/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* QML with Highlight.js \*\*/

import QtQuick 2.5 // good version

Window {

id: root

width: 1024; height: 600

color: "black"

property int highestZ: 0 // 0 is lowest, +infinity is highest

property real defaultSize = 200.1

signal activated(real xPosition, real yPosition)

// show the file picker

FileDialog {

id:fileDialog // an id in a comment should not be detected

title: "Choose a folder with some images"

onAccepted: folderModel.folder = fileUrl + "/" // if this is on property

}

Flickable {

id: flickableproperty

contentHeight: height \* surfaceViewportRatio

property real zRestore: 0

Behavior on scale { NumberAnimation { duration: 200 } }

Repeater {

model: FolderListModel {

id: folderModel

nameFilters: ["\*.png", "\*.jpg", "\*.gif"]

}

Component.onCompleted: {

var x;

x = Math.random() \* root.width - width / 2

rotation = Math.random() \* 13 - 6

if (pinch.scale > 0) {

photoFrame.rotation = 0;

photoFrame.scale = Math.min(root.width, root.height) / Math.max(image.sourceSize.width, image.sourceSize.height) \* 0.85

} else {

photoFrame.rotation = pinch.previousAngle

photoFrame.scale = pinch.previousScale

}

}

function setFrameColor() {

if (currentFrame)

currentFrame.border.color = "black";

currentFrame = photoFrame;

}

}

}

Timer { id: fadeTimer; interval: 1000; onTriggered: { hfade.start(); vfade.start() } }

Component.onCompleted: fileDialog.open()

}