

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего
образования

«Национальный исследовательский

Нижегородский государственный университет им. Н.И. Лобачевского»

(ННГУ)

Институт информационных технологий, математики и механики

Направление подготовки: «Фундаментальная информатика и информационные
технологии»

Отчет по практическому заданию №3:

«Инструменты разработки мобильных приложений»

Выполнила:

студентка группы 381906-2

Яшина Дарья Степановна

Нижний Новгород

2022

Постановка задачи

Цель: освоить базовые навыки построения пользовательских интерфейсов, позиционирования, отрисовки и перемещения элементов. Научиться анимировать элементы. Научиться создавать диалоги и взаимодействовать с ними.

Шаги:

- 1) Создать новый проект со стандартной заготовкой приложения.
- 2) Нарисовать 3 квадрата красного, зелёного и синего цветов следующим образом:
- 3) Поместить текст “Квадрат” белого цвета по центру синего квадрата.
- 4) Нарисовать 5 квадратов с использованием Column и Row следующим образом:
- 5) Нарисовать те же 5 квадратов с использованием Grid.
- 6) Сделать из квадрата “А” прямоугольник “В” с использованием объектов Translate, Scale и Rotation
- 7) Нарисовать квадрат и анимировать его перемещение вниз с увеличением его размера. Документация по анимации доступна по адресу <http://doc.qt.io/qt-5/qml-qtquick-animation.html>.
- 8) Реализовать диалог с двумя текстовыми полями, в которые вводятся числа. После нажатия на кнопку “Подтвердить” в консоль выводится сумма чисел. Для преобразования строк к числам использовать функцию `parseInt("42")`. Валидацией и обработкой ошибок можно пренебречь
- 9) Дополнительное задание

Руководство программиста

Для реализации данной лабораторной работы нам потребовались следующие инструменты :

1) **Rectangle** - отрисовка квадратов:

width:– ширина квадрата

height: – высота квадрата

color: – цвет квадрата

x: – смещение по оси x

y: – смещение по оси y

2) **Transform** – трансформирование квадратов:

Scale – масштабирование квадрата

Rotation – вращение квадрата

Translate – изменение положения по осям

3) **ParallelAnimation** – анимирование квадрата

running: true – анимация включена

property:– указывает на то, что изменяется

to: – конечное значение анимации

duration: – продолжительность анимации

4) **Column** и **Row** - отрисовка квадратов

spacing: - отступ между строками и столбцами

5) **Grid** - отрисовка сеткой

columns: 3 – число столбцов

rows: 2 – число строк

spacing: – отступ между строками и столбцами

6) **Text** - текст

text: – текст надписи

color: – цвет надписи

anchors.centerIn: – размещение по центру

7) **Dialog** - создание диалога

TextField:

id:

placeholderText:– текст блока

strictValidation: true – указывает на работу выбранного типа

validator: IntValidator{} – разрешен ввод только значений типа int

Button:

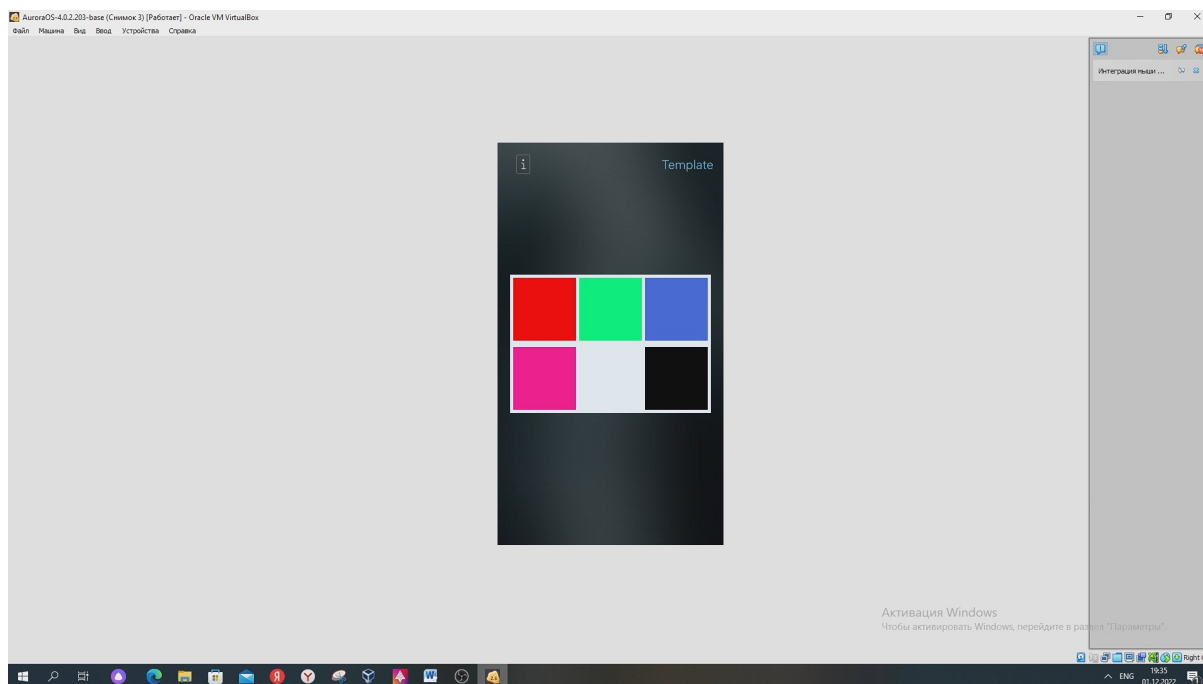
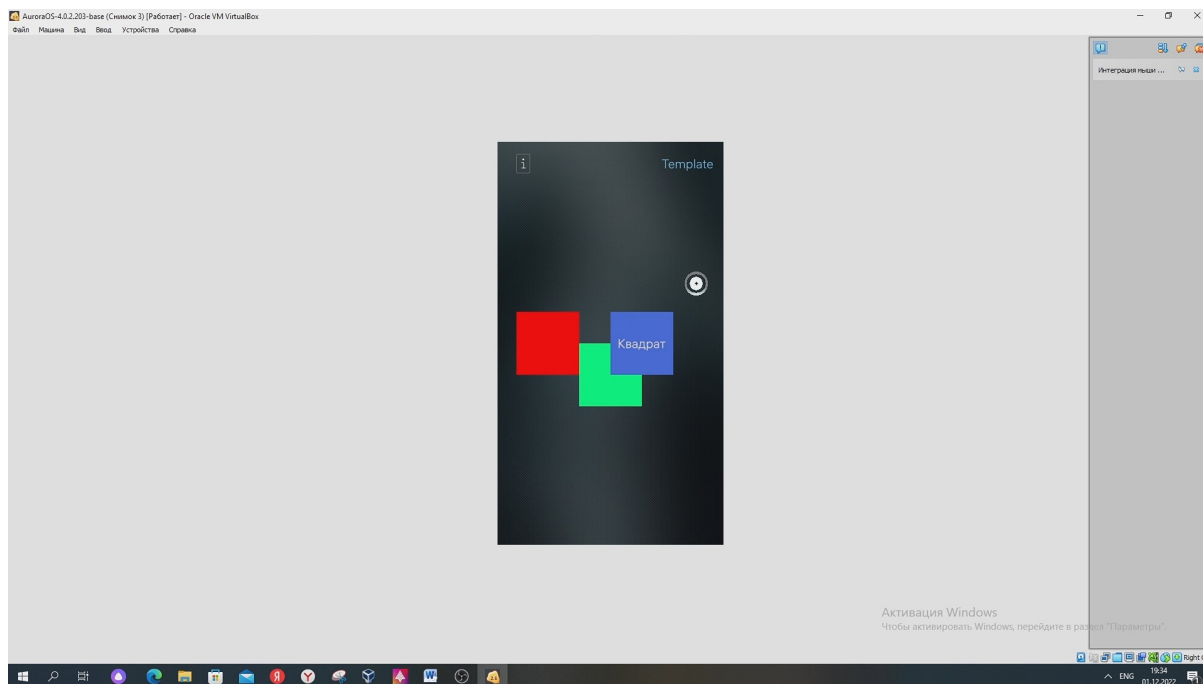
text: – название кнопки

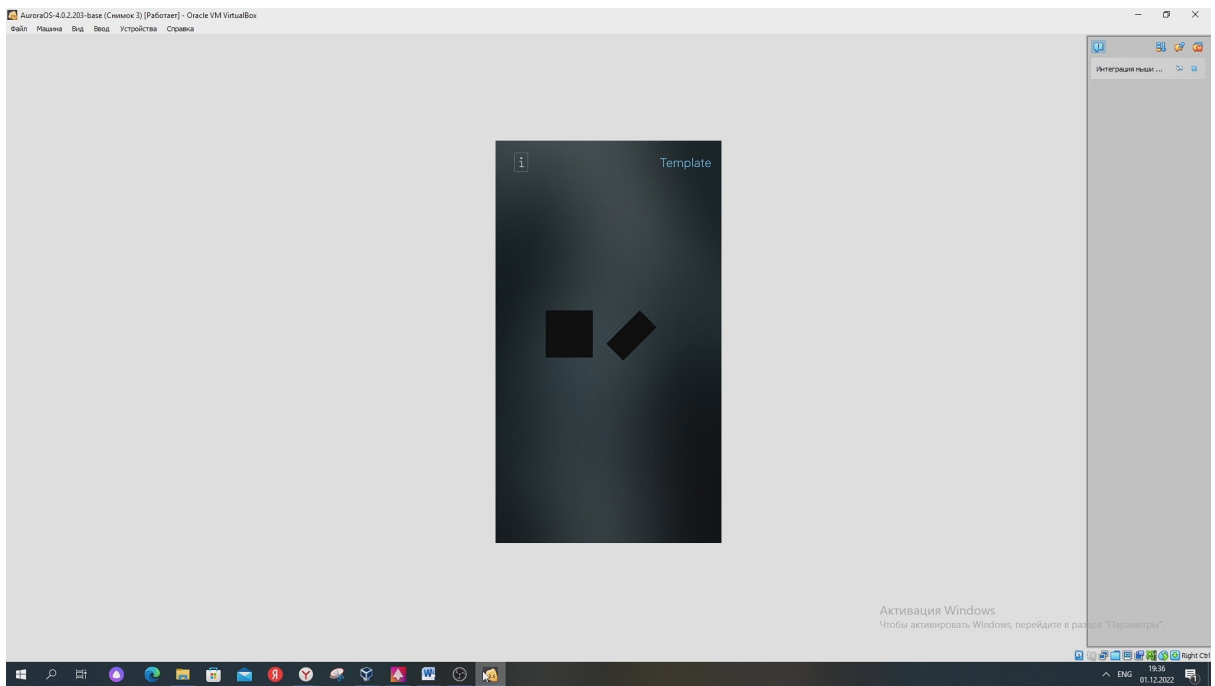
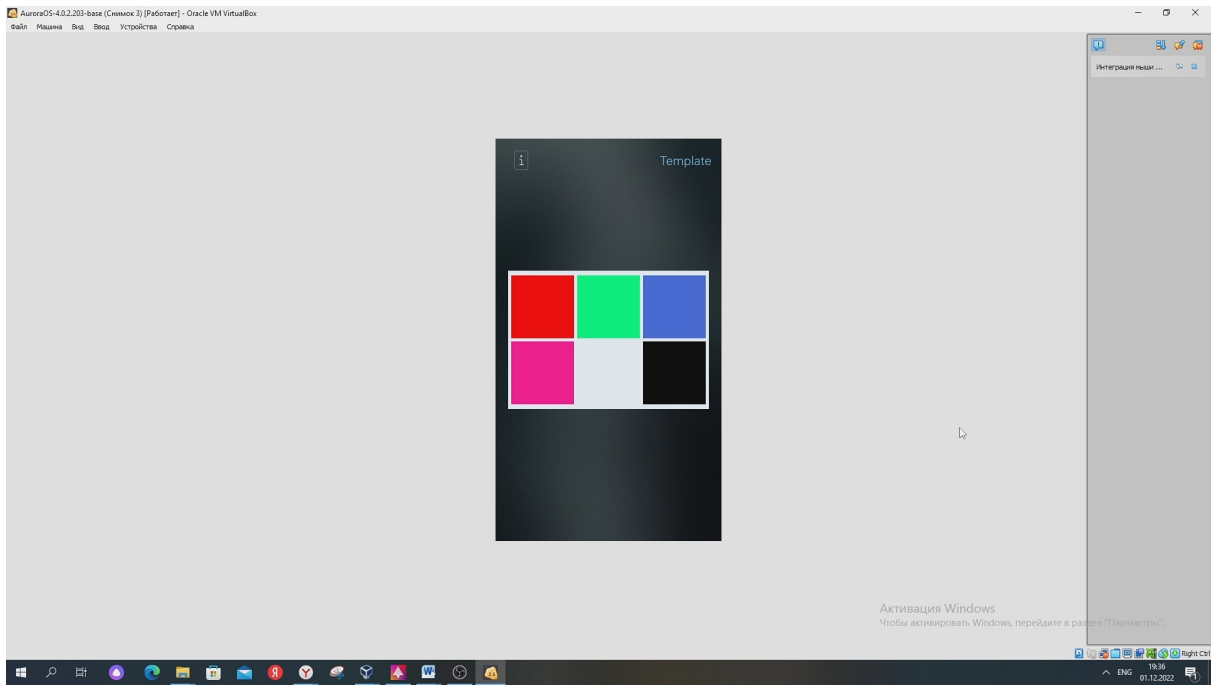
width: *parent.width* – ширина кнопки

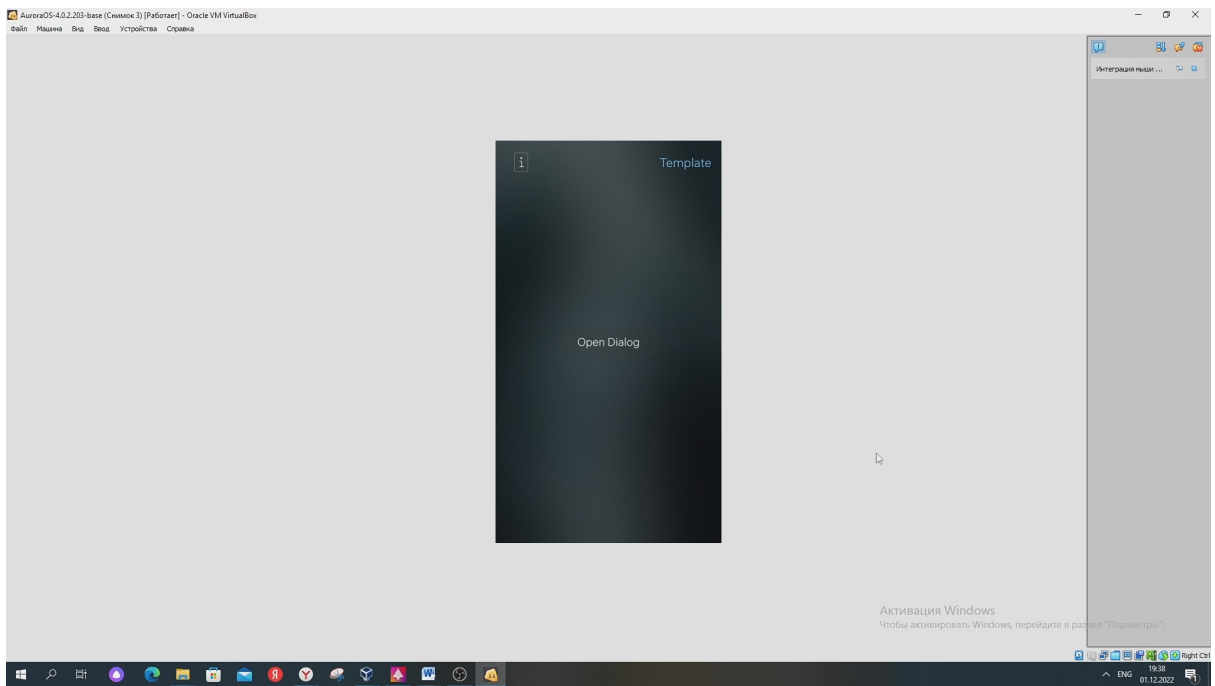
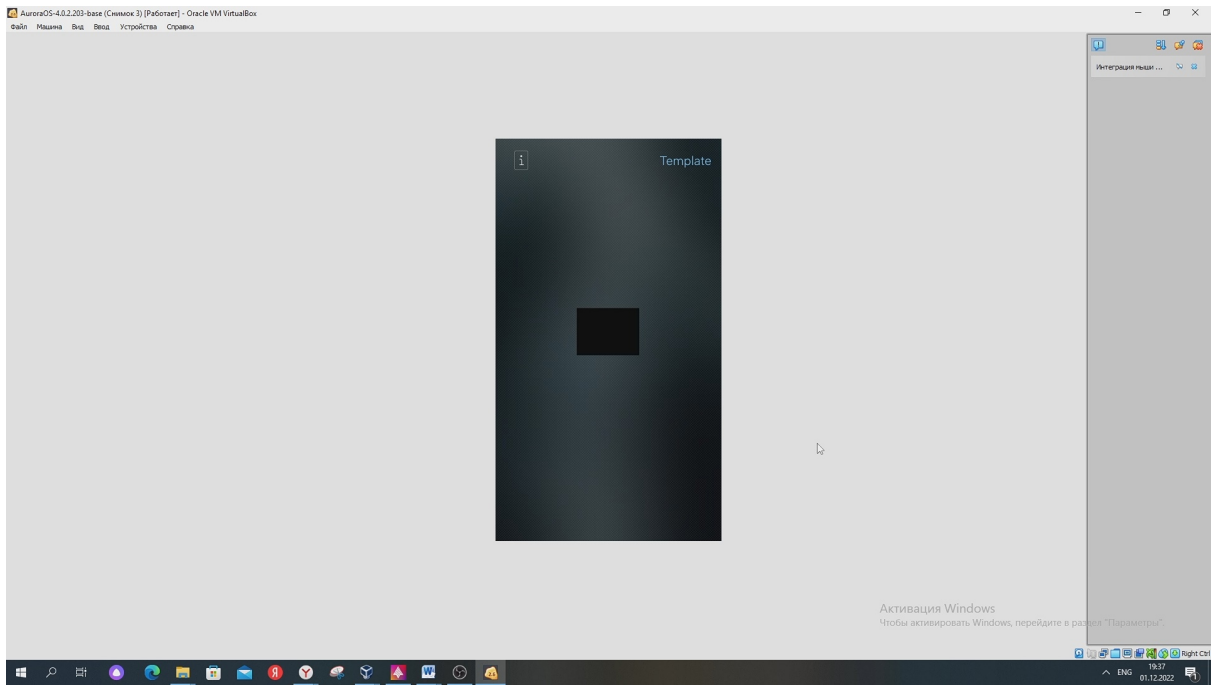
onClicked: *console.log(parseInt(num1.text) + parseInt(num2.text))* –
обработчик события

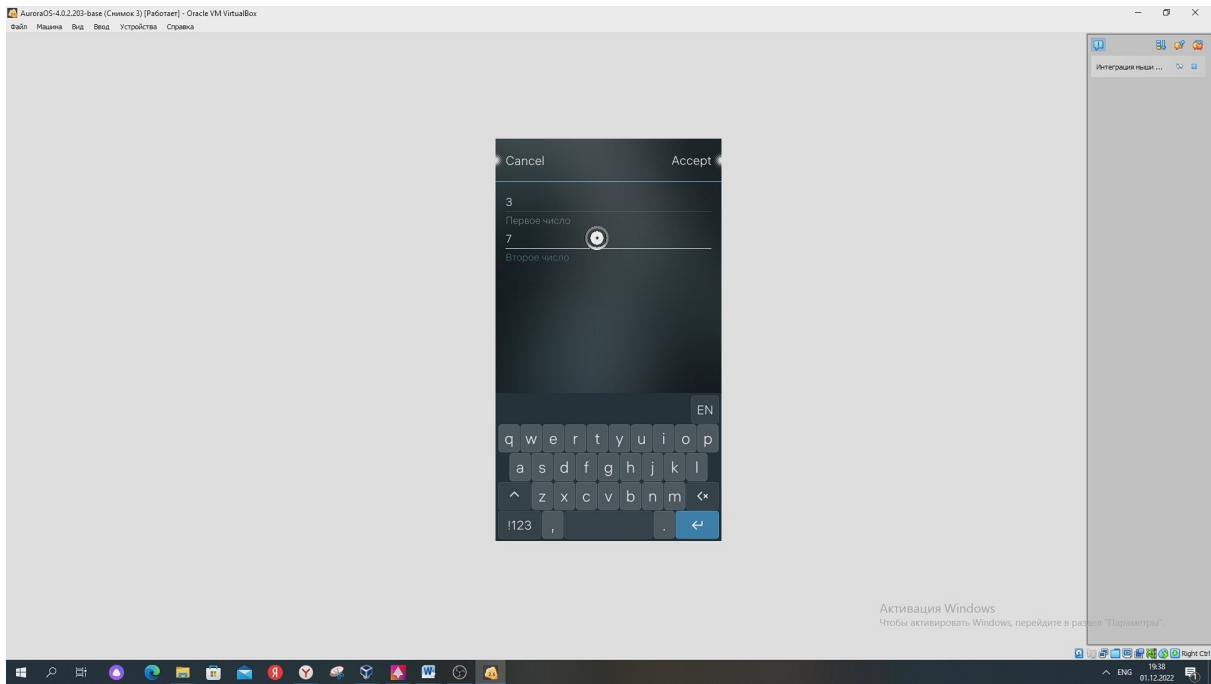
Руководство пользователя

После запуска программы, открывается окно эмулятора, в котором отображается страницы с данным практическим заданием:

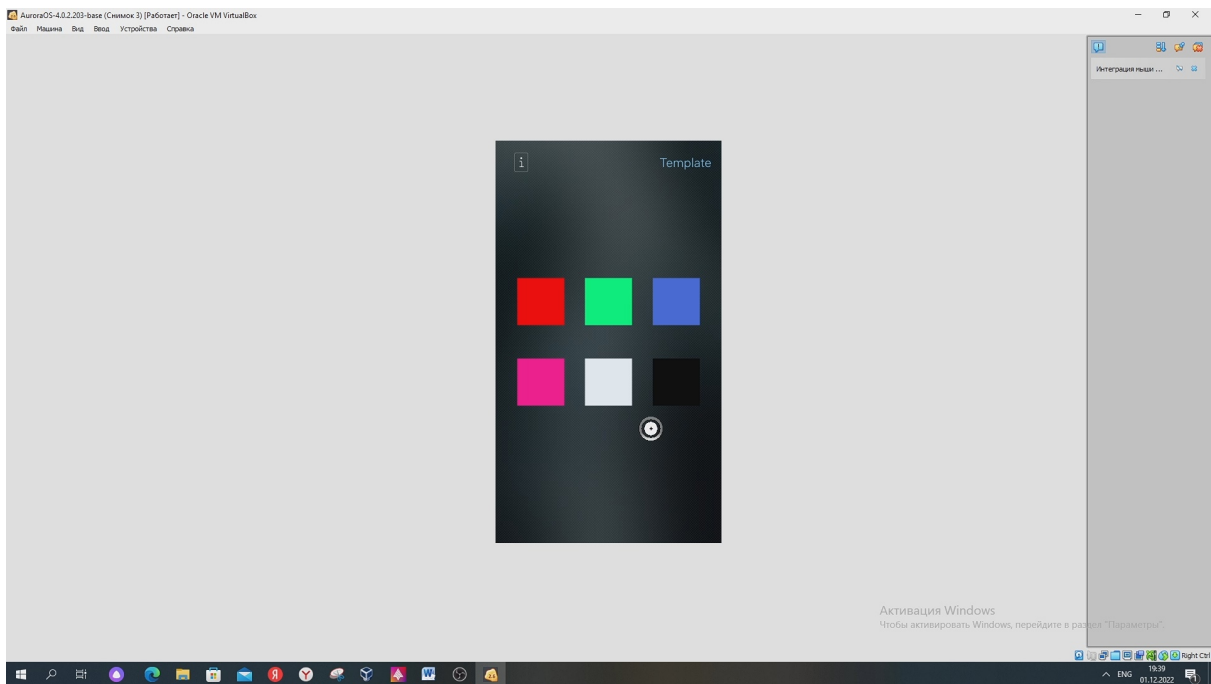








Дополнительное задание:



Заключение

В данной лабораторной работе удалось освоить базовые навыки построения пользовательских интерфейсов, позиционирования, отрисовки и перемещения элементов. Я научилась анимировать элементы, создавать диалоги и взаимодействовать с ними.

Приложение

```
import QtQuick 2.0

import Sailfish.Silica 1.0

Page {

    objectName: "Task_1_2_3"

    allowedOrientations: Orientation.All


    PageHeader {

        objectName: "pageHeader"

        title: qsTr("Template")

        extraContent.children: [

            IconButton {

                objectName: "aboutButton"

                icon.source: "image://theme/icon-m-about"

                anchors.verticalCenter: parent.verticalCenter


                onClicked: pageStack.push(Qt.resolvedUrl("AboutPage.qml"))

            }

        ]

    }

    Row {

        id: row

        anchors.centerIn: parent

        spacing: Theme.paddingLarge

        Item {

            id: item
```

```
//x: 10
```

```
//y: 10
```

```
width: 600
```

```
height: 200
```

```
Rectangle {
```

```
    id:red
```

```
    x:0
```

```
    y:0
```

```
    width: 200
```

```
    height: 200
```

```
    color: "#ff0000"
```

```
}
```

```
Rectangle {
```

```
    id:green
```

```
    width: 200
```

```
    height: 200
```

```
    anchors.top: red.verticalCenter
```

```
    anchors.left: red.right
```

```
    color: "#00ff7f"
```

```
}
```

```
Rectangle {
```

```
    id:blue
```

```
    //z:10
```

```

        width: 200

        height: 200

        anchors.left: green.horizontalCenter

        color: "#4169e1"

        Text {

            id: text_blue

            anchors.centerIn: blue

            font.pointSize: 30

            color: "#faebd7"

            text: qsTr("Квадрат")

        }

    }

}

}

}

```

```

import QtQuick 2.0

import Sailfish.Silica 1.0

```

```

Page {

    objectName: "Task_4"

    allowedOrientations: Orientation.All


    PageHeader {

        objectName: "pageHeader"

        title: qsTr("Template")

        extraContent.children: [

            IconButton {

```

```

        objectName: "aboutButton"

        icon.source: "image://theme/icon-m-about"

        anchors.verticalCenter: parent.verticalCenter

        onClicked: pageStack.push(Qt.resolvedUrl("AboutPage.qml"))
    }
]
}

Column {

    id:column

    anchors.centerIn: parent

    //spacing: 5

    Rectangle {

        width: 640

        height: 220

        color: "#f0f8ff"

        Row {

            id:row1

            anchors.verticalCenter: parent.verticalCenter

            anchors.horizontalCenter: parent.horizontalCenter

            spacing: 10


            Rectangle {

                width: 200

                height: 200

                color: "#ff0000"

            }

```

```
    Rectangle {
        width: 200
        height: 200
        color: "#00ff7f"
    }

    Rectangle {
        width: 200
        height: 200
        color: "#4169e1"
    }
}
```

```
Rectangle {
    width: 640
    height: 220
    color: "#f0f8ff"

    Row {
        id: row2

        anchors.verticalCenter: parent.verticalCenter
        anchors.horizontalCenter: parent.horizontalCenter
        spacing: 10
    }
}
```

```
Rectangle {
    width: 200
    height: 200
    color: "#ff1493"
```

```

        }

        Rectangle {
            width: 200
            height: 200
            color: "#f0f8ff"
        }

        Rectangle {
            width: 200
            height: 200
            color: "#000000"
        }
    }
}

```

```

import QtQuick 2.0
import Sailfish.Silica 1.0

```

```

Page {
    objectName: "Task_5"
    allowedOrientations: Orientation.All

    PageHeader {
        objectName: "pageHeader"
        title: qsTr("Template")
        extraContent.children: [
            IconButton {
                objectName: "aboutButton"
            }
        ]
    }
}

```

```

        icon.source: "image://theme/icon-m-about"

        anchors.verticalCenter: parent.verticalCenter

        onClicked: pageStack.push(Qt.resolvedUrl("AboutPage.qml"))
    }
}

Column {

    id:col

    anchors.centerIn: parent

    Rectangle {

        width: 640

        height: 440

        color: "#f0f8ff"

        Grid{

            id:grid

            anchors.verticalCenter: parent.verticalCenter

            anchors.horizontalCenter:
parent.horizontalCenter

            spacing: 10

            columns:3

            rows:2

            Rectangle {

                width: 200

                height: 200

                color: "#ff0000"

```



```
}
```

```
Rectangle {
```

```
    width: 200
```

```
    height: 200
```

```
    color: "#00ff7f"
```

```
}
```

```
Rectangle {
```

```
    width: 200
```

```
    height: 200
```

```
    color: "#4169e1"
```

```
}
```

```
Rectangle {
```

```
    width: 200
```

```
    height: 200
```

```
    color: "#ff1493"
```

```
}
```

```
Rectangle {
```

```
    width: 200
```

```
    height: 200
```

```
    color: "#f0f8ff"
```

```
}
```

```
    Rectangle {
```

```
        width: 200
```

```
        height: 200
```

```
        color: "#000000"
```

```
    }
```

```
}
```

```
}
```

```

    }

}

import QtQuick 2.0

import Sailfish.Silica 1.0

Page {

    objectName: "Task_6"

    allowedOrientations: Orientation.All

    PageHeader {

        objectName: "pageHeader"

        title: qsTr("Template")

        extraContent.children: [

            IconButton {

                objectName: "aboutButton"

                icon.source: "image://theme/icon-m-about"

                anchors.verticalCenter: parent.verticalCenter

                onClicked: pageStack.push(Qt.resolvedUrl("AboutPage.qml"))

            }

        ]

    }

    Item {

        id: item
    }

```

```

anchors.centerIn: parent

width: 400

height: 200

Rectangle{

    color: "#000000"

    width: 150

    height: 150}

Rectangle{

    color: "#000000"

    width: 150

    height: 150

    transform:[

        Scale{

            xScale: 0.5},

        Rotation{

            angle: 45},

        Translate{

            x:300}

    ]

}

}

```

```

import QtQuick 2.0

import Sailfish.Silica 1.0

```

Page {

```
objectName: "Task_6"
```

```
allowedOrientations: Orientation.All
```

```
PageHeader {
```

```
objectName: "pageHeader"
```

```
title: qsTr("Template")
```

```
extraContent.children: [
```

```
    IconButton {
```

```
        objectName: "aboutButton"
```

```
        icon.source: "image://theme/icon-m-about"
```

```
        anchors.verticalCenter: parent.verticalCenter
```

```
        onClicked: pageStack.push(Qt.resolvedUrl("AboutPage.qml"))
```

```
    }
```

```
]
```

```
}
```

```
Item {
```

```
id: item
```

```
anchors.centerIn: parent
```

```
width: 600
```

```
height: 600
```

```
Rectangle{
```

```
    id: rect
```

```
    width: 150
```

```
    height: 150
```

```
    color: "#000000"
```

```
ParallelAnimation{  
  
  id:animation  
  
  running: true  
  
  loops: Animation.Infinite
```

```
  NumberAnimation {  
  
    target: rect  
  
    property: "width"  
  
    from: 150  
  
    to: 200  
  
    duration: 1000  
  
  }
```

```
  NumberAnimation {  
  
    target: rect  
  
    property: "hight"  
  
    from: 150  
  
    to: 200  
  
    duration: 1000  
  
  }
```

```
  NumberAnimation {  
  
    target: rect  
  
    property: "y"  
  
    from: 0  
  
    to: 200  
  
    duration: 1000  
  
  }
```

```
  NumberAnimation {  
  
    target: rect
```

```

        property: "x"

        from: 0

        to:200

        duration: 1000

    }

}

}

}

```

```

import QtQuick 2.0
import Sailfish.Silica 1.0
import QtQuick.Layouts 1.1

Page {

    objectName: "Task_8"

    allowedOrientations: Orientation.All


    PageHeader {

        objectName: "pageHeader"

        title: qsTr("Template")

        extraContent.children: [

            IconButton {

                objectName: "aboutButton"

                icon.source: "image://theme/icon-m-about"

                anchors.verticalCenter: parent.verticalCenter


                onClicked: pageStack.push(Qt.resolvedUrl("AboutPage.qml"))
            }
        ]
    }
}

```

```

        }

    ]

}

Button {

    id: openDialogButton

    anchors.centerIn: parent

    text: qsTr("Open Dialog")

    onClicked: {

        dialog.open();

    }

}

```

```

Dialog{

    id:dialog

    Column{

        width: parent.width

        DialogHeader{}

        TextField{

            id: num1

            label: "Первое число"

            placeholderText: "Первое число"

            color : "#f5fffa"

            strictValidation: true

            validator: IntValidator{}

        }

        TextField{

            id: num2

```

```

        label: "Второе число"

        placeholderText: "Второе число"

        color : "#f5fffa"

        strictValidation: true

        validator: IntValidator{}

    }

}

    onAccepted: console.log(parseInt(num1.text) + parseInt(num2.text))

}

}

import QtQuick 2.0

import Sailfish.Silica 1.0

Page {

    objectName: "Task_9"

    allowedOrientations: Orientation.All

    PageHeader {

        objectName: "pageHeader"

        title: qsTr("Template")

        extraContent.children: [

            IconButton {

                objectName: "aboutButton"

                icon.source: "image://theme/icon-m-about"

                anchors.verticalCenter: parent.verticalCenter

```



```
        onClicked: pageStack.push(Qt.resolvedUrl("AboutPage.qml"))
    }

]

}

Rectangle {

width: 150

height: 150

color: "#ff0000"

x: parent.width / 5 - this.width / 2

y: parent.height / 5 * 2 - this.height / 2

}

Rectangle {

width: 150

height: 150

color: "#00ff7f"

x: parent.width / 2 - this.width / 2

y: parent.height / 5 * 2 - this.height / 2

}

Rectangle {

width: 150

height: 150

color: "#4169e1"

x: parent.width / 5 * 4 - this.width / 2

y: parent.height / 5 * 2 - this.height / 2

}
```

```
Rectangle {  
  
width: 150  
  
height: 150  
  
color: "#ff1493"  
  
x: parent.width / 5 - this.width / 2  
y: parent.height / 5 * 3 - this.height / 2  
  
}  
  
Rectangle {  
  
width: 150  
  
height: 150  
  
color: "#f0f8ff"  
  
x: parent.width / 2 - this.width / 2  
y: parent.height / 5 * 3 - this.height / 2  
  
}  
  
Rectangle {  
  
width: 150  
  
height: 150  
  
color: "#000000"  
  
x: parent.width / 5 * 4 - this.width / 2  
y: parent.height / 5 * 3 - this.height / 2  
  
}  
}
```