

Задача 1: Список покупок

Задание: У вас есть список продуктов, которые нужно купить. Напишите программу, которая выводит этот список в виде пронумерованного перечня.

Начальный код:

```
shopping_list = ["milk", "bread", "eggs", "butter", "apples"]  
  
print("Список покупок:")  
  
# Ваш код здесь  
  
# Используйте цикл для перебора каждого элемента в shopping_list  
  
# Внутри цикла выводите номер и название продукта
```

Решение:

```
shopping_list = ["milk", "bread", "eggs", "butter", "apples"]  
print("Список покупок:")  
  
for index, item in enumerate(shopping_list):  
    print(f"{index + 1}. {item}")
```

Задача 2: Обратный отсчет до запуска

Задание: Напишите программу, которая имитирует обратный отсчет перед запуском ракеты. Программа должна запросить у пользователя число (например, 5), а затем вывести отсчет от этого числа до 1, после чего напечатать "Go!".

Решение:

```
def countdown():  
    try:  
        start_number = int(input("Введите число для начала обратного отсчета: "))  
        if start_number <= 0:  
            print("Ошибка: Число должно быть больше 0.")  
            return  
        for i in range(start_number, 0, -1):  
            print(i)  
        print("Go!")  
    except ValueError:  
        print("Ошибка: Пожалуйста, введите целое число.")  
  
if __name__ == "__main__":  
    countdown()
```

Задача 3: Угадай число

Задание: Напишите простую игру. Программа "загадывает" число от 1 до 10. Пользователь должен угадать это число. Цикл `while` должен продолжаться до тех пор, пока пользователь не введет правильный ответ.

Начальный код:

```
secret_number = 7

# Ваш код здесь

# Создайте цикл, который работает, пока ввод пользователя не будет равен
'secret_number'

# После того как пользователь угадает, выведите любое сообщение
```

Решение:

```
secret_number = 7
while True:
    try:
        guess = int(input("Угадайте число от 1 до 10: "))

        if guess == secret_number:
            print("Поздравляю, вы угадали число!")
            break # Выходим из цикла, когда пользователь угадал
        elif guess < 1 or guess > 10:
            print("Пожалуйста, введите число от 1 до 10.")
        else:
            print("Неверно. Попробуйте еще раз.")

    except ValueError:
        print("Ошибка: Пожалуйста, введите целое число.")
```

Задача 4: Обработка данных

Задание: У вас есть список с баллами студентов. Вам нужно посчитать сумму баллов, но с особыми правилами:

- Если балл отрицательный (< 0), его нужно проигнорировать и перейти к следующему.
- Если встречается балл 0, это сигнал, что список данных обрывается, и нужно немедленно прекратить подсчет.
- Если цикл прошел через все баллы и не был прерван (т.е. не встретил 0), нужно вывести сообщение, что все данные были обработаны корректно.

Используйте `break`, `continue` или `else` для выполнения условий.

Начальный код:

```
scores = [75, 88, -10, 95, 100, -25, 89]
```

```
total_score = 0
```

```
for score in scores:
```

```
    # Ваш код здесь
```

```
    # Проверка, если балл < 0
```

```
    # Проверка, если балл == 0
```

```
    # В остальных случаях добавляйте балл к total_score print(f"Добавлен балл: {score}")
```

```
# если цикл не был прерван, выводим сообщение "Все данные обработаны"
```

```
print(f"\nИтоговая сумма баллов: {total_score}")
```

Решение:

```
scores = [75, 88, -10, 95, 100, -25, 89]
```

```
total_score = 0
```

```
for score in scores:
```

```
    if score < 0:
```

```
        continue # Пропускаем отрицательные баллы
```

```
    if score == 0:
```

```
        print("Обработка прервана.")
```

```
        break # Прерываем цикл при встрече нуля
```

```
    total_score += score
```

```
    print(f"Добавлен балл: {score}")
```

```
else:
```

```
    print("Все данные успешно обработаны.") #Выводим сообщение только при успешном  
    завершении цикла без прерывания
```

```
print(f"Итоговая сумма баллов: {total_score}")
```

Задача 5: Рисуем Прямоугольник (optional)

Задание: Напишите программу, которая запрашивает у пользователя два числа: высоту и ширину. Затем программа должна "нарисовать" в консоли прямоугольник из звездочек (*) указанного размера.

Подсказка:

- Внешний цикл будет отвечать за количество строк (высоту).
- Внутренний цикл будет в каждой строке выводить нужное количество звездочек (ширину).

Решение:

```
def draw_rectangle():
```

```
    try:
```

```
        height = int(input("Введите высоту прямоугольника: "))
```

```
        width = int(input("Введите ширину прямоугольника: "))
```

```
    if height <= 0 or width <= 0:
```

```
        print("Ошибка: Высота и ширина должны быть больше 0.")
```

```
        return
```

```
    for i in range(height): # строки
```

```
        for j in range(width): # звездочки в строке
```

```
            print("*", end="") # выводим звездочку без переноса строки
```

```
        print() # переходим на новую строку после каждой строки звездочек
```

```
    except ValueError:
```

```
        print("Ошибка: Пожалуйста, введите целые числа для высоты и ширины.")
```

```
if __name__ == "__main__":
```

```
    draw_rectangle()
```