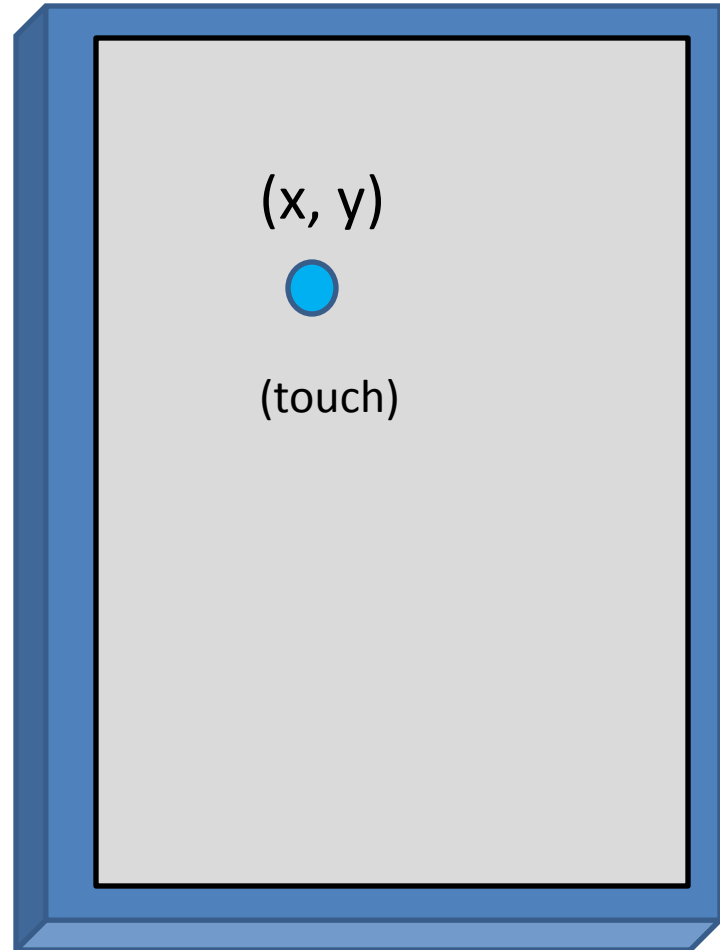


Reading TouchEvent Data Android Java

Mr. Michaud
Marist School

User Touch Events

- What type of touch?
- Where touch:
 - Started
 - Is currently
 - Stopped
- What is the ID number of the touch?



Structure of onTouchEvent() Function

```
public boolean onTouchEvent(MotionEvent event) {  
  
    int action = event.getActionMasked();  
    int actionIndex = event.getActionIndex();  
  
    return true; // required  
}
```

The object `event` represents the object through which we will gather the touch data.

The integer `action` holds the type of touch event.

The integer `actionIndex` holds the ID of the touch (which touch in multi touch situations)

Android Touch Events

- `MotionEvent.ACTION_DOWN`
- `MotionEvent.ACTION_UP`
- `MotionEvent.ACTION_MOVE`

And many others . . .

MotionEvent Data

Given a MotionEvent object called `event` and

An object named `actionIndex` for the index of event:

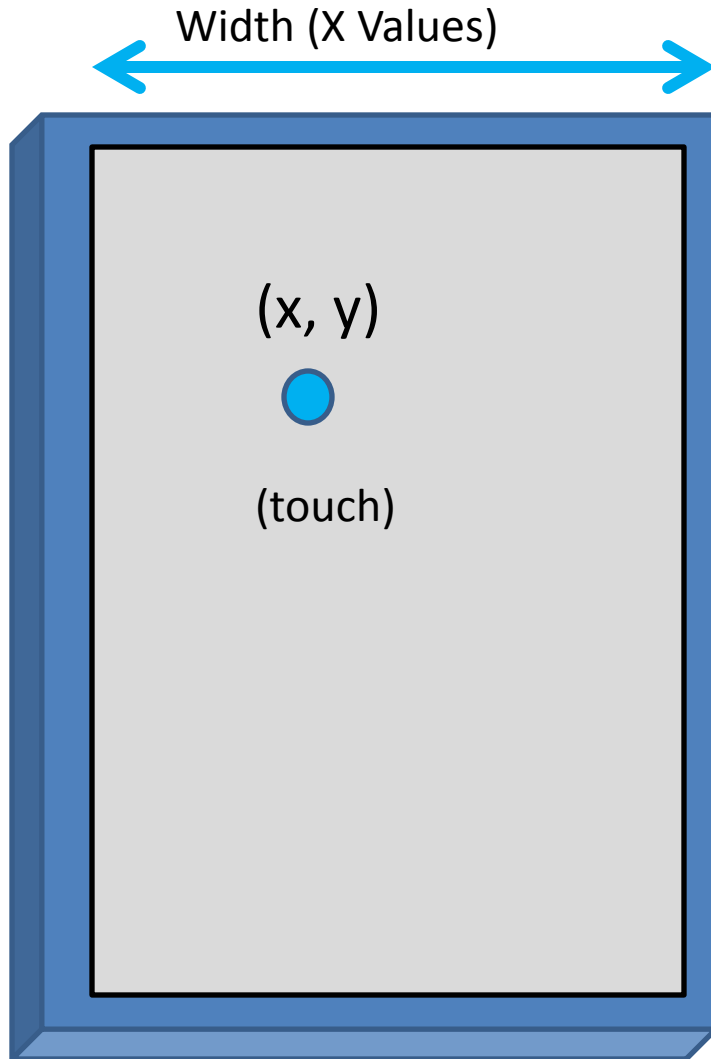
```
event.getX(actionIndex)
```

```
// Returns X position of touch at  
actionIndex
```

```
event.getY(actionIndex)
```

```
// Returns Y position of touch at  
actionIndex
```

MotionEvent Data



Height (Y Values)

```
float x =  
event.getX(actionIndex);  
  
float y =  
event.getY(actionIndex);
```

Selected Action Index Values:

- 0 indicates Touch Down
- 1 indicates Touch Up
- 2 indicates Touch Moved

Example from Doodle App:

```
100
101 public boolean onTouchEvent(MotionEvent event)
102 {
103     // get the event type and the ID of the pointer
104     int action = event.getActionMasked(); // event type
105     int actionIndex = event.getActionIndex(); // pointer
106
107     // determine which type of action the given motion event
108     // represents, then call the corresponding handling method
109     if (action == MotionEvent.ACTION_DOWN || action == MotionEvent.ACTION_POINTER_DOWN)
110     {
111         touchStarted(event.getX(actionIndex), event.getY(actionIndex), event.getPointerId(actionIndex));
112     } // end if
113     else if (action == MotionEvent.ACTION_UP || action == MotionEvent.ACTION_POINTER_UP)
114     {
115         touchEnded(event.getPointerId(actionIndex));
116     } // end else if
117     else
118     {
119         touchMoved(event);
120     } // end else
121
122     invalidate(); // re draw
123     return true;
124 } // end method on TouchEvent
125
```