACTION_DOWN and then get ever next x and y position from ACTION_MOVE, outside the switch statement, invalidate the canvas. draw the line with this starting x,y and ending x,y positions.

Code of these all explanation is below:

```
public class TouchEventView extends View {
float downxpos;
float downypos;
float upxpos;
float upypos;
private Paint paint = new Paint();
private Path path = new Path()
public TouchEventView(Context context, AttributeSet attrs) {
    super(context, attrs);
    paint.setColor(Color.GREEN);
}

@Override
protected void onDraw(Canvas canvas) {
        canvas.drawLine(downxpos, downypos, upxpos, upypos, paint);
}

@Override
public boolean onTouchEvent(MotionEvent event) {
    switch (event.getAction()) {
        case MotionEvent.ACTION_DOWN:
            downxpos = event.getX();
            downypos = event.getY();
        case MotionEvent.ACTION_MOVE:
            upxpos = event.getX();
            upypos = event.getY();
            break;
        default:
```

```
            return false;
        }
        invalidate();
        return true;
    }

    }
```

hope this will help you and other community if you need more description feel free to ask.

ShareImprove this answerFollow

answered Apr 12, 2019 at 17:20

asadullah

111 1 8