Nama : Maskhan Musa

No. Peserta : 202438

Tugas : Exercise Python Basics dan Object Oriented Programming (OOP) #02

**TEORI**

1. Perbedaan *break, continue, dan pass* dinyatakan sebagai berikut:

* *Break* : Pernyataan break dalam Python mengubah aliran loop dengan menghentikannya setelah kondisi tertentu terpenuhi.
* *Continue* : Pernyataan continue dalam Python digunakan untuk melewati kode yang tersisa di dalam loop hanya untuk iterasi saat ini.
* *Pass* : Pernyataan pass dalam Python digunakan saat suatu pernyataan atau kondisi harus ada dalam program, tetapi kita tidak ingin perintah atau kode apa pun dijalankan. Biasanya digunakan sebagai pengganti kode di masa mendatang.

Menurut saya command *pass* dan none tidak dapat saling menggantikan, karena keduanya memiliki fungsi yang berbeda. Command *pass* adalah pernyataan kosong yang digunakan sebagai placeholder. Misalnya, dalam definisi fungsi atau kelas yang belum diimplementasikan bisa menggunakan *pass* untuk menghindari kesalahan sintaks sedangkan command *none* adalah objek khusus yang menunjukkan ketidakadaan nilai atau tidak ada nilai. Biasanya digunakan untuk menginisialisasi variabel atau sebagai nilai pengembalian dari fungsi yang tidak mengembalikan apa-apa. Jadi, meskipun keduanya berfungsi dalam konteks yang berbeda, mereka tidak dapat digunakan secara bergantian.

1. Perbedaan *list* dan *tuple* yaitu *list* dapat diubah, sedangkan *tuple* tidak dapat diubah. Tuple cocok digunakan untuk menyimpan data yang tidak boleh diubah, seperti rekaman basis data. Tuple juga membuat kode lebih aman dari modifikasi yang tidak disengaja.
2. fungsi dan perbedaan antara “*dictionary*” dan “*set*” adalah sebagai berikut:

* Dictionary
  1. Fungsi: Menyimpan data dalam pasangan kunci-nilai (key-value).
  2. Struktur: {kunci: nilai}.
  3. Akses: Menggunakan kunci.
* Set
  1. Fungsi: Menyimpan item unik tanpa duplikasi.
  2. Struktur: {item}.
  3. Akses: Menggunakan keberadaan item.

Perbedaan Utama antara *dictionary* dan *set* yaitu:

* Kunci vs. Item: Dictionary memiliki kunci; set hanya memiliki item.
* Dukungan Duplikasi: Dictionary mengizinkan nilai duplikat; set tidak.
* Urutan: Dictionary mempertahankan urutan; set tidak terurut.

1. Menurut saya fungsi dan perbedaannya fungsi dan perbedaan antara “*try…except…finally*” dan “*try…except…else*” yaitu:

* *try...except...finally* mempunyai fungsi untuk menangani pengecualian. Kode di dalam finally selalu dieksekusi, terlepas dari apakah terjadi pengecualian atau tidak. Berguna untuk pembersihan (misal: menutup file).
* *try…except…else* mempunyai fungsi untuk menangani pengecualian. Kode di dalam else hanya dieksekusi jika tidak ada pengecualian dalam blok *try*. Berguna untuk kode yang hanya perlu dijalankan jika semuanya berhasil.

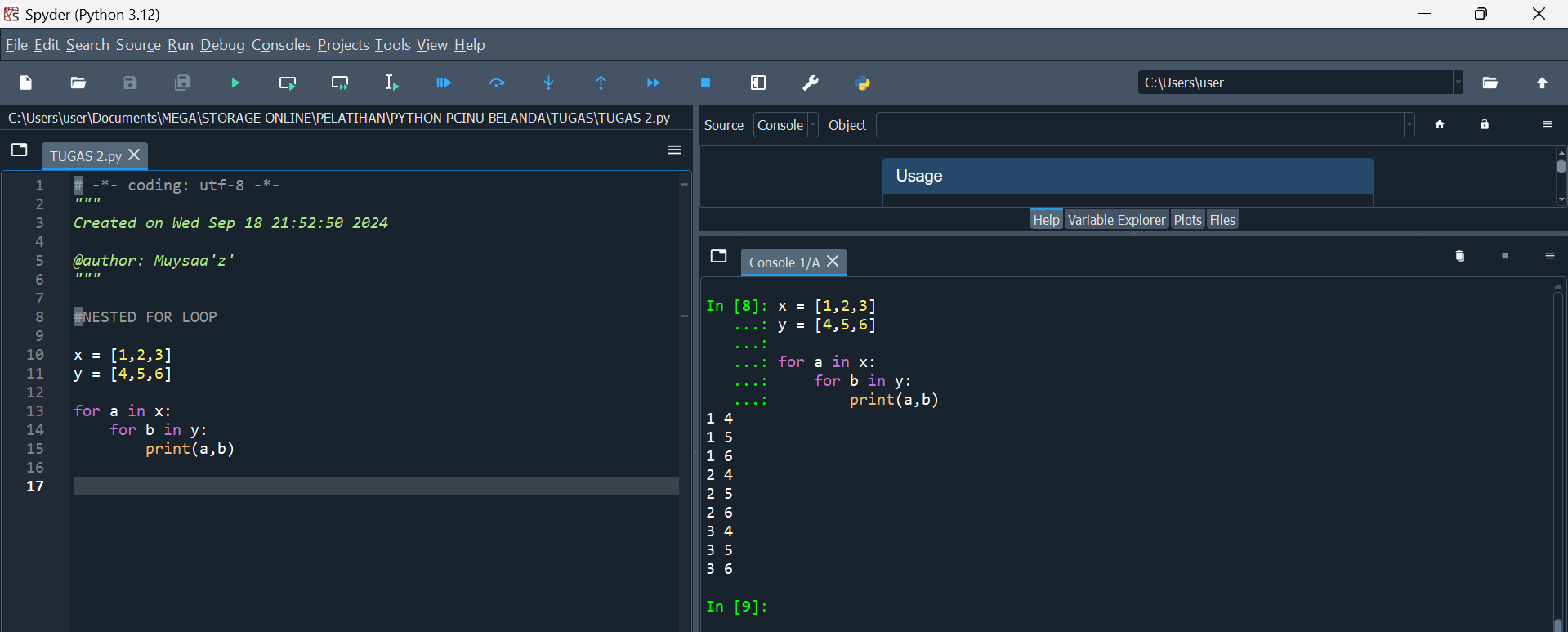
1. 5 contoh “built-in exceptions” di Python adalah:

* ValueError: Terjadi ketika argumen yang diberikan tidak sesuai dengan tipe yang diharapkan, meskipun tipe datanya benar.
* TypeError: Terjadi ketika operasi atau fungsi diterapkan pada objek dengan tipe yang tidak sesuai.
* IndexError: Terjadi ketika mencoba mengakses indeks yang tidak ada dalam sebuah list atau tuple.
* KeyError: Terjadi ketika mencoba mengakses kunci yang tidak ada dalam sebuah dictionary.
* ZeroDivisionError: Terjadi ketika mencoba membagi angka dengan nol.

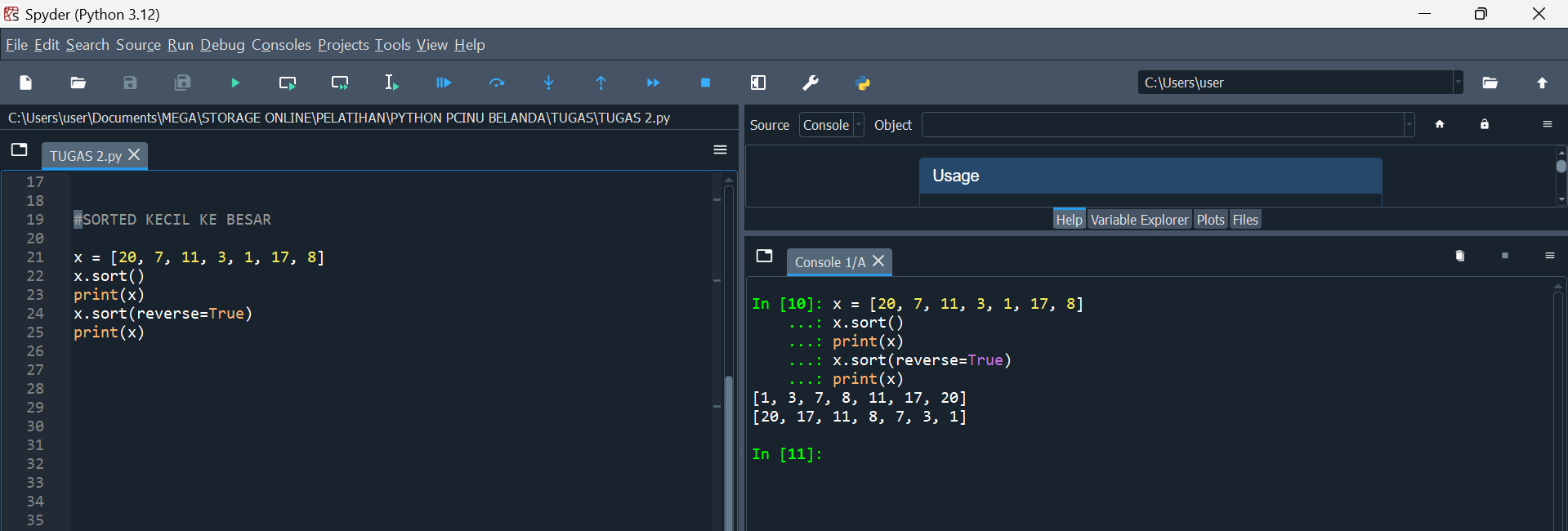
1. Dalam Python, *function* adalah blok kode yang dapat dipanggil untuk menjalankan tugas tertentu. Fungsi membantu dalam mengorganisir dan mengelola kode, membuatnya lebih modular dan mudah dibaca. komponen-komponen utamanya:
2. Definisi Fungsi: Dimulai dengan kata kunci *def*, diikuti dengan nama fungsi dan tanda kurung.
3. Nama Fungsi: Menunjukkan apa yang dilakukan fungsi tersebut. Nama harus mengikuti aturan penamaan variabel.
4. Parameter (opsional): Variabel yang diterima fungsi sebagai input. Didefinisikan di dalam tanda kurung.
5. Dokumentasi String (opsional): Menyediakan penjelasan tentang fungsi. Diletakkan di bawah definisi fungsi menggunakan triple quotes.
6. Body Fungsi: Kode yang dijalankan ketika fungsi dipanggil. Harus diindentasikan dengan baik.
7. Return Statement (opsional): Mengembalikan nilai dari fungsi ke pemanggil. Jika tidak ada, fungsi akan mengembalikan None secara default.

**PRAKTEK**

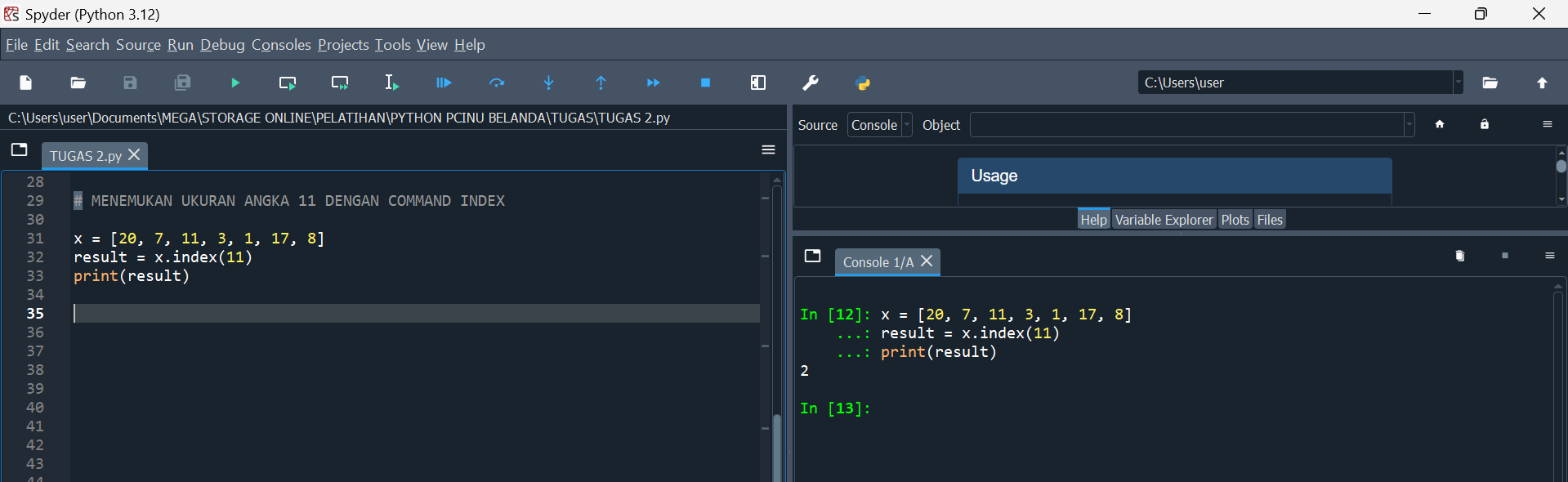
1. Contoh hasil dari nested for loop adalah sebagai berikut



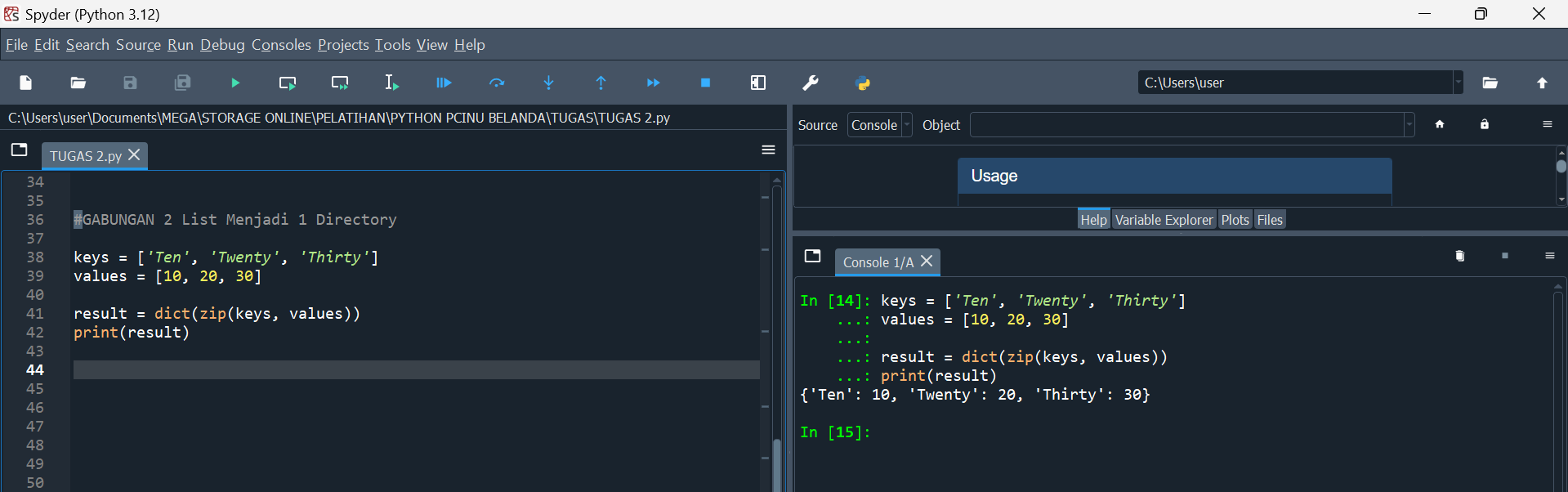
1. Program menggunakan command “sorted” dari kecil ke besar dan sebaliknya



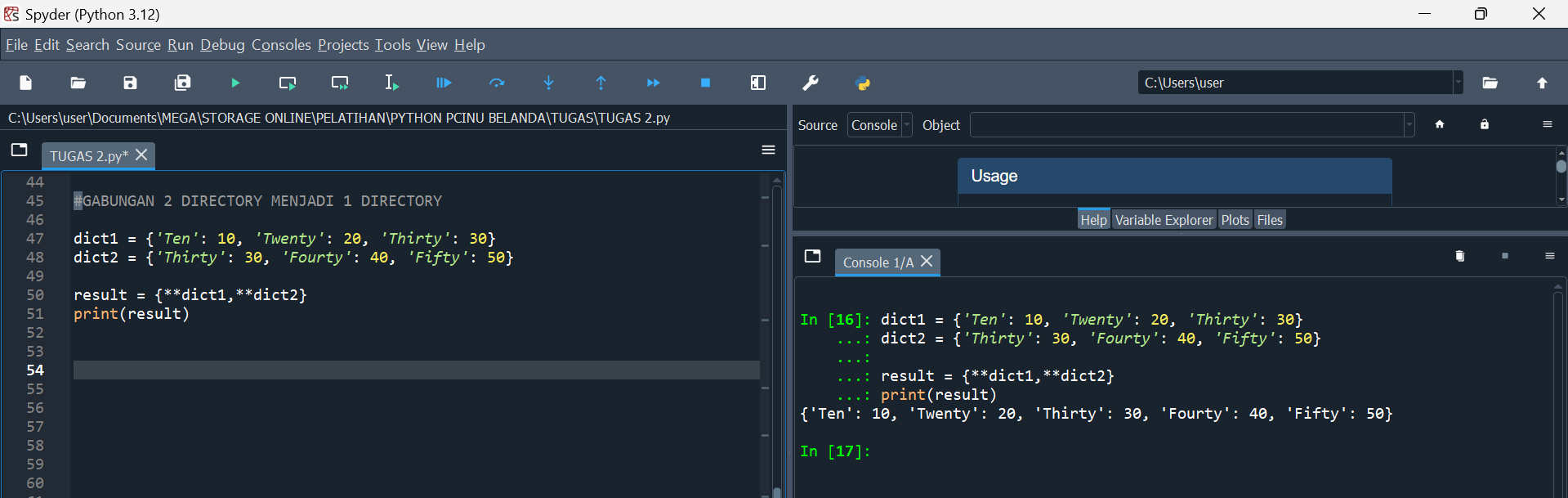
1. Menemukan urutan angka 11 pada list sebagai berikut



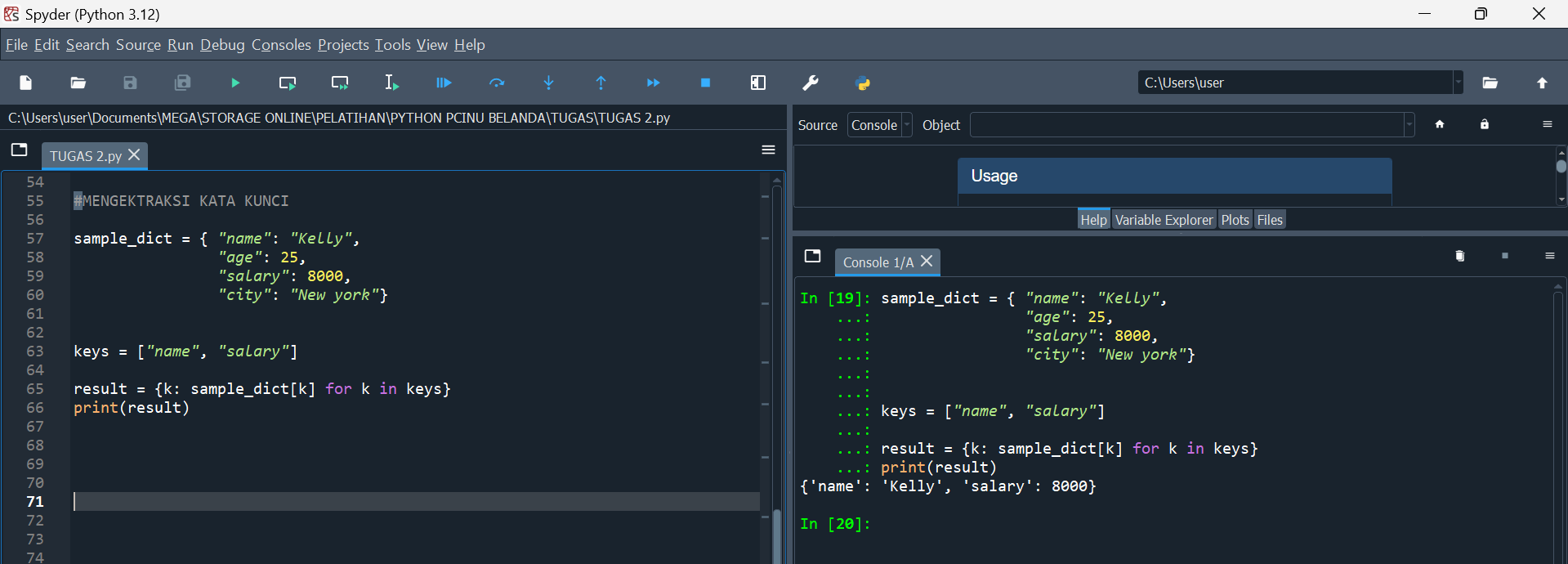
1. Gabungan 2 List menjadi 1 Dictionary



1. Gabungan 2 Dictionary menjadi 1 Dictionary



1. Mengektraksi kata kunci pada Dictionary



1. Function Calculate\_area()

