



thinkable

Thinkable

Знакомство с
математическими операциями
Урок 2

Создадим новый проект



Create New Project ×

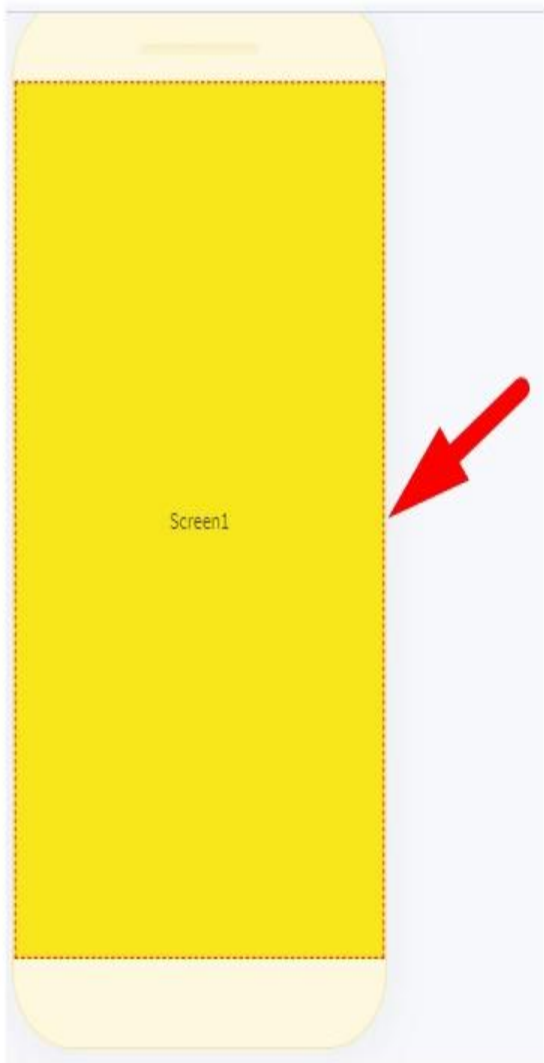
New Project Name:

Private Public

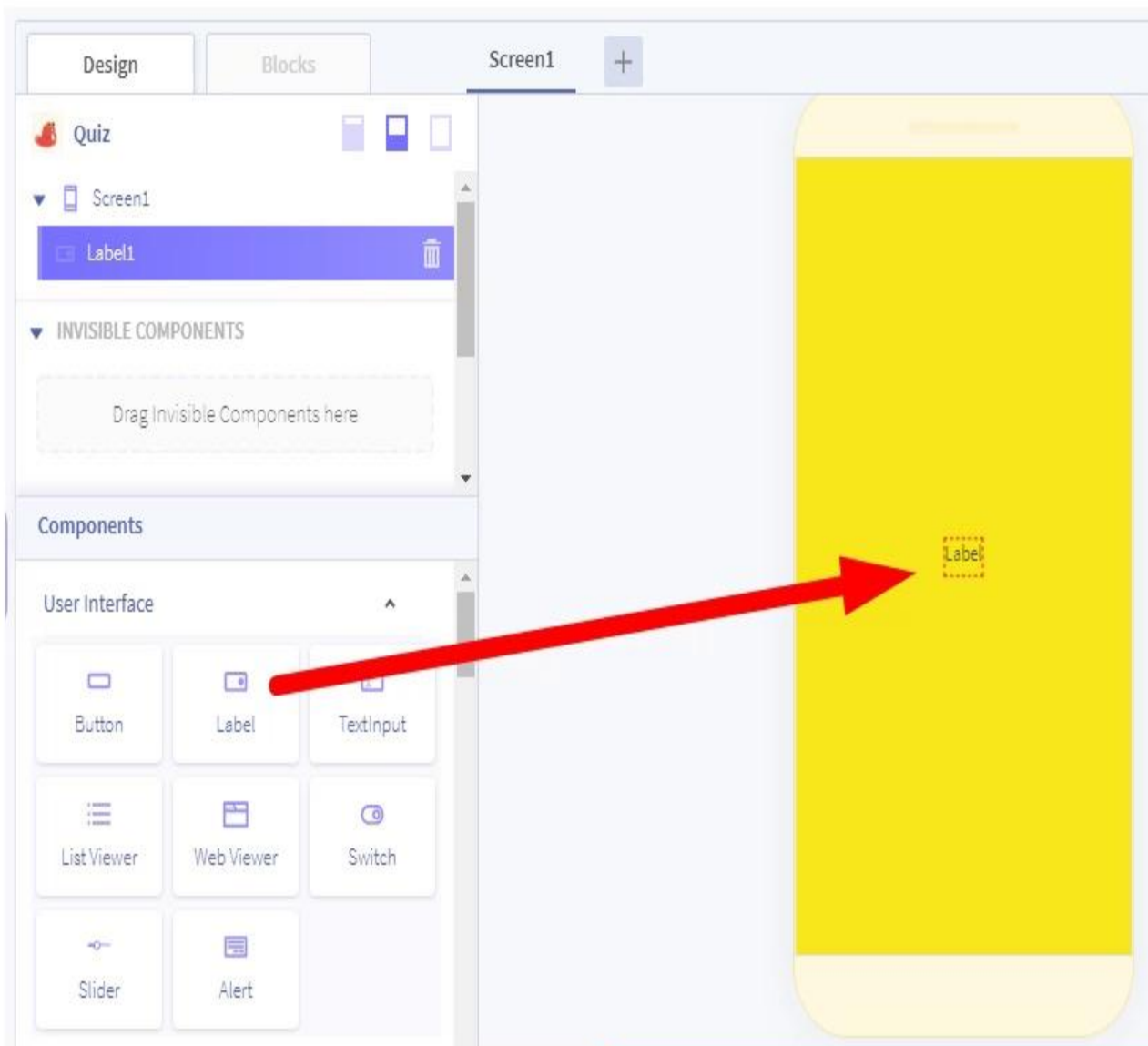
Private projects are only viewable and accessible to their creators.

Cancel Create

Изменим цвет фона



Добавим надпись



Настроим надпись



Label1

< Simple Adv

Text

Игра "Угадай логотип" X

FontSize

24

Two red arrows point to the 'Adv' tab and the 'FontSize' input field.

FontStyle

normal

Height

Absolute Size

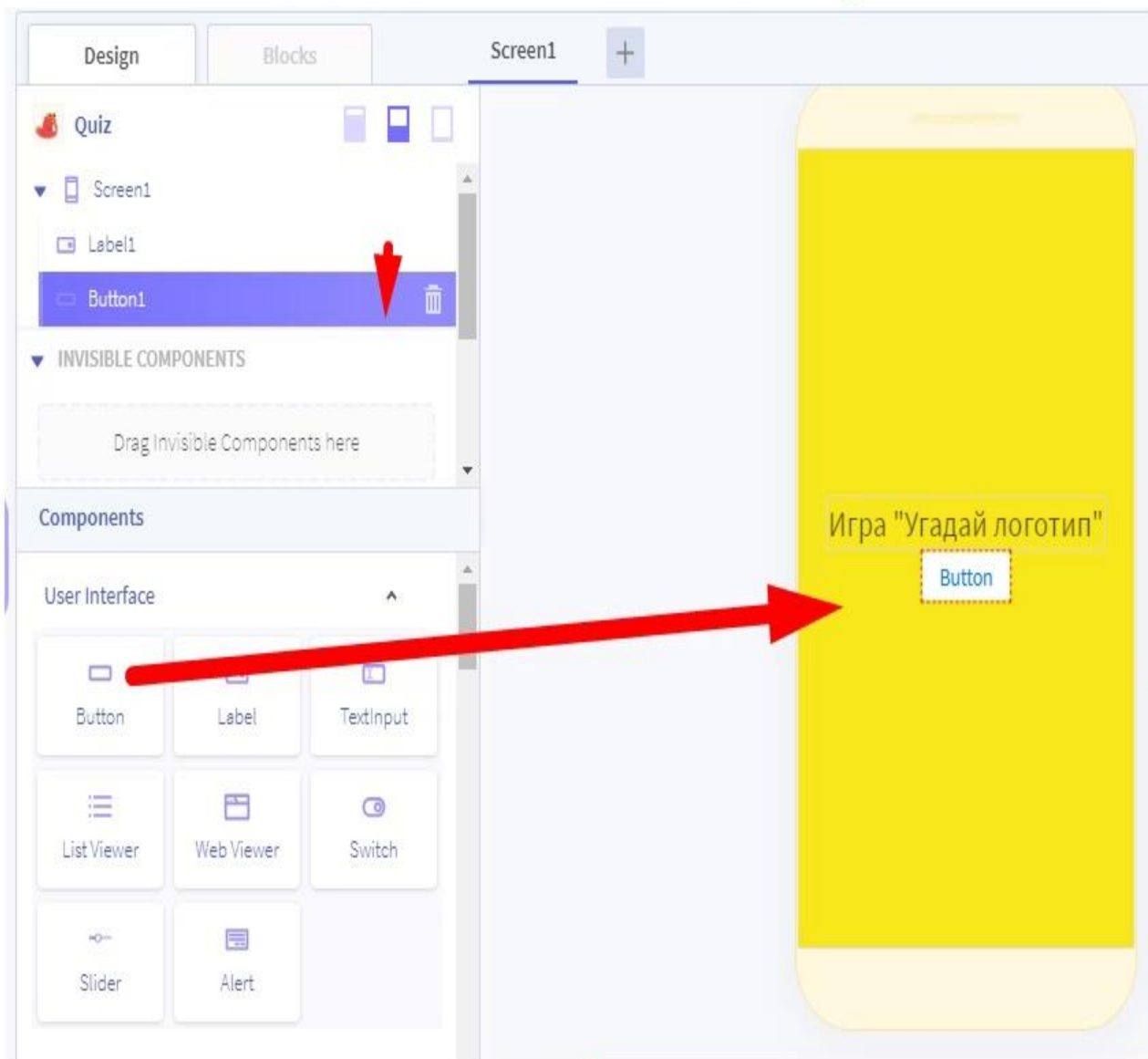
35

Width

Pick One: Fit contents, Fill container

A red arrow points to the '35' value in the Height section.

Добавим кнопку



Настроим кнопку



Button1

< Simple Adv >

Text

Играть! X

TextColor

#007aff

BackgroundColor

#ffffff

FontSize

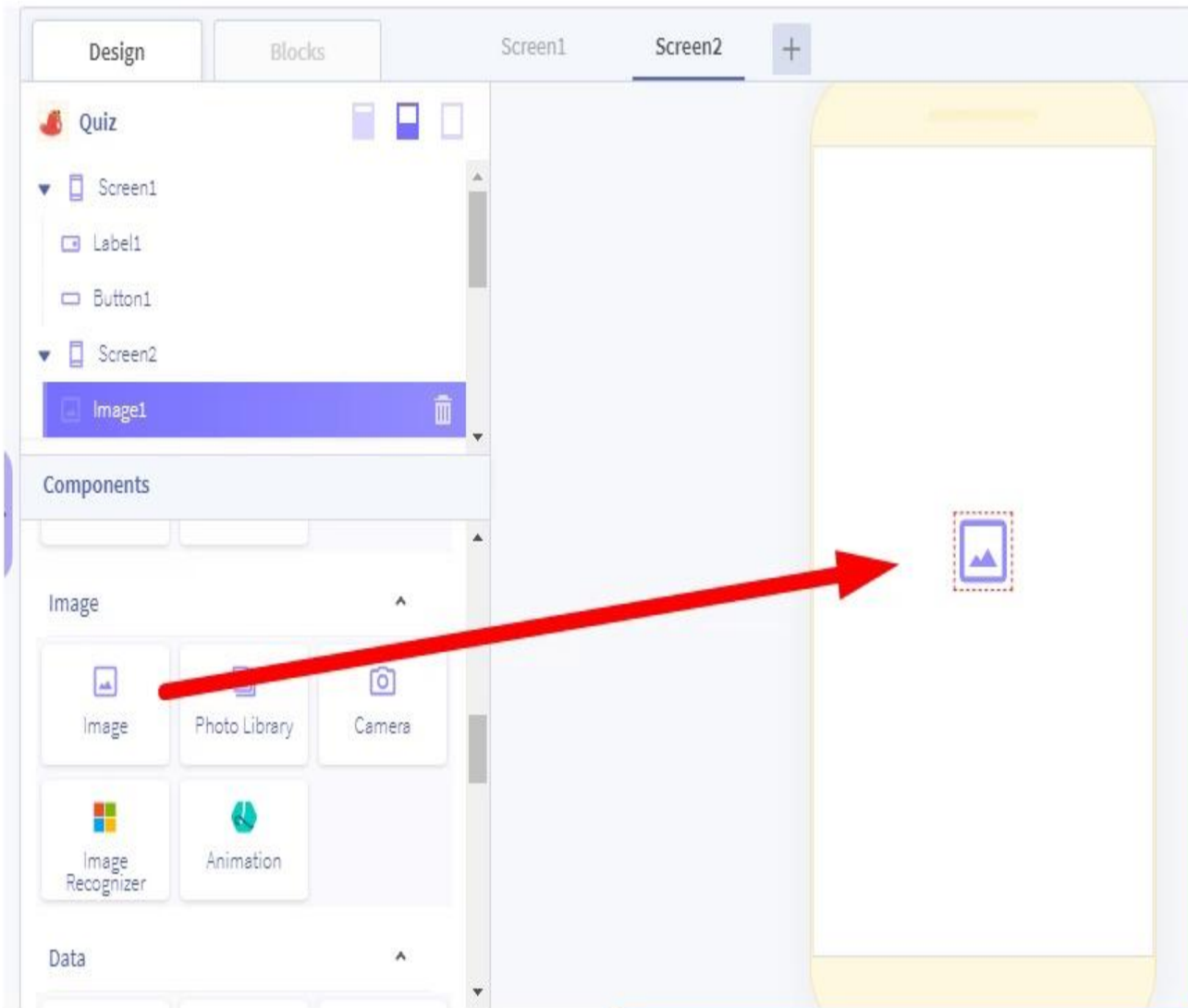
20

A settings panel for a button named "Button1". It has two tabs: "Simple" (selected) and "Adv". Under "Text", there is an input field containing "Играть!" and a close button "X". A red arrow points to this input field. Under "TextColor", there is a color picker set to "#007aff". Under "BackgroundColor", there is a color picker set to "#ffffff". Under "FontSize", there is an input field containing "20". A red arrow points to this input field.

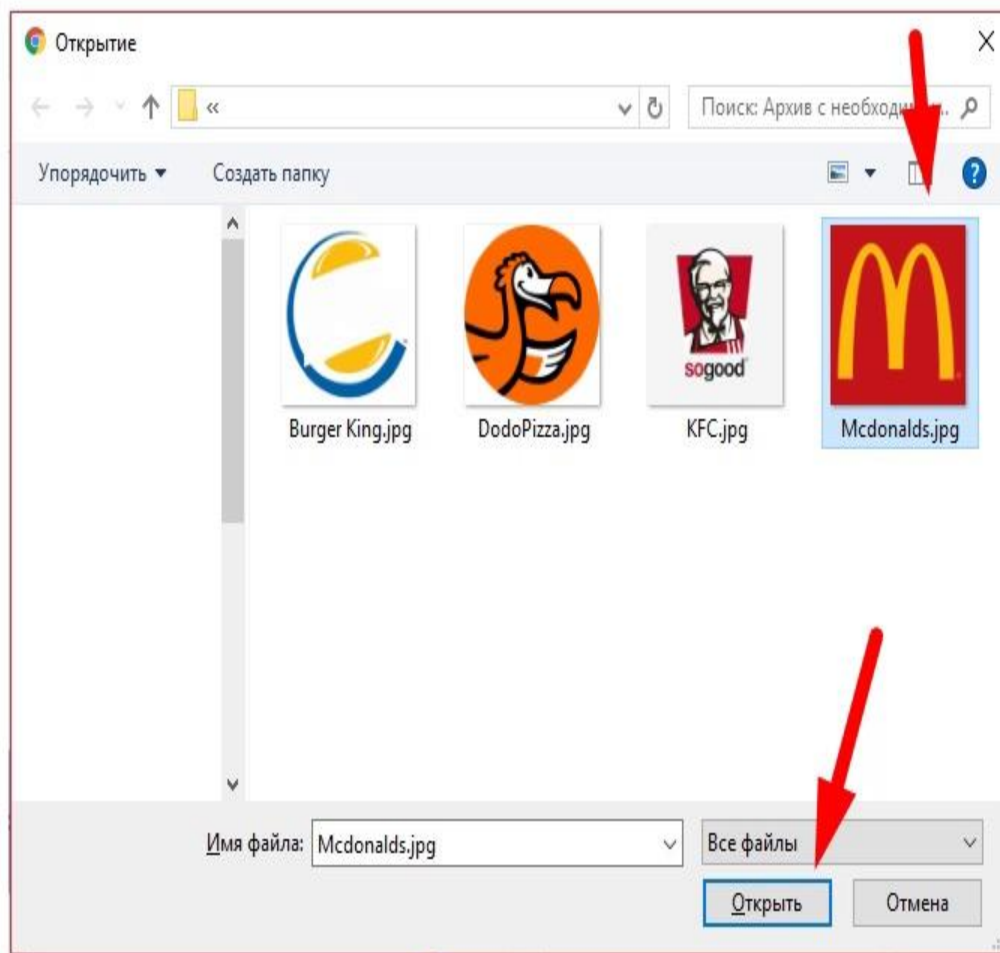
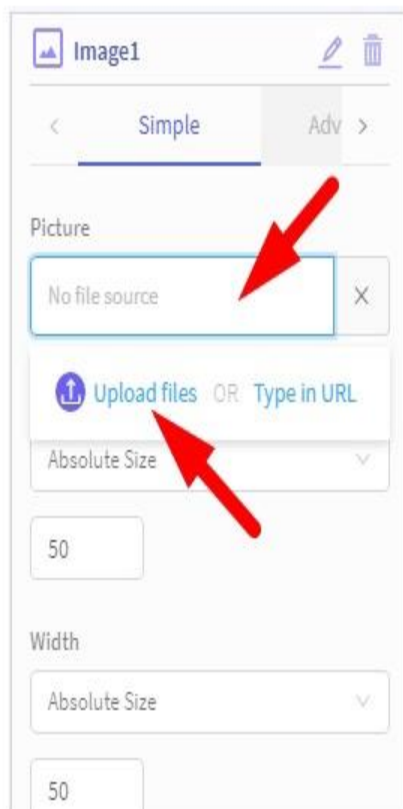
Создадим второй экран



Добавим изображение



Добавим изображение



Добавим изображение

Image1

< Simple Adv >

Picture

No file source

Upload files OR Type in URL

Mcdonalds.jpg

50

Picture

Mcdonalds.jpg

Height

Absolute Size

200

Width

Absolute Size

200

Image1

< Simple Adv >

Sizing >

Spacing >

Positioning

Position

relative

Overflow

Select option

EdgeOffsets

top bottom

px 50 px

left right

px # px

Добавим кнопки

The screenshot shows an IDE interface for designing a mobile application. At the top, there are tabs for 'Design', 'Blocks', 'Screen1', and 'Screen2'. The 'Design' tab is active, showing a project named 'Quiz' with a hierarchy of 'Screen2' containing 'Image1', 'Button4', 'Button3', and 'Button2'. A red arrow points from 'Button4' in the hierarchy to the 'Button' component in the 'User Interface' section of the 'Components' panel. The 'Components' panel lists various UI elements like Button, Label, TextInput, List Viewer, Web Viewer, Switch, Slider, and Alert. To the right, a mobile device preview displays the McDonald's logo and three buttons stacked vertically. A red arrow points from the 'Button' component in the 'User Interface' panel to the top button in the mobile preview.

Добавим изображение



Width

Absolute Size



120

Добавим действие

The screenshot displays a visual programming environment with a 'Design' view on the left and a 'Blocks' view on the right. The 'Design' view shows a list of categories: Control, Logic, Math, Text, Lists, Color, Objects, Variables, Functions, Timer1, Label1, Button1, and Screen1. The 'Button1' category is highlighted in blue. The 'Blocks' view shows a list of blocks for 'Button1': Click, LongClick, TouchDown, and TouchUp. A red arrow points from the 'Click' block in the 'Blocks' view to the 'Design' view, where a 'when Button1 Click' block is being added to the screen. Another red arrow points from the 'Click' block in the 'Blocks' view to the 'Click' block in the 'Design' view. A third red arrow points from the 'Button1' category in the 'Design' view to the 'Click' block in the 'Design' view. The 'Design' view also shows a 'when Button1 Click' block already present on the screen.

Добавим действие

The image shows a visual programming environment with a 'Blocks' panel on the left and a workspace on the right. The 'Control' category is selected in the 'Blocks' panel, and a 'when Button1 Click' block is attached to the workspace. A 'navigate to Screen2' block is being added to the 'do' slot of the 'when Button1 Click' block. Red arrows highlight the 'Control' category and the 'navigate to Screen2' block.

Design | Blocks | Screen1 | Screen2 | +

Control

Logic

Math

Text

Lists

Color

Objects

Variables

Functions

Label1

Button1

Screen1

if

do

navigate to Screen1

wait 1 seconds

repeat 10 times

do

when Button1 Click

do

navigate to Screen2

Добавим действие

The screenshot displays a development environment with a top navigation bar containing three tabs: "Screen1", "Screen2", and "Screen3". A red arrow points to "Screen2", and another red arrow points to a "+" button next to it. Below the navigation bar, three orange script blocks are visible, each starting with "when" and "do" blocks. The first block is for "Button4 Click", the second for "Button3 Click", and the third for "Button2 Click". Each "do" block contains a "navigate to Screen3" action.

Screen1 Screen2 Screen3 +

when Button4 Click
do navigate to Screen3

when Button3 Click
do navigate to Screen3

when Button2 Click
do navigate to Screen3

Тестируем приложение!

Добавим переменную

The screenshot displays a visual programming environment with a 'Blocks' palette on the left and a workspace on the right. The workspace is divided into five screens: Screen1, Screen2, Screen3, Screen4, and Screen5. A red arrow points from the 'initialize app variable name to' block in Screen1 to the 'initialize app variable Success to' block in Screen2. The 'Variables' category in the 'Blocks' palette is highlighted with a red arrow. The workspace contains the following blocks:

- Screen1:
 - initialize app variable name to
 - set app Success to
 - change app Success by 1
 - app Success
 - set app "name" to
 - change app "name" by 1
 - app "name"
- Screen2:
 - initialize app variable Success to
- Screen4:
 - when Button1 Click do navigate to Screen2

Добавим переменную

The screenshot displays a visual programming environment with a 'Design' view on the left and a 'Blocks' view on the right. The 'Design' view shows a list of categories: Control, Logic, Math, Text, Lists, Color, Objects, Variables, Functions, Label1, Button1, and Screen1. A red arrow points to the '0' value in the 'initialize app variable Success to' block in the 'Blocks' view. Another red arrow points from this '0' to the '0' value in the 'initialize app variable Success to' block in the 'Design' view. The 'Design' view also shows a 'when Button1 Click' block with a 'do' block containing 'navigate to Screen2'.

Design

Blocks

Screen1 Screen2 Screen3 Screen4 Screen5 +

Control

Logic

Math

Text

Lists

Color

Objects

Variables

Functions

Label1

Button1

Screen1

random integer from 1 to 100

random fraction

initialize app variable Success to 0

when Button1 Click

do

navigate to Screen2

Определим верную кнопку

The image shows a software development interface for designing a mobile application. The interface is divided into several sections:

- Design:** Shows the current screen, **Screen2**, which contains a large red square with the yellow McDonald's logo. Below the logo is a list of restaurant names: **McDonald's**, **Burger King**, and **KFC**. A red dashed box highlights the **McDonald's** text.
- Blocks:** A tree view on the left shows the hierarchy of the app. It includes **Quiz**, **Screen1** (with **Label1** and **Button1**), and **Screen2** (with **Image1** and **Button4**). A red arrow points to **Button4**.
- Components:** A panel at the bottom left shows various UI components available for use, including **Button**, **Label**, **TextInput**, **List Viewer**, **Web Viewer**, **Switch**, **Slider**, and **Alert**.

Изменим значение переменной

The screenshot displays a visual programming environment with a 'Blocks' palette on the left and a workspace on the right. The workspace is divided into five screens: Screen1, Screen2, Screen3, Screen4, and Screen5. A red arrow points from the 'set app Success to' block in Screen2 to the 'set app Success to' block in Screen3. Another red arrow points to the 'Variables' category in the 'Blocks' palette.

Blocks Palette:

- Control
- Logic
- Math
- Text
- Lists
- Color
- Objects
- Variables**
- Functions
- Image1
- Button4
- Button3
- Button2
- Screen2

Screen1:

- initialize app variable name to
- set app Success to
- change app Success by 1
- app Success
- set app "name" to
- change app "name" by 1
- app "name"

Screen2:

- when Button4 Click do navigate to Screen3 set app Success to

Screen3:

- when Button3 Click do navigate to Screen3
- when Button2 Click do navigate to Screen3

Изменим значение переменной

The image shows the Scratch interface. On the left is the 'Blocks' palette with categories: Control, Logic, Math, Text, Lists, Color, Objects, Variables, Functions, Timer1, and Image1. A red arrow points to the 'Math' category, which contains several blocks including a '0' block, a '1 + 1' block, a '1 - 1' block, a '1 x 1' block, and a '1 ÷ 1' block. A second red arrow points from the '1 + 1' block in the palette to a script area on the right. The script area is for 'Screen2' and contains three event-driven blocks: 'when Button4 Click', 'when Button3 Click', and 'when Button2 Click'. Each event block has a 'do' block containing 'navigate to Screen3'. The 'when Button4 Click' block also includes a 'set app Success to' block followed by the '1 + 1' block from the Math category.

Изменим значение переменной

The screenshot displays a visual programming environment with a 'Blocks' panel on the left and a workspace on the right. The workspace is currently set to 'Screen2'.

Blocks Panel (Left):

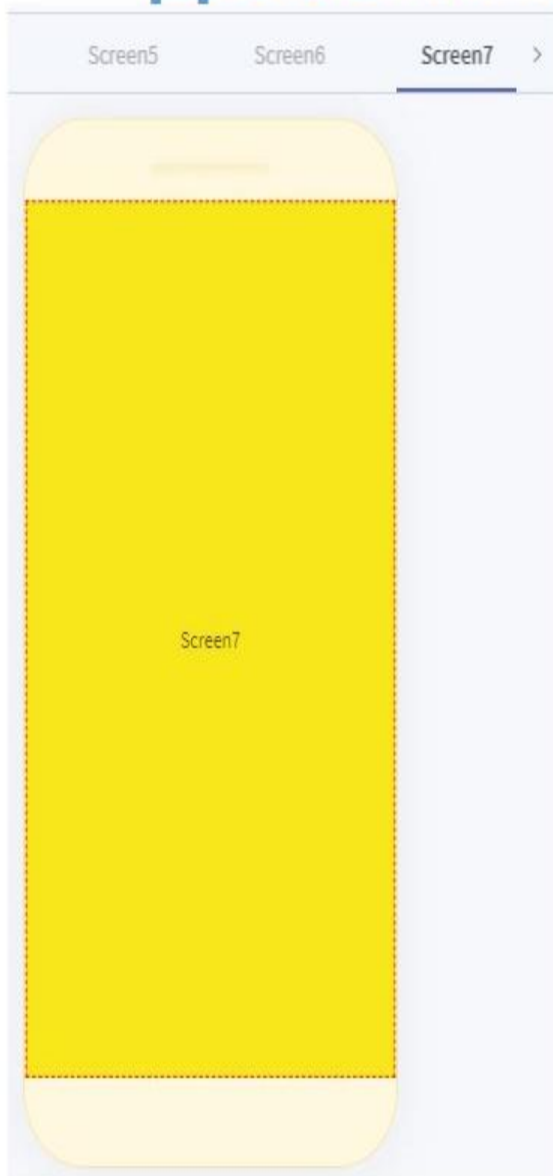
- Control:** initialize app variable name to
- Logic:** set app Success to
- Math:** change app Success by 1
- Text:** app Success
- Lists:**
- Color:**
- Objects:**
- Variables:** app Success

Workspace (Right):

- Screen1:** initialize app variable name to
- Screen2:**
 - when Button4 Click do:
 - navigate to Screen3
 - set app Success to app Success + 1
 - when Button3 Click do:
 - navigate to Screen3
- Screen3:**
- Screen4:**
- Screen5:**

Red arrows highlight the 'app Success' variable in the Variables panel and its use in the 'set app Success to' block within the Button4 Click event.

Добавим последний экран



Добавим надписи



Добавим действие

The screenshot shows the Scratch interface with the 'Blocks' palette open for 'Screen7'. The left sidebar has 'Screen7' selected. The palette contains several blocks:

- Control: when Screen7 Opens (do)
- Logic: when Screen7 Starts (do)
- Variables: from Screen7 set BackgroundColor to
- Functions: from Screen7 get BackgroundColor
- Screen7: from Screen7 set BackgroundPicture to
- Screen7: from Screen7 get BackgroundPicture

A red arrow points from the 'when Screen7 Opens' block in the palette to the same block on the stage. Another red arrow points from the 'Screen7' category in the sidebar to the 'from Screen7 get BackgroundPicture' block.

Изменим надпись

The image shows a visual programming environment with a 'Blocks' palette on the left and a workspace on the right. The workspace is titled 'Screen7' and contains a 'when Screen7 Opens' event block. Inside this event block is a 'do' block containing a 'from Label3 set Text to' block. A red arrow points from the 'set Text to' block in the palette to the 'do' block in the workspace. Another red arrow points from the 'set Text to' block in the palette to the 'do' block in the workspace.

The 'Blocks' palette on the left contains the following blocks:

- Control
- Logic
- Math
- Text
- Lists
- Color
- Objects
- Variables
- Functions
- Label2
- Label3**
- Screen7

The workspace contains the following blocks:

- when Screen7 Opens
- do
- from Label3 set Text to

Изменим надпись

The screenshot displays a programming environment with a 'Blocks' palette on the left and a script area on the right. The 'Text' category is selected in the palette, and a 'join' block is being dragged from it to a script block. The script area shows a sequence of blocks: 'Screen7 Opens', 'from Label3 set Text to', and a 'join' block containing 'hello' and 'world'. A red arrow points from the 'join' block in the palette to the 'join' block in the script. Another red arrow points from the 'join' block in the script to the 'set Text to' block. The 'Design' tab is active, and the current screen is 'Screen7'.

Design | Blocks | Screen3 | Screen4 | Screen5 | Screen6 | Screen7 | +

Control

Logic

Math

Text

Lists

Color

Objects

Variables

Functions

Label2

Label3

Screen7

Screen7 Opens

from Label3 set Text to

join

hello

world

join

hello

world

Изменим надпись

The screenshot displays the Scratch IDE interface for editing a script on Screen7. The 'Blocks' palette on the left is open, showing the 'Variables' category selected. A red arrow points from the 'app Success' variable block in the palette to its use in a script on the stage.

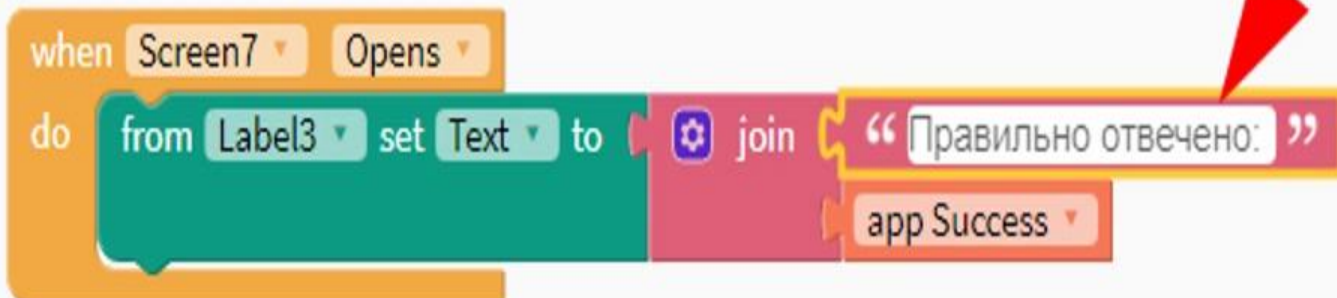
The script on the stage is as follows:

```
when Screen7 Opens  
do  
  from Label3 set Text to join "hello" app Success
```

The 'do' block contains a 'from Label3 set Text to' block followed by a 'join' block. The 'join' block has two inputs: a text block containing "hello" and the 'app Success' variable block.

Изменим надпись

```
when Screen7 Opens  
do  
  from Label3 set Text to join  
  “ Правильно отвечено: ”  
  app Success
```



Тестируем приложение!

Добавим таймер

The screenshot shows the Xcode interface for a quiz application. The main canvas displays a yellow mobile screen with the text "Игра 'Угадай логотип'" and a button labeled "Играть!". A red arrow points from the "Timer" component in the "Sensors" section of the Components palette to the bottom of the mobile screen. A dashed box at the bottom of the screen indicates the location of the "Timer1" component.

The screenshot shows the settings for the "Timer1" component. The "Simple" tab is selected. The "Enabled" toggle is set to "false", the "IntervalMilliseconds" is set to "1000", and the "Loops" toggle is set to "true". Red arrows point to each of these settings.

Timer1

Simple Advanced

Enabled
 false

IntervalMilliseconds
1000

Loops
 true

Добавим переменную

The screenshot shows the Scratch IDE interface with the 'Blocks' palette on the left and the 'Screen1' script area on the right. A red arrow points from the 'Variables' category in the palette to the 'initialize app variable Success to 0' block in the script area. Another red arrow points from the 'initialize app variable name to' block in the palette to the 'initialize app variable Time to 0' block in the script area. The script area contains the following blocks:

- initialize app variable Success to 0
- initialize app variable Time to 0
- when Button1 Click do
navigate to Screen2

The 'Variables' category in the palette is highlighted, and the 'initialize app variable Success to 0' block is selected.

Активируем таймер

The screenshot displays a visual programming interface with a 'Blocks' panel on the left and a workspace on the right. The workspace is divided into two sections: initialization and a button click event.

Initialization Section:

- initialize app variable Success to 0
- initialize app variable Time to 0

Button Click Event Section:

- when Button1 Click
- do
 - navigate to Screen2
 - from Timer1 set Enabled to true

Timer1 Block List (Left Panel):

- when Timer1 Fires
- do
 - from Timer1 set Enabled to true
 - from Timer1 get Enabled
 - from Timer1 set IntervalMilliseconds to 3000
 - from Timer1 get IntervalMilliseconds
 - from Timer1 set Loops to true
 - from Timer1 get Loops

A red arrow points from the 'from Timer1 set Enabled to true' block in the 'do' section of the 'when Button1 Click' event to the 'from Timer1 set Enabled to true' block in the 'do' section of the 'when Timer1 Fires' event.

Считаем время

The screenshot displays the Kodular IDE interface for editing Screen2. The left sidebar shows a list of components, with 'Timer1' selected under the 'Function' category. The main workspace is divided into two panels: 'Design' and 'Blocks'.

Design Panel: Shows the visual layout of Screen2 with several buttons (Button2, Button3, Button4) and a timer component (Timer1).

Blocks Panel: Contains the following logic blocks:

- when Timer1 Fires** (yellow block):
 - do **navigate to Screen3** (orange block)
 - do **change app Success by 1** (red block)
- from Timer1 set Enabled to true** (green block)
- from Timer1 get Enabled** (green block)
- from Timer1 set IntervalMilliseconds to 3000** (green block)
- from Timer1 get IntervalMilliseconds** (green block)
- from Timer1 set Loops to true** (green block)
- from Timer1 get Loops** (green block)

Two red arrows point from the 'when Timer1 Fires' block to the 'when Button4 Click' and 'when Timer1 Fires' blocks in the right panel, indicating a sequence of events.

Right Panel (Screen2): Contains the following logic blocks:

- when Button4 Click** (orange block):
 - do **navigate to Screen3** (orange block)
 - do **change app Success by 1** (red block)
- when Button3 Click** (orange block):
 - do **navigate to Screen3** (orange block)
- when Button2 Click** (orange block):
 - do **navigate to Screen3** (orange block)
- when Timer1 Fires** (yellow block):
 - do (empty block)

Считаем время



```
when Timer1 Fires  
do set app Time to app Time + 1
```

The image shows a Scratch code block. It is an orange 'when' block with 'Timer1' selected in the dropdown and 'Fires' in the event type dropdown. Below it is a 'do' block containing a 'set' block. The 'set' block has 'app Time' in the dropdown and 'to' as the operator. The value being set is 'app Time + 1', where 'app Time' is in a dropdown, '+' is the operator, and '1' is the value.

Задание:

Добавьте подсчет времени на все остальные экраны



Добавим надпись



Остановим таймер

The screenshot displays a visual programming environment with a 'Blocks' palette on the left and a workspace on the right. The workspace shows a sequence of events for 'Screen7' opening.

Blocks Palette (Left):

- Control
 - when Timer1 Fires
- Logic
 - do
- Text
 - from Timer1 set Enabled to true
- Lists
 - from Timer1 get Enabled
- Color
 - from Timer1 set IntervalMilliseconds to 3000
- Variables
 - from Timer1 get IntervalMilliseconds
- Function
 - from Timer1 set Loops to true
- Timer1
 - from Timer1 get Loops

Workspace (Right):

- when Screen7 Opens
 - do
 - from Label3 set Text to join " Правильно отвечено: "
 - app Success
 - from Timer1 set Enabled to false

Red arrows point to the 'Timer1' category in the palette, the 'when Screen7 Opens' event block, and the 'from Timer1 set Enabled to false' block.

Изменим надпись

```
when Screen7 Opens  
do  
  from Label3 set Text to join " Правильно отвечено: "  
  from Timer1 set Enabled to false  
  from Label4 set Text to join " Затрачено времени: "
```


Тестируем приложение!

