

# Bouncing Menu using Qt Quick

This article demonstrates how to create a drop-down menu with gravitational feel using [Qt Quick](#).

## Introduction

In this article we will see how to create a drop-down menu with gravitational feel. To complete this example we need [QML Easing](#) effect. We will create five [Rectangles](#) and will drop them from top to bottom with *OutBounce* easing effect.



Menu with gravitational feel   Selected menu

 Note: This image isn't animated. Try the example to see it working!

## Implementation

Let's create an empty QML project. We create a [Rectangle](#) on the top of the screen which we call the menu bar and five different [Rectangles](#) with same state.

```
Rectangle{
    id:rectMenuBar
    border.color: "White";
    border.width: 2
    height: 50
    width: 300
    radius: 5
    anchors.horizontalCenter: parent.horizontalCenter
    anchors.top: parent.top
    gradient: Gradient {
        id: idleColor
        GradientStop { position: 0.0; color: "#434343" }
        GradientStop { position: 0.4; color: "#343434" }
        GradientStop { position: 0.6; color: "#202020" }
        GradientStop { position: 1.0; color: "#191919" }
    }
    opacity: mouseAreaMenu.pressed ? 0.5 : 1.0
    MouseArea
    {
        id:mouseAreaMenu
        anchors.fill: parent
        anchors.margins: -5
        height: parent.height+20
        width: parent.width +20
    }
}
```

```

        onClicked: {
            if (rectMenu1.state == '') rectMenu1.state = "right"; else
rectMenu1.state = ''
            if (rectMenu2.state == '') rectMenu2.state = "right"; else
rectMenu2.state = ''
            if (rectMenu3.state == '') rectMenu3.state = "right"; else
rectMenu3.state = ''
            if (rectMenu4.state == '') rectMenu4.state = "right"; else
rectMenu4.state = ''
            if (rectMenu5.state == '') rectMenu5.state = "right"; else
rectMenu5.state = ''
            console.log("Menu Clicked");
        }
    }
    Text {
        id: textMenu
        text: qsTr("Click Me!")
        color: "white"
        anchors.verticalCenter: rectMenuBar.verticalCenter
        anchors.horizontalCenter: rectMenuBar.horizontalCenter
        font.pixelSize: 21
    }
} // endof menu

```

When user clicks on the menu bar it changes the current *state* of the [Rectangles](#) to a new *state* with the *PropertyChanges* of its *y* value and also change its position with an *OutBounce* easing effect.

```

...
states : State {
    name: "right"
    PropertyChanges { target: rectMenu5; y: window.height - 390; }
}
transitions: Transition {
    NumberAnimation { properties: "y"; easing.type: Easing.OutBounce;
duration: 1000 }
}
...

```

Each menu item has a [MouseArea](#) event, which displays a [Dialog](#) when clicked.

```

CommonDialog{
    id:dialog
    titleText: "Menu Dialog"
    onClickedOutside: close();
    content: Item {
        id: itemDialog
        Text {
            id: textDialog
            text:""
            font.pixelSize: 21
            color: "white"
            anchors.top: parent.top
            anchors.topMargin: 10
            anchors.left: parent.left
            anchors.leftMargin: 20
        }
    }
}

```

```
}  
}
```

## Summary

---

This article shows is a very basic use of [QML Easing](#) effect to create an animated menu item which has a gravitational/bouncing feel.

## Source Code

---

- The full source code of the example is available here: [File:BouncingMenuQML.zip](#)

## Reference

---

- [Animation: Easing Example](#)