

## Практическая работа 10

### Разработка теста

Для начала напишем код:

```
import flet as ft

class Question:
    def __init__(self, text, options, correct_index):
        self.text = text
        self.options = options
        self.correct_index = correct_index

class QuizApp:
    def __init__(self, page: ft.Page):
        self.page = page
        self.page.title = "Угадай стандарт"
        self.page.window.width = 600
        self.page.window.height = 500

        self.questions = self.create_questions()
        self.current_question = 0
        self.correct_answers = 0
        self.answered = False

        self.setup_ui()

    def create_questions(self):
        return [
            Question(
                "Какой стандарт определяет представление чисел с плавающей точкой?", [
                    "ASCII", "IEEE 754", "Unicode", "ISO 9001"],
                1
            ),
            Question(
                "Какой стандарт используется для кодирования символов всех письменностей мира?", [
                    "ASCII", "IEEE 802.11", "Unicode", "IEEE 754"],
                2
            ),
            Question(
                "Какой стандарт относится к беспроводной связи (Wi-Fi)?", [
                    "IEEE 802.11", "ISO 9001", "ASCII", "IEEE 754"],
                0
            )
        ]
```

```
        ),
        Question(
            "Какой стандарт определяет требования к системе менеджмента
качества?",  

            ["IEEE 754", "Unicode", "IEEE 802.11", "ISO 9001"],  

            3
        ),
        Question(
            "Какой стандарт является американским кодом для обмена
информацией?",  

            ["Unicode", "ASCII", "ISO 9001", "IEEE 802.11"],  

            1
        )
    ]
]
```

```
def setup_ui(self):
    self.question_number = ft.Text("", size=20, weight=ft.FontWeight.BOLD)
    self.question_text = ft.Text("", size=16)

    self.radio_group = ft.RadioGroup(
        content=ft.Column([])
    )

    self.result_text = ft.Text("", size=14, color="green")

    self.progress_text = ft.Text("Правильных ответов: 0", size=14)

    self.next_btn = ft.ElevatedButton("Далее", on_click=self.next_question,
width=200)

    self.restart_btn = ft.ElevatedButton("Пройти заново",
on_click=self.restart_quiz, width=200, visible=False)
    self.finish_btn = ft.ElevatedButton("Завершить", on_click=self.finish_quiz,
width=200, visible=False)

    self.quiz_container = ft.Container(
        content=ft.Column([
            self.question_number,
            ft.Divider(),
            self.question_text,
            ft.Container(height=20),
            self.radio_group,
            ft.Container(height=20),
            self.result_text,
            ft.Container(height=20),

```

```

        self.progress_text,
        ft.Container(height=20),
        ft.Row([self.next_btn, self.restart_btn, self.finish_btn],
alignment=ft.MainAxisAlignment.CENTER)
    ],
padding=20
)

self.page.add(self.quiz_container)
self.show_question()

def show_question(self):
if self.current_question < len(self.questions):
    q = self.questions[self.current_question]
    self.question_number.value = f'Вопрос {self.current_question + 1} из {len(self.questions)}'
    self.question_text.value = q.text

    self.radio_group.content = ft.Column([
        ft.Radio(value=str(i), label=opt) for i, opt in enumerate(q.options)
    ])
    self.radio_group.value = None

    self.result_text.value = ""
    self.answered = False
    self.next_btn.visible = True
    self.restart_btn.visible = False
    self.finish_btn.visible = False

    self.page.update()

def next_question(self, e):
if self.radio_group.value is None:
    self.result_text.value = "Выберите ответ!"
    self.result_text.color = "red"
    self.page.update()
    return

if not self.answered:
    q = self.questions[self.current_question]
    selected = int(self.radio_group.value)

    if selected == q.correct_index:
        self.correct_answers += 1
        self.result_text.value = "Правильно!"

```

```
        self.result_text.color = "green"
    else:
        self.result_text.value = f"Неправильно! Правильный ответ:
{q.options[q.correct_index]}"
        self.result_text.color = "red"

    self.progress_text.value = f"Правильных ответов: {self.correct_answers}"
    self.answered = True

    for radio in self.radio_group.content.controls:
        radio.disabled = True

    self.page.update()
else:
    self.current_question += 1
    if self.current_question < len(self.questions):
        self.show_question()
    else:
        self.show_results()

def show_results(self):
    percentage = (self.correct_answers / len(self.questions)) * 100

    self.question_number.value = "Результат теста"
    self.question_text.value = f"Вы ответили правильно на {self.correct_answers} из {len(self.questions)} вопросов."
    self.radio_group.content = ft.Column([
        ft.Text(f"Процент правильных ответов: {percentage:.0f}%", size=18,
weight=ft.FontWeight.BOLD)
    ])

    self.result_text.value = ""
    self.progress_text.value = ""

    self.next_btn.visible = False
    self.restart_btn.visible = True
    self.finish_btn.visible = True

    self.page.update()

def restart_quiz(self, e):
    self.current_question = 0
    self.correct_answers = 0
    self.answered = False
```

```
self.progress_text.value = "Правильных ответов: 0"
self.show_question()

def finish_quiz(self, e):
    self.page.window.close()

def main(page: ft.Page):
    QuizApp(page)

ft.app(target=main)
```

Демонстрация работы показана на рисунке 1:

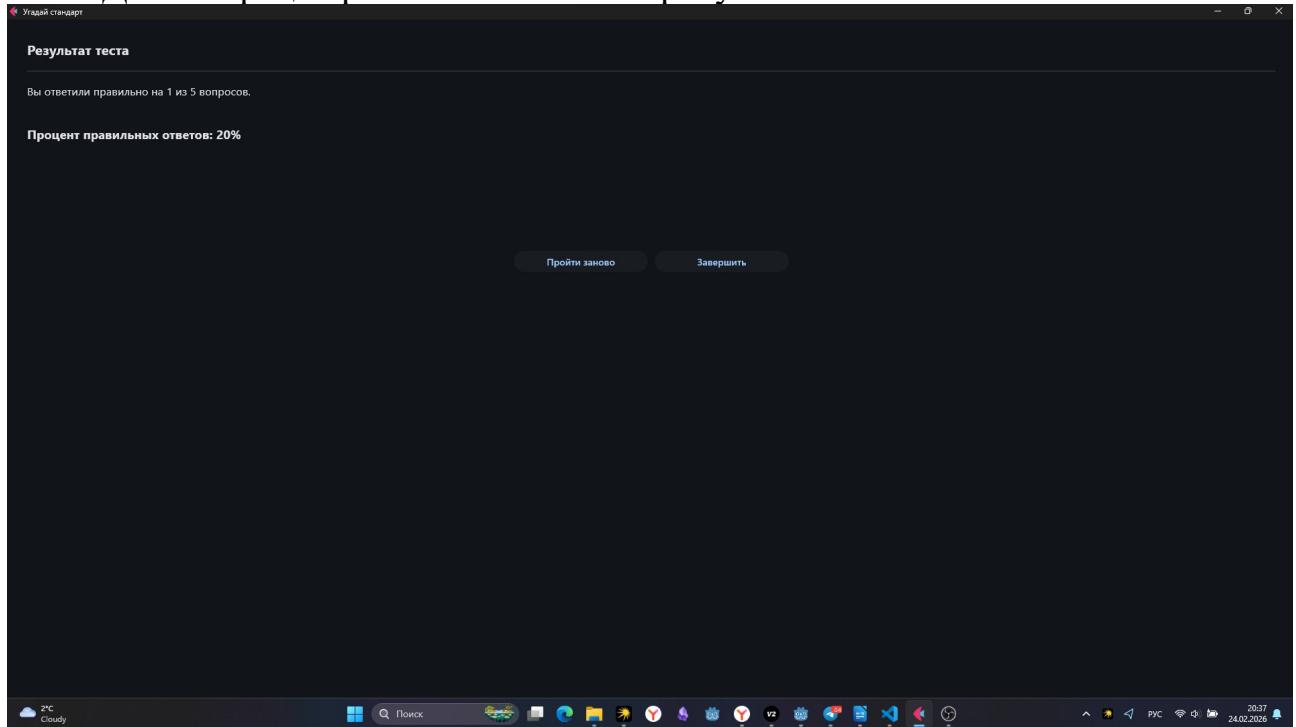


Рисунок 1 — Демонстрация работы