**Тема 1-2-3: Условия**

**Задания:**

1. Синтаксис условного оператора.
2. Вложенные условные инструкции.
3. Операторы сравнения.
4. Типы данных, используемые в условных выражениях.
5. Логические операторы.
6. Каскадные условные инструкции.

**Методические указания для выполнения заданий:**

Условные операторы позволяют программе выполнять разные действия в зависимости от соблюдения или несоблюдения определённых условий. Основной оператор условия в Python — это if. Условие указывается после оператора if, за которым ставится двоеточие, а затем следуют инструкции, которые выполняются, если условие истинно.

Если условие ложно, то используется блок else, который также оканчивается двоеточием. Этот блок содержит инструкции, выполняемые, если условие в if оказалось ложным.

**Синтаксис условного оператора в Python:**

python

Копировать код

if Условие:

Блок инструкций 1

else:

Блок инструкций 2

Блок инструкций 1 выполняется, если условие истинно. Если условие ложно, выполняется блок инструкций 2.

**1. Синтаксис условного оператора**

Пример программы, которая проверяет, является ли число положительным или отрицательным:

python

Копировать код

x = int(input("Введите число: "))

if x > 0:

print(f"{x} - положительное число")

else:

print(f"{x} - отрицательное число")

В этом примере проверяется условие x > 0. Если это условие истинно, программа выводит сообщение о том, что число положительное. Если же оно ложно (например, при вводе отрицательного числа), выполняется блок else, который выводит сообщение об отрицательном числе.

**2. Вложенные условные инструкции**

Иногда необходимо проверить несколько условий. Для этого можно использовать вложенные условные операторы.

Пример:

python

Копировать код

x = int(input("Введите число: "))

if x > 0:

print(f"{x} - положительное число")

else:

if x == 0:

print(f"{x} - это ноль")

else:

print(f"{x} - отрицательное число")

Здесь, если x не больше 0, выполняется дополнительная проверка: равно ли x нулю. В зависимости от результата выводится соответствующее сообщение.

**3. Операторы сравнения**

В Python для сравнения значений используются следующие операторы:

* ==: равно
* !=: не равно
* <: меньше
* >: больше
* <=: меньше или равно
* >=: больше или равно

Пример:

python

Копировать код

a = 10

b = 20

if a == b:

print("a равно b")

else:

print("a не равно b")

**4. Типы данных, используемые в условиях**

В условиях можно использовать различные типы данных, такие как целые числа, строки, логические значения.

Пример:

python

Копировать код

name = input("Введите имя: ")

if name == "Али":

print("Привет, Али!")

else:

print("Привет, незнакомец!")

**5. Логические операторы**

Для построения более сложных условий в Python используются логические операторы:

* and: возвращает True, если оба условия истинны.
* or: возвращает True, если хотя бы одно из условий истинно.
* not: инвертирует значение условия.

Пример:

python

Копировать код

age = int(input("Введите ваш возраст: "))

if age > 18 and age < 60:

print("Вы трудоспособного возраста")

else:

print("Вы не трудоспособного возраста")

Здесь проверяется, принадлежит ли возраст к диапазону от 19 до 59 лет включительно.

**6. Каскадные условные инструкции**

Когда есть много условий, которые нужно проверить, можно использовать каскадные условные инструкции с оператором elif (сокращение от "else if").

Пример:

python

Копировать код

score = int(input("Введите вашу оценку: "))

if score >= 90:

print("Отлично!")

elif score >= 70:

print("Хорошо!")

elif score >= 50:

print("Удовлетворительно")

else:

print("Неудовлетворительно")

Здесь проверяются различные диапазоны оценок и выводится соответствующее сообщение в зависимости от значения переменной score.

**Вывод:**

Условные операторы являются мощным инструментом в программировании и позволяют программе принимать решения на основе различных условий. Правильное использование этих операторов делает программу гибкой и адаптивной к различным ситуациям.