Каріна Зубко 4 варіант

Лаболаторна 13:

import scalafx.application.JFXApp3

import scalafx.scene.Scene

import scalafx.scene.paint.Color.\_

import scalafx.scene.shape.Circle

import scalafx.animation.\_

import scalafx.util.Duration

import javafx.event.{ActionEvent, EventHandler}

//Zubko

object Main extends JFXApp3 {

private val duration = Duration(1000)

override def start(): Unit = {

stage = new JFXApp3.PrimaryStage {

title = "Traffic Light Simulation"

scene = new Scene(200, 400) {

fill = LightGray

val redLight = new Circle {

centerX = 100

centerY = 100

radius = 50

fill = Red

opacity = 0.3

}

val yellowLight = new Circle {

centerX = 100

centerY = 200

radius = 50

fill = Yellow

opacity = 0.3

}

val greenLight = new Circle {

centerX = 100

centerY = 300

radius = 50

fill = Green

opacity = 0.3

}

content = Seq(redLight, yellowLight, greenLight)

val timeline = new Timeline {

cycleCount = Timeline.Indefinite

keyFrames = Seq(

KeyFrame(duration \* 0, onFinished = new EventHandler[ActionEvent] {

override def handle(event: ActionEvent): Unit = {

redLight.opacity = 1.0

yellowLight.opacity = 0.3

greenLight.opacity = 0.3

}

}),

KeyFrame(duration \* 1, onFinished = new EventHandler[ActionEvent] {

override def handle(event: ActionEvent): Unit = {

redLight.opacity = 1.0

yellowLight.opacity = 1.0

greenLight.opacity = 0.3

}

}),

KeyFrame(duration \* 2, onFinished = new EventHandler[ActionEvent] {

override def handle(event: ActionEvent): Unit = {

redLight.opacity = 0.3

yellowLight.opacity = 1.0

greenLight.opacity = 0.3

}

}),

KeyFrame(duration \* 3, onFinished = new EventHandler[ActionEvent] {

override def handle(event: ActionEvent): Unit = {

redLight.opacity = 0.3

yellowLight.opacity = 0.3

greenLight.opacity = 1.0

}

})

)

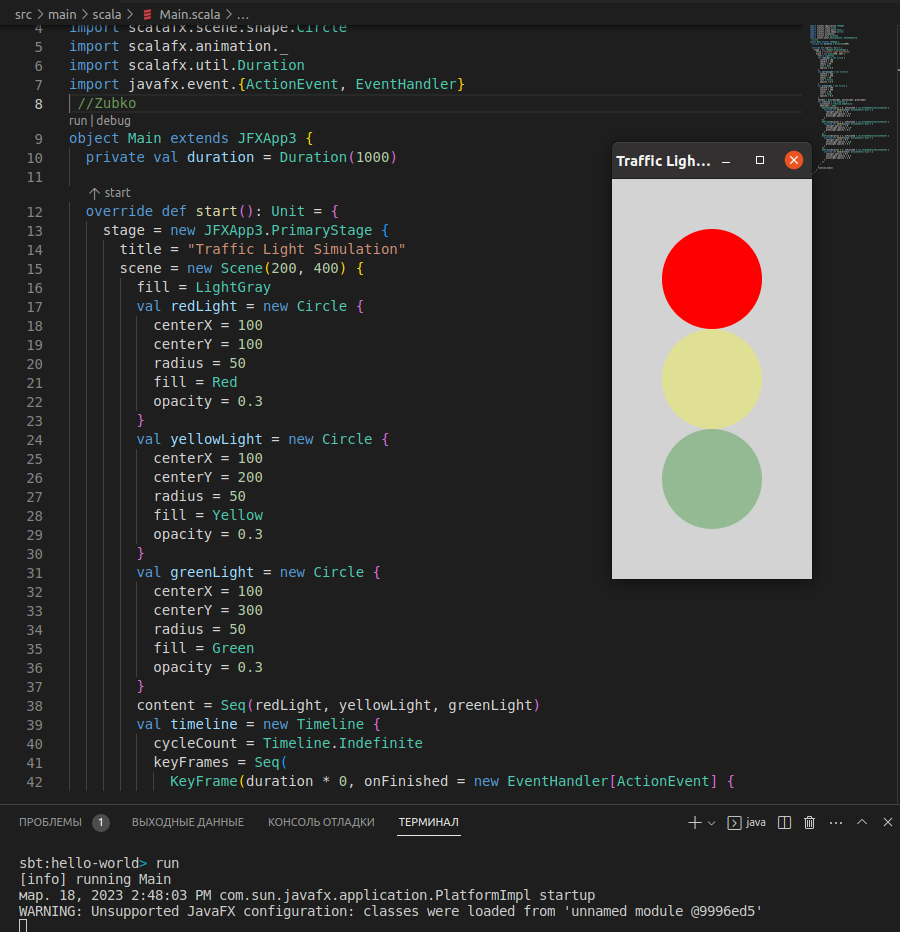
}

timeline.play()

}

}

}

}