

Image array

Click part of the image to store a number in a 2d array representing the rectangles of the image.

Click a rectangle multiple times and it will toggle between 1 and 2.

A colour-coded rectangle appears on the image to show what value was in the array before it was updated.

This is intended as a demo of one way to treat an image as a grid.

[imageArray.js](#)

```
var cellData=[],img;
//Called when application is started.
function OnStart()
{
    //Create a layout with objects vertically centered.
    var lay = app.CreateLayout( "linear", "VCenter,FillXY" );

    //Create image control and add it to layout.
    img = app.CreateImage( null, 0.9, 0.5 );
    img.cellWidth=0.1;
    img.cellHeight=0.1;

    img.SetColor("#ff999999");
    img.SetOnTouch(imgOnTouch);

    lay.AddChild( img );

    // horizontal layout for buttons
    var hlay=app.CreateLayout("Linear","Horizontal");
    lay.AddChild(hlay);
    //buttons
    var btnSave=app.CreateButton("Save");
    btnSave.SetOnTouch(btnSaveOnTouch);
    hlay.AddChild(btnSave);
    var btnLoad=app.CreateButton("Load");
    btnLoad.SetOnTouch(btnLoadOnTouch);
    hlay.AddChild(btnLoad);
    var btnClear=app.CreateButton("Clear");
    btnClear.SetOnTouch(btnClearOnTouch);
    hlay.AddChild(btnClear);
    var btnFresh=app.CreateButton("Refresh");
    btnFresh.SetOnTouch(populate);
    hlay.AddChild(btnFresh);

    //Add layout to app.
    app.AddLayout( lay );
}
```

```
//force initial load fro file
btnLoadOnTouch();
}

//user touched image - update cell
function imgOnTouch(ev)
{
    var row,col,val ;
    if(ev.action==="Down")
    {
        row = Math.floor(ev.Y/this.cellHeight);
        col = Math.floor(ev.X/this.cellWidth);
        val = getData(row,col);
        val = (val+1)%2;
        setData(row,col,val);
        drawCell(row,col);
    }
}

//draw a single value to img
function drawCell(row,col)
{
    var val=getData(row,col)
    var x1 = col*img.cellWidth;
    var y1 = row*img.cellHeight;
    var x2 = x1 + img.cellWidth;
    var y2 = y1 + img.cellHeight;
    img.SetLineWidth(1);
    img.SetPaintStyle("Fill")
    if(val < 0)
    {
        img.SetPaintColor("#ffff0000");
        img.SetPaintStyle("Line")
        img.DrawRectangle(x1,y1,x2,y2);
    }
    else if(val === 0)
    {
        img.SetPaintColor("#ff00ff00");
        img.DrawRectangle(x1,y1,x2,y2);
    }
    else if(val === 1)
    {
        img.SetPaintColor("#ff0000ff");
        img.DrawRectangle(x1,y1,x2,y2);
    }
}

// draw all data to img
function populate()
{
```

```
var row, cell, dataLen, rowLen, i, j;
dataLen = cellData.length;
img.SetAutoUpdate(false);
img.Clear();
img.SetColor("#ff999999");
for(i=0; i<dataLen ;i++)
{
    row = cellData[i];
    rowLen = row.length;
    if(rowLen===0) rowLen=1 ;
    for(j=0;j<rowLen;j++)
    {
        drawCell(i,j);
    }
}
img.Update();
img.SetAutoUpdate(true);
}

// get a value from 2d array
function getData(row,col)
{
    var ret;
    if (cellData.length < row +1)
        return -1; // represents unset
    ret = cellData[row][col];
    if(ret===undefined||ret===null)
        return -1;
    return ret;
}

//write a value to 2d array
function setData(row,col,value)
{
    // create any intervening rows
    while (cellData.length < row +1)
    {
        cellData.push([]);
    }
    cellData[row][col]=value;
}

//save data to file
function btnSaveOnTouch()
{
    var data = JSON.stringify(cellData);
    app.WriteFile("/sdcard/picData",data);
}

// get data from file and redraw
function btnLoadOnTouch()
```

```
{
    var data = app.ReadFile("/sdcard/picData");
    cellData = JSON.parse(data);
    populate();
}

// clear data and redraw
function btnClearOnTouch()
{
    cellData = [];
    populate();
}
```

From:

<http://sgarman.net/wiki/> - **DroidScript** wiki

Permanent link:

http://sgarman.net/wiki/doku.php?id=sample_code:image_array

Last update: **2015/09/29 09:27**

