

# --- Função para rodar um quiz ---

def quiz(perguntas):

score = 0

for i, p in enumerate(perguntas, 1):

print(f"\nPergunta {i}: {p['pergunta']}")

for op in p['opcoes']:

print(op)

resposta = input("Sua resposta: ").strip().upper()

if resposta == p['correta']:

print("✅ Correto!")

score += 1

else: print(f"❌ Errado! Resposta correta: {p['correta']}")

print(f"\nPontuação final: {score}/{len(perguntas)}")

**Explicação linha a linha**

Segue uma explicação detalhada, linha a linha do código:

# --- Função para rodar um quiz ---

def quiz(perguntas):

* Define uma função chamada quiz que recebe um argumento perguntas.
* perguntas espera ser um iterável (normalmente uma lista) de dicionários com informação sobre cada pergunta (ver exemplo no final).

score = 0

* Cria e inicializa a variável score com 0. Ela guarda a contagem de respostas corretas.

for i, p in enumerate(perguntas, 1):

* Laço for que percorre perguntas.
* enumerate(perguntas, 1) retorna pares (índice, elemento) começando o índice em 1 (em vez de 0).
* i é o número da pergunta (1, 2, 3, ...). p é o item atual (espera-se que seja um dicionário representando a pergunta).

print(f"\nPergunta {i}: {p['pergunta']}")

* Imprime uma linha com número da pergunta e o texto da pergunta.
* f"...{...}..." é um *f-string* (interpolação). \n adiciona uma linha em branco antes da pergunta.
* p['pergunta'] acessa o valor da chave 'pergunta' no dicionário p. (Se essa chave não existir, ocorrerá KeyError.)

for op in p['opcoes']:

* Outro for aninhado que percorre p['opcoes'] — espera-se que seja uma lista/iterável com as opções de resposta (strings).

print(op)

* Imprime cada opção (por exemplo "A) ...", "B) ...") em sua própria linha.

resposta = input("Sua resposta: ").strip().upper()

* input("Sua resposta: ") exibe o prompt e lê a entrada do usuário como string (bloqueante).
* .strip() remove espaços em branco do começo/fim (útil se o usuário digitar " A ").
* .upper() converte a string para letras maiúsculas — facilita comparação sem se preocupar com maiúsc/minúsc.
* O resultado é armazenado em resposta.

if resposta == p['correta']:

* Compara a resposta do usuário com o valor esperado p['correta'].
* Atenção: essa comparação assume que p['correta'] já está no formato compatível (por exemplo 'A' em maiúscula). Se p['correta'] puder estar em minúsculas, uma versão segura seria: if resposta == p['correta'].strip().upper():

print("✅ Correto!")

* Mostra uma mensagem indicando acerto (tem emoji, que funciona em terminais que suportam).

score += 1

* Incrementa score em 1 (soma 1 ao contador de acertos).

else:

* Bloco executado quando a condição if é falsa (resposta errada).

print(f"❌ Errado! Resposta correta: {p['correta']}")

* Informa que a resposta está errada e mostra a resposta correta (ex.: A). Mostra o valor como está em p['correta'].

print(f"\nPontuação final: {score}/{len(perguntas)}")

* Após o laço (terminado), imprime a pontuação final.
* len(perguntas) dá o número total de perguntas para formar score/total.

Exemplo de quiz:

## Exemplo de estrutura de perguntas

**perguntas = [**

**{**

**'pergunta': 'Qual a capital do Brasil?',**

**'opcoes': ['A) Brasília', 'B) Rio de Janeiro', 'C) São Paulo'],**

**'correta': 'A'**

**},**

**{**

**'pergunta': '2 + 2 = ?',**

**'opcoes': ['A) 3', 'B) 4', 'C) 5'],**

**'correta': 'B'**

**}**

**]**