# BAB V

# FUNCTION DAN METHOD

## 5.1 Tujuan (Minimal 5)

## 5.2 Dasar Teori (Tidak boleh copas modul, Disertai sumber)

### 5.2.1 Function

### 5.2.2 Method

### 5.2.3 Penggunaan Function

### 5.2.4 Penggunaan Method

## 5.3. Hasil Percobaan dan Analisis

### 5.3.1 Function pada Java

### 5.3.2 Method pada Java

### 5.3.3 Function pada C++

### 5.3.4 Method pada C++

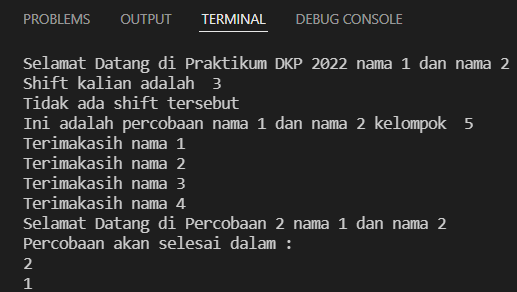
### 5.3.5 Function pada Python

### 5.3.6 Method pada Python

Berikut adalah contoh *source* *code* *method* pada Python:

|  |
| --- |
| class contoh\_method:      #init method      def \_\_init\_\_(self, praktikan1, praktikan2):          self.praktikan1 = praktikan1          self.praktikan2 = praktikan2      #self parameter      def mulai(self):          print(f"Selamat Datang di Percobaan 2 {self.praktikan1} dan {self.praktikan2}")      #method dengan parameter      def selesai(self, waktu):          print("Percobaan akan selesai dalam :")          while waktu > 0:              print(waktu)              waktu -= 1  from percobaan2 import \*  #Function dengan non return type  def non\_return\_func(praktikan1, praktikan2):      print(f"Selamat Datang di Praktikum DKP 2022 {praktikan1} dan {praktikan2}")  #Function dengan return type  def return\_func(shift):      print(f"Shift kalian adalah ", shift)      if (shift == 1) or (shift == 2) :          return print(f"Fungsi return mengembalikan nilai menjadi ", shift \* 2)      else:          return print("Tidak ada shift tersebut")  #Function dengan Arbitrary Type  def arbitrary\_func(\*penutup):      for nama in penutup:          print("Terimakasih", nama)  #Anonymous Function  anonim\_func = lambda praktikan1, praktikan2, kelompok: print(f"Ini adalah percobaan {praktikan1} dan {praktikan2} kelompok ", kelompok )  #Pemanggilan Fungsi  non\_return\_func("nama 1", "nama 2")  return\_func(3)  anonim\_func("nama 1", "nama 2", 5)  arbitrary\_func("nama 1", "nama 2", "nama 3", "nama 4")  praktikan = contoh\_method("nama 1", "nama 2")  praktikan.mulai()  praktikan.selesai(2) |

Berikut adalah hasil *output* dari *source* *code* di atas:

****

Gambar 5. X Output percobaan method pada Python

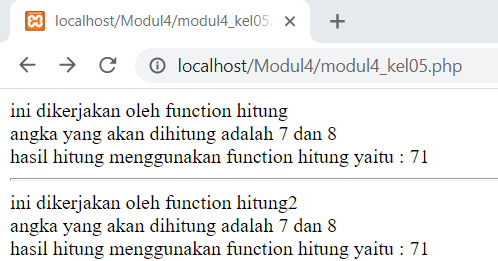
Program di atas menggunakan satu kelas baru yang digunakan untuk menampung *method*. Pada *source* *code* di atas, *method* menggunakan init () yang dijalankan secara otomatis ketika kelas metode dipanggil pada objek. Kemudian *self*-*parameter* sebagai parameter yang berhubungan dengan *class* itu sendiri dan harus dideklarasikan sebagai parameter pertama dari *method*. Parameter ini perlu diletakkan pertama dan *user* dapat menamainya dengan bebas.

### 5.3.7 Function pada PHP

Berikut adalah contoh *source* *code* *function* pada PHP:

|  |
| --- |
| <?php  //ini return type  function hitung ($bil1, $bil2) {  return $bil1 + $bil2 \* $bil2;  }  $bil1 = 7;  $bil2 = 8;  $hasil = hitung($bil1, $bil2);  echo "ini dikerjakan oleh function hitung <br>";  echo "angka yang akan dihitung adalah $bil1 dan $bil2 <br>";  echo "hasil hitung menggunakan function hitung yaitu : $hasil";    echo "<hr>";  //ini non return type  function hitung2 ($bil1, $bil2) {  $hasil = hitung($bil1, $bil2);  echo "angka yang akan dihitung adalah $bil1 dan $bil2 <br>";  echo "hasil hitung menggunakan function hitung yaitu : $hasil";  }  echo "ini dikerjakan oleh function hitung2 <br>";  hitung2($bil1, $bil2); |

Berikut adalah hasil *output* dari *source* *code* di atas:

****

Gambar 5. X Output percobaan function pada PHP

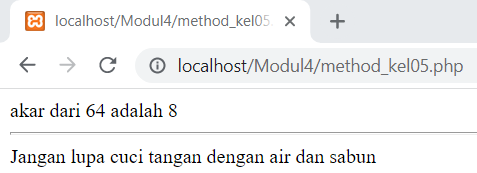
*Source* *code* di atas terdapat dua jenis *function* pada PHP, yaitu *return* *type* dan *non return type*. Pada *return* *type*, setelah membuat *function* langsung membuat *return* dengan *code* return $bil1 + $bil2 \* $bil2; supaya fungsi dapat mengembalikan nilai. Sedangkan pada *non return type* tidak diperlukan adanya *return*, namun langsung dengan mendeklarasikan suatu formula atau argumen seperti $hasil = hitung($bil1, $bil2);.

### 5.3.8 Method pada PHP

Berikut adalah contoh source code method pada PHP:

|  |
| --- |
| <?php  declare(strict\_types = 1);  //class pendefinisi method  class pendefinisi {  public function akar(float $angka) : float {  return sqrt($angka);  }  public function staysafe (string $noun1, string $noun2) {  echo "Jangan lupa cuci tangan dengan $noun1 dan $noun2";  }  }  //code pemanggil method  $angka = 64;  //$objek merupakan instance dari class pendefinisi  $objek = new pendefinisi();    // OOP pada PHP menggunakan -> bukan .  echo "akar dari $angka adalah ".$objek -> akar($angka);  echo "<hr>";  $objek -> staysafe('air', 'sabun'); |

Berikut adalah hasil output dari source code di atas:

****

Gambar 5. X Output percobaan method pada PHP

Pada program di atas untuk menyatakan method menggunakan panah, seperti pada $objek -> staysafe('air', 'sabun'). Ini berbeda pada bahasa pemrograman pada umumnya yang menggunakan titik.

## 5.4 Tugas (Source code diketik, diberi SS hasil, dan di analisa)

## 5.5 Kesimpulan (Minimal 5, bukan berasal dari dasar teori tetapi berdasarkan dari yang didapatkan dalam praktikum dan menjawab tujuan)

1. *Method* dan *function* sama-sama merupakan *function*, namun perbedaan terdapat pada letaknya apakah di dalam *class* atau luar class.

2. Penggunaan *method* dan *function* dalam setiap bahasa pemrograman berbeda-beda.

3. Indentasi dalam penggunaan *method* dan *function* penting.

4. *Method* dapat berparameter dan juga tidak berparameter.

5. *Function* ada dua jenis yaitu *return* *type* dan *non return type*.

**Tugas:**

Buatlah sebuah program yang menggunakan materi dari modul 1 hingga 4. (Wajib ada pengkondisian, perulangan, function dan method) menggunakan salah satu Bahasa pemrograman yang telah diajarkan. Program WAJIB MELIPUTI:

* Function
* Method
* Return type berparameter dan tanpa parameter (Funtion/Method)
* Non-return type (Funtion/Method)
* Apabila function sudah menggunakan return type maka untuk methodnya wajib ada non return type, begitu juga sebaliknya

**TUGAS TIDAK BOLEH MEMBUAT PROGRAM KALKULATOR**.

**TUGAS TIDAK BOLEH SAMA** antar Kelompok (bila sama nilai akan dibagi per jumlah kelompok yang sama). Untuk itu praktikan dapat mengisi judul program terlebih dahulu pada sheet berikut

https://docs.google.com/spreadsheets/d/1dIFyU6-JDTHUDws3IYJSY\_db8JpMhRoIn\_mdY\_MR7wE/edit?usp=sharing

Project tugas di upload ke GitHub dengan tiap anggota harus menjadi kolaborator, dan minimal memberikan 1 commit pada project.

Sertakan link GitHub pada laporan, di bagian bawah Analisis Tugas

**Ketentuan Penulisan Format Praktikum**

1. Format Penulisan :
2. Margin : 4433 (kiri, atas, kanan, bawah)
3. Ukuran kertas : A4
4. Menggunakan *font* *Times New Roman* ukuran 12pt warna hitam
5. Jarak antar baris tiap paragraf 1.5 dengan spasi *after* dan *before* : 0pt
6. Paragraf menggunakan *justify*
7. Semua kata yang tidak ada KBBI atau istilah asing diketik miring (*italic*)
8. Source code menggunakan *font* *Courier new* ukuran 10pt warna hitam , dan diberi kotak
9. Jarak antar baris di *source code* 1.0
10. Keterangan gambar menggunakan fitur *Insert Caption* yang ada di tab *References ,* menggunakan font times new roman ukuran 10 warna hitam.
11. Jarak antar gambar ke keterangan 1.0, rata tengah (center), tidak perlu di-**bold**
12. Penulisan sumber menggunakan font times new roman ukuran 10, rata kanan.
13. Tiap sub-bab ganti halaman (menggunakan *page break)* dan tiap sub-sub-bab diberi jarak satu *Enter*.
14. Judul sub-bab diterapkan heading runtut dan sejajarkan.
15. **Apabila ketahuan *Copy Paste* atau *Download Edit***, maka akan disuruh membuat laporan baru dan jika tidak membuat laporan baru, maka **nilai laporan= 0.**
16. Asistensi Responsi dapat dimulai tanggal 20 Maret 2022 – 26 Maret 2022

**MAKSIMAL AC Minggu, 27 Maret 2022 Pukul 23.59 WIB**

**MAKSIMAL ACC Jumat, 30 Maret 2022 Pukul 23.59 WIB**

Lewat Deadline nilai akan dikurangi 30 poin per-harinya.

1. Kelompok dari shift 1 asistensi laporan dan responsi ke Farrel Ahmad Yudithia.
2. Kelompok dari shift 2 asistensi laporan dan responsi ke Abdullah Faqih.
3. Laporan dikirim ke e-mail dengan *subject* dan nama *file*:

**LAPORAN\_PDKP\_MODUL4\_KELXX.docx**  
(SUBJECT SALAH BERPOTENSI LAPORAN TIDAK TERPERIKSA)

1. Konfirmasi ke asprak yang bersangkutan apabila telah mengirim laporan

Kontak Asisten:

* Farrel Ahmad Yudithia [farrelfay.ce@gmail.com]

Line: farrelay01

* Abdullah Faqih [asprakdkp.faqih@gmail.com]

Line : alexaqih