МИНИСТЕРСТВО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ И ЗАНЯТОСТИ НАСЕЛЕНИЯ ПРИМОРСКОГО КРАЯ

КРАЕВОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ   
ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

«НАХОДКИНСКИЙ ГОСУДАРСТВЕННЫЙ

ГУМАНИТАРНО-ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

|  |  |  |
| --- | --- | --- |
|  |  |  |

Дисциплина: Основы алгоритмизации и программирования.

Специальность:09.02.07 Информационные системы и программирование (Разработчик веб и мультимедийных приложений)

**ЗАДАНИЕ**

Переменные, как оптимизатор работы в python.

|  |  |  |
| --- | --- | --- |
| Студенты группы 721Веб/Б |  | С.Ю. Мереняну |
|  |  |  |
|  |  |  |
| Руководитель |  | Б.М. Дияров |
|  |  |  |

г. Находка, 2023

**Содержание**

1. Переменная3

2. Типы переменных3

2.1. Статическая4

2.2. Динамическая4

3. Переменные, как оптимизатор работы в python5

3.1. Хранение промежуточных значений5

3.2. Подмена повторяющихся значений5

3.3. Счётчик5

7. Заключение7

8. Список использованных источников8

1. **Переменная**

Переменная — это простейшая именованная структура данных, в которой может быть сохранён промежуточный или конечный результат работы программы. Переменную в Python создать очень просто - нужно присвоить некоторому идентификатору значение при помощи оператора присваивания «=».

Пример кода:

а = 10  
b = 3.1415926  
c = "Hello"  
d = [1, 2, 3]

Как можно заметить, в примере используют четыре типа переменных:

* переменная а хранит значение типа int (целое число),
* переменная b - типа float (действительное число),
* переменная c - типа str (строка),
* переменная d - типа list (список, в данном случае из трех целых чисел)

На самом деле переменная в python является лишь ссылкой на объект в памяти. При создании любой переменной (число, строка или массив) в неё записывается ссылка на объект, а сам объект находится где-то в оперативной памяти далеко от самой переменной со ссылкой. Таким образом, несколько переменных могут указывать на один объект, и при изменении объекта (например, списка) изменится результат обращения к нему с использованием каждой переменной.

1. **Типы переменных**

Компьютерные программы работают с разнообразными данными — числами, словами (символами), структурами (массивами, деревьями) и т. д. Соответственно, приложение должно уметь их различать, а также понимать, где они хранятся и в какой момент их необходимо задействовать. Поэтому в программировании используются следующие виды переменных:

**2.1. Статическая**

В этом случае тип данных переменной определяется на этапе компиляции программы — то есть заранее задается программистом. Это позволяет в большинстве случаев сократить расходы ресурсов компьютера. Кроме того, статическая типизация переменных упрощает обнаружение ошибок, что снижает время разработки и повышает надежность программы. В то же время она делает программный код менее гибким, так как любые изменения в типе данных программисту придется указывать вручную.

**2.2. Динамическая**

При такой типизации тип переменной определяется на этапе присвоения значения (то есть исполнения программы), а не его объявления. Благодаря этому в различных участках программного кода одна и та же переменная может принимать значения разных типов. Динамическая типизация делает программу более гибкой, позволяет ей лучше взаимодействовать с изменяющимся окружением, упрощает ее переход на новый формат данных. В то же время повышается вероятность ошибок, увеличивается расход ресурсов платформы при исполнении.

Для каждой переменной устанавливается своя область видимости — часть программы, в пределах которой ее идентификатор (имя) остается с ней связанным, то есть позволяет через себя обратиться к ее значению. В зависимости от этого переменные могут быть:

локальными — задействуются только конкретной подпрограммой (фрагментом кода);

глобальными — используются всей программой.

1. **Переменные, как оптимизатор работы в python**
   1. **Хранение промежуточных значений**

Одной из основных «фишек» переменной является возможность передавать в переменную новые значения различными способами. К примеру:

b = input()  
>>>5  
b = b+5  
print(b)  
>>>10

На данном примере видно, как я использую одну и ту же переменную, экономы выделяемую память.

* 1. **Подмена повторяющихся значений**

К примеру, у вас есть определённая шаблонная форма, в которой энное количество раз повторяется имя одного и того же человека. Дабы сократить время на редактирование каждого упоминания, можно заменить их переменной, а в начале выполнения программы спрашивать её значение:

name = str(input("Введите ваше имя: "))

# проверка на наличие этого имени в БД. Введённое имя уже найдено

print("Уважаемый", name, "!\n",

"Учётная запись под именем", name, "уже существует :(\n"

"Попробуйте ввести другой вариант.")

В данном примере использован вариант с всего лишь двумя упоминаниями, но на деле их может гораздо больше.

* 1. **Счётчик**

В python также есть такое понятие, как цикл. Циклы позволяют нам сокращать код. Иногда сокращения незначительные, а иногда они могут избавить от тысяч строк кода. Например, необходимо 1000 раз вывести какую-либо переменную или строку. Мы можем написать 1000 строк кода типа print(‘Hello’). Или же зациклить эти действия и от 1000 строк перейти всего лишь к 1-2 строкам кода.

Исходя из этого, сразу возникает вопрос, нам ведь должно что-то помогать отслеживать количество выполнений цикла. Да, для этого нам необходима переменная (счетчик). При помощи счетчиков мы можем контролировать выход из цикла. Для этого сравнивают переменную с каким-либо значением.

x = 1

while x < 10:

print(x)

x = x + 1

В данном случае у нас переменная x будет являться счетчиком и если мы запустим цикл, то увидим, что у нас вывелись цифры от 1 до 9 включительно. А значит наш цикл выполнился 9 раз.

**Заключение**

Переменная — это простейшая именованная структура данных, в которой может быть сохранён промежуточный или конечный результат работы программы. Бывают переменные четырёх типов: int, float, str, list.

На самом деле, способов оптимизации кода с их помощью может быть множество, но пока я знаком лишь с вышеописанными ситуациями, где они действительно могут помочь.

**Список использованных истоников**

1. Статья о переменных − [Электронный ресурс] –– <https://informatics.msk.ru/mod/book/view.php?id=42166&chapterid=839>
2. Статья о цикле while − [Электронный ресурс] –– https://dzen.ru/a/ZRJr3\_8oj1EXL4ZF?referrer\_clid=1400