|  |  |
| --- | --- |
| A picture containing text, clipart  Description automatically generated | **Laporan Praktikum Algoritma dan Pemrograman**  Semester Genap 2023/2024 |

|  |  |
| --- | --- |
| **NIM** | **71230973** |
| **Nama Lengkap** | **<ISI DENGAN NAMA LENGKAP ANDA>** |
| **Minggu ke / Materi** | **04 / Modular Programming** |

**SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.**

**SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.**

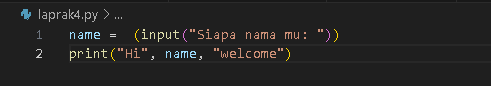
**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
YOGYAKARTA  
2024**

# BAGIAN 1: MATERI MINGGU INI (40%)

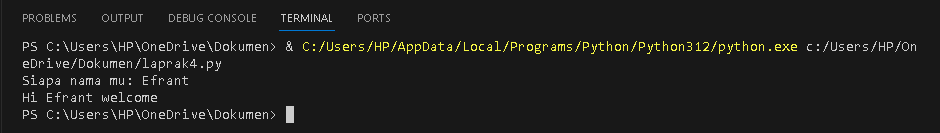
Pada minggu ini, materi yang disampaikan adalah modular programming. Pada materi ini, kita belajar mengenai fungsi, argument, parameter, return value, named argument, optional argument, dan lambda. Pada materi-materi tersebut harapannya kita bisa memahami anatomi dan struktur dari suatu fungsi, penggunaan parameter dan hasil dari suatu fungsi dan dapat mendefinisikan fungsi dan parameter yang dibutuhkan dalam memecahkan suatu masalah.

## Fungsi, Argument, dan Parameter

Source Code :



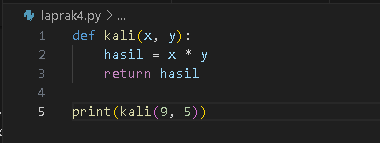
Output :



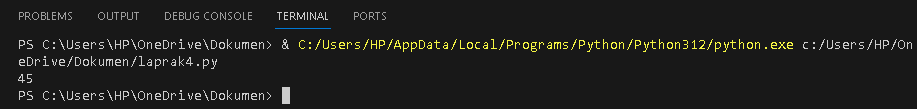
Program yang berada pada gambar diatas menampilkan seorang user yang perlu menginputkan namanya. Lalu program tersebut akan menyapa nama yang diinputkan. Tentu ini hal yang biasa lakukan pada program biasa. Disana teradpat 2 fungsi program bawaan python, yaitu input() dan print(). Input() berguna untuk menerima input yang dimasukkan oleh pengguna, sedangkan print() untuk menampilkan sebuah teks pada output yang kita inginkan. Lalu apa kaitan nya dengan Modular Programming?

Fungsi merupakan kumpulan perintah yang digabungkan menjadi satu untuk menciptakan suatu program yang memiliki tujuan tertentu dan memiliki kegunaan khusus yang bisa digunakan berulang-ulang. Jika kita ingin membuat suatu program yang kompleks, tentunya kita harus mengumpulkan beberapa kode program menjadi satu block, dan tentunya membutuhkan banyak langkah. Karena itu disebut modular. Berikut merupakan contoh.

Source Code :

\

Output :



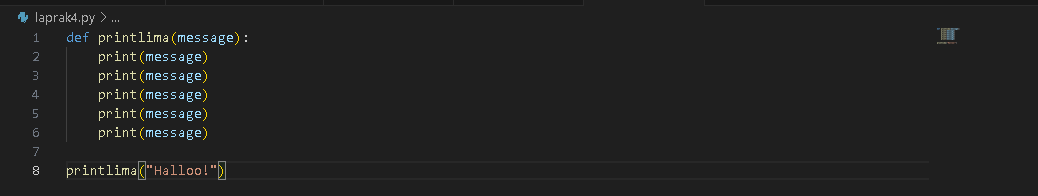
Dari fungsi diatas code “def” digunakan untuk mendefinisikan sebuah fungsi. Lalu nama fungsi yang dibuat yaitu “kali”. Perlu diingat, setelah kita menuliskan def dan nama fungsi, isi dari fungsi harus berada dalam def agar masuk dengan jarak 1 tab. Fungsi kalli ini harus memuliki setidaknya 2 argument yang nantinya biasa disebut dengan parameter x dan y. Fungsi diatas nantinya akan menghasilkan hasil perkalian yang ditampung dari sebuah variable dengan cara menambahkan “return” pada source code. Return ini digunakan untuk mengembalikan nilai dari suatu fungsi.

Fungsi diatas bsia dipanggil dengan cara “print(nama fungsi(parameter1, parameter2))”. Dengan cara ini, kita akan mendapatkan hasil output dari fungsi tersebut. Penulisan dari parameter harus dituliskan secara berurut. Parameter akan mengikuti parameter mana yang ditulis pertama. Seperti itulah ilustrasi pertama pada modular programming ini.

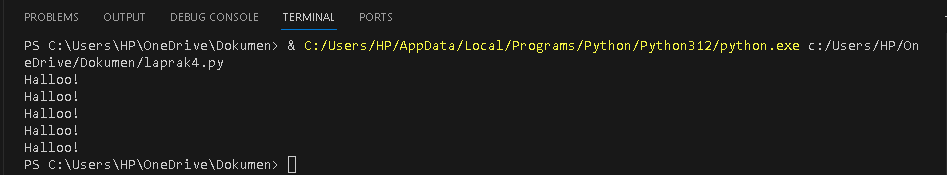
## Return Value

Berdasarkan hasil yang dikeluarkan fungsi, secara umum ada 2 jenis yaitu fungsi yang mengembalikan nilai dan fungsi yang tidak mengembalikan nilai. Yang akan kita bahas disini yaitu fungsi yang tidak mengembalikan nilai yang biasa disebut dengan void function. Sebagai contoh pada Source Code dibawah tersebut.

Source Code :

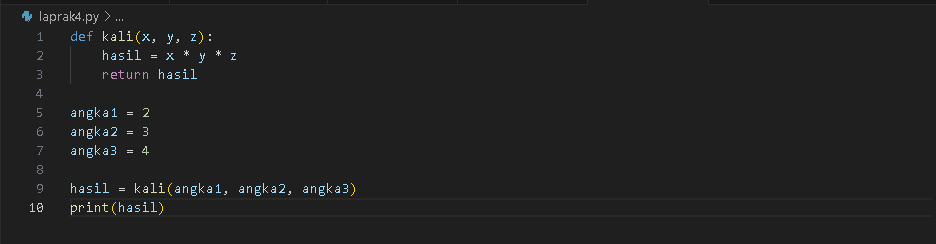


Output :

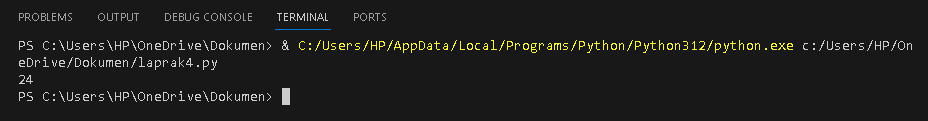


Pada fungsi printlima(), hanya dibutuhkan 1 parameter yaitu massage. Didalam fungsi “printlima()” tersebut akan menampilkan variable massage sebanyak 5 kali. Cara memanggilnya kita cukup menulis nama fungsi lalu memasukan parameter yang kita inginkan seperti pada source code diatas. Lalu akan menghasilkan output parameter yang kita masukkan sebanyak 5 kali. Contoh lainnya seperti berikut.

Source Code :



Output :

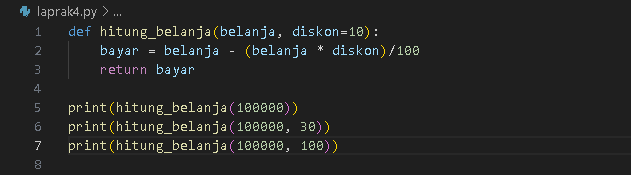


Pada fungsi kali(), dibutuhkan 3 parameter yaitu x, y, z. Kemudian fungsi kali tersebut didefinisikan suatu variable hasil yang ditulis x\*y\*z. Kemudian variable hasil akan dikeluarkan dengan cara menuliskan keyword “return”. Return ini digunakan untuk mengeluarkan hasil dari fungsi dan juga mengakhiri suatu fungsi.

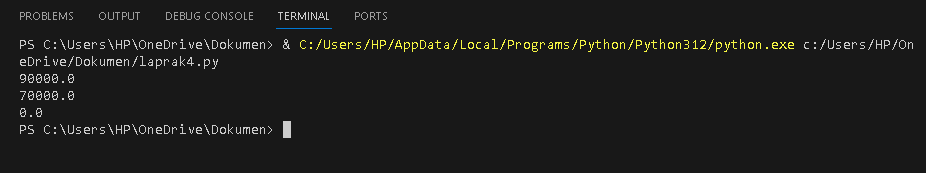
## Optional Argument dan Named Argument

Fungsi dapat memiliki opsional parameter. Ada 2 kemungkinan yaitu parameter yang bersifat opsional dan ada juga parameter yang memiliki nilai default(bawaan). Untuk mendiefinisikan optional parameter, kita harus mendefinisikan nilai bawaannya terlebih dahulu seperti pada contoh berikut ini.

Source Code :



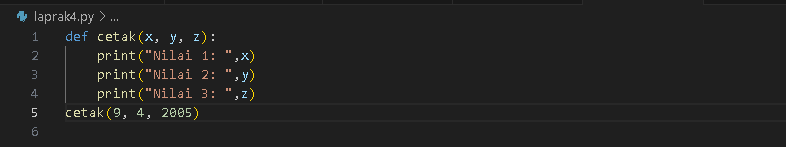
Output :



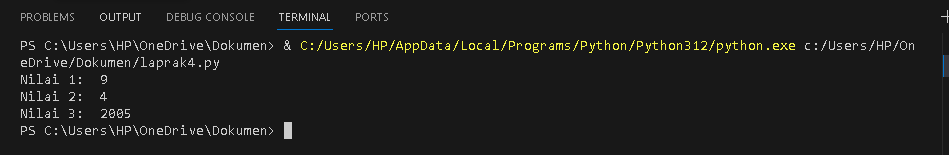
Fungsi hitung\_belanja() ini memiliki dua parameter yaitu belanja dan diskon. Pada penyontohan parameter optional, ditujukan pada parameter “belanja”. Pada parameter “belanja” ini akan ditentukan pada pemanggilan fungsi, sedangkan untuk penyontohan parameter default, ditujukan pada parameter “diskon”. Pada source code bisa dilihat “diskon = 10” untuk menyatakan bahwa diskon awal yang pasti didapatkan 10%. Jadi pada pemanggilan kita tidak perlu menuliskan parameter diskon seperti contoh contoh print(hitung\_belanja(100000)). Tetapi parameter default tersebut bisa digantikan jika kita menginginkannya dengan cara menuliskan Kembali parameter diskon pada pemanggilan seperti contoh print(hitung\_belanja(100000, 30)).

Pada contoh yang diatas, itu merupakan pemanggilan dengan 1 argument, sedangkan pemanggilan kedua dan ketiga menggunakan dua argument. Setiap parameter pada fungsi pastinya memiliki nama yang masing-masing. Karena itu pada pemanggilan fungsi juga dapat disertakan nama paarameterny dan tidak perlu sesuai dengan urutan yang diberikan seperti contoh berikut.

Source Code :



Output :

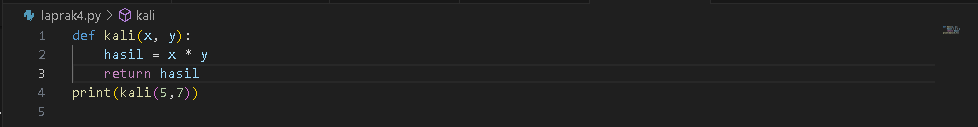


## Anonymous Function(Lambda)

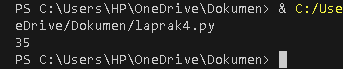
Sesuai dengan Namanya Anonymous Function merupakan fungsi yang tidak ada nama atau tanpa nama. Anonymous Function atau lambda ini merupakan fungsi tambahan pada Python bukan fitur utama. Berbeda dengan bahasa pemrograman lainnya yang merupakan bahasa fungsional. Keyword lambda mendefinisikan Anonymous Function pada Python. Berikut merupakan contoh perbedaan fungsi biasa dengan lambda.

Biasa :

Source Code :

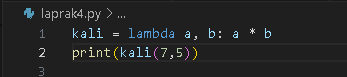


Output :

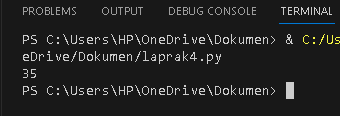


Lambda

Source Code :



Output :



Bisa dilihat perbedaan fungsi biasa dengan fungsi lambda. Fungsi lambda jauh lebih efesien dibandingkan dengan fungsi biasa. Setiap anonymous function pada Python terdiri dari beberapa bagian berikut ini:

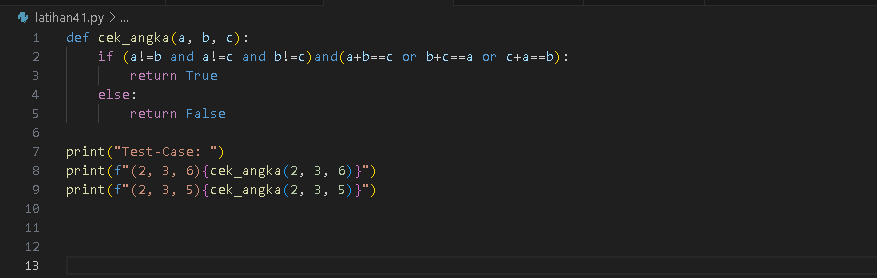
1. Keyword: lambda
2. Bound variable: argument pada lambda function
3. Body: bagian utama lambda, berisi ekspresi atau statement yang menghasilkan suatu nilai.

# BAGIAN 2: LATIHAN MANDIRI (60%)

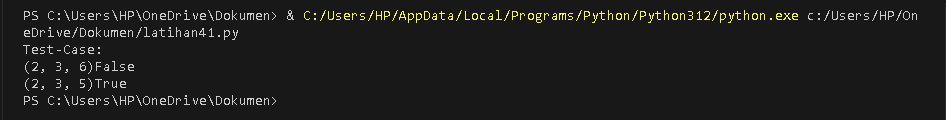
Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

## SOAL 1

Source Code :



Output :

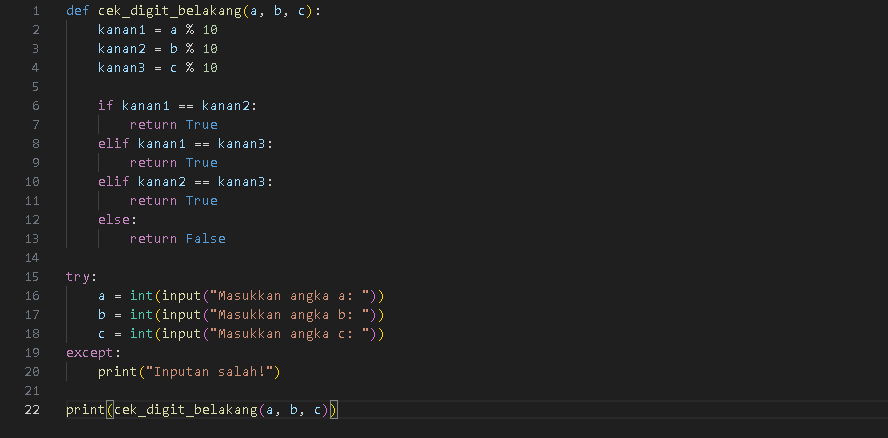


Penjelasan :

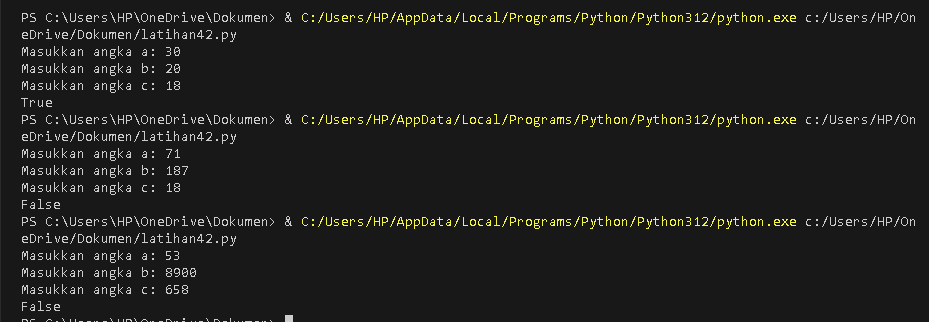
Nama fungsi yang digunakan yaitu cek\_angka dan memiliki tiga parameter yaitu a, b, dan c. jadi pada condition pertama yang ada pada if, dinyatakan setiap angka pada pemanggilan parameter tidak boleh sama harus berbeda semua, jika ada yang sama maka hasilnya akan false. Lalu dua angka pada parameter jika dijumlahkan hasilnya adalah angka yang tersisa. Jika tidak sesuai hasil penjumlahannya maka akan menghasilkan False. Lalu untuk pemanggilannya dengan cara print((nama fungsi(ketiga parameter)).

## SOAL 2

Source Code :



Output :

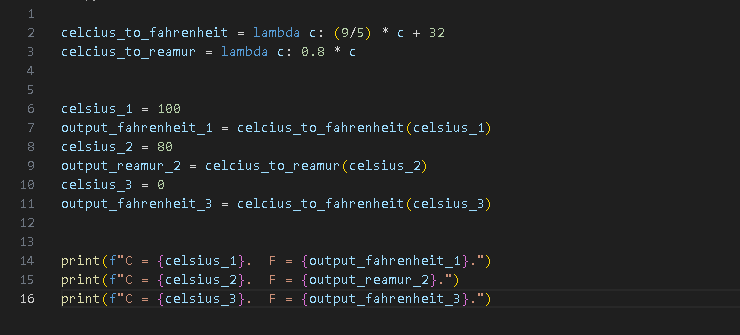


Penjelasan :

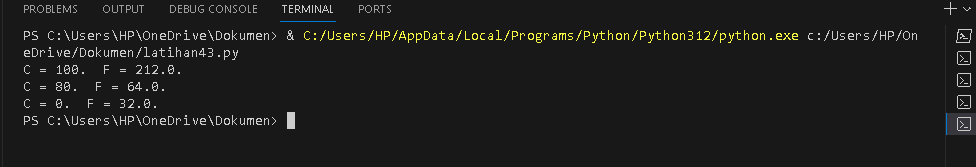
Nama fungsi yang digunakan yaitu cek\_digit\_belakang, dan memiliki 3 parameter yaitu a, b, c. Untuk menghasilkan angka paling kanan atau belakang yaitu dengan cara menggunakan modulus 10(% 10). Maka dibuatlah 3 variabel yaitu kanan1-3 untuk menghasilakn angka paling kanan dari parameter yang akan diinputkan oleh user. Lalu digit paling kanan tersebut harus ada yang sama dengan digit kanan parameter lainnya setidaknya 1 yang sama. Jika tidak ada digit paling kanan yang sama maka akan menghasilkan output False.

## SOAL 3

Source Code :



Output :



Penjelasan :

Jadi pada lambda function, dituliskan nama fungsi nya masing- masing terlebih dahulu yaitu celcius\_to\_fahrenheit dan celcius\_to\_reamur. Lalu menuliskan rumus fungsi yang kita inginkan. Setelah itu kita membuat variable untuk pemanggilan fungsi lambda tersebut yaitu celcius\_1-3. Lalu kita lakukan pemanggilan dengan cara print( variable yang kita buat) dan menghasilkan output seperti pada soal yang diinginkan.