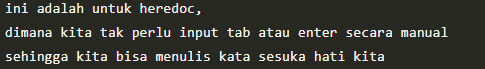
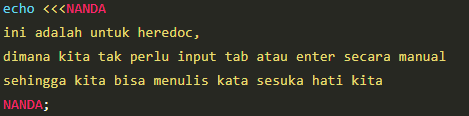
* **HEREDOC & NOWDOC**

**HEREDOC**

* Ini untuk membuat string yang panjang, sehingga tidak perlu manual untuk enter, tab dan yang lain



**NOWDOC**

* Mirip seperti heredoc, bedanya nowdoc **tidak punya kemampuan parsing atau double quote**



* **CONSTANT**

**DEFINISI**

* Bisa dibuat dengan **define(“nama\_const”, “isi\_const”);**





* **NULL**

**DEFINISI**

* Ini untuk mengisi variable dengan nilai yang tidak ada apa-apa
* Untuk mengecek, apakah variable bernilai null, dengan cara **is\_null($nama\_var)**;, outoputnya adalah **boolean**

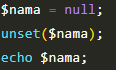




* **UNSET**

**DEFINISI**

* Ini untuk menghapus sebuah variable, dan tidak bisa mengakses variabelnya lagi
* Dengan cara **unset($nama\_var)**

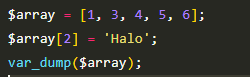


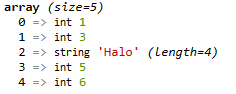


* **OPERASI ARRAY**

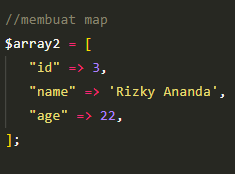
**DEFINISI**

* **$array[index]**, untuk **mengakses** data di array dengan nomor index
* **$array[index] = value**, untuk **mengubah** data di array dengan nomor index dengan value baru
* **$array[] = value**, **menambah** data di array pada posisi yang **paling belakang**
* **unset($array[index])**, **menghapus** data di array dan **index hilang** juga
* **count($array)**, untuk **mengambil total** data di array





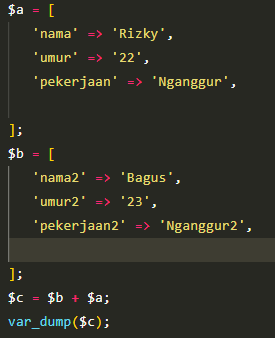
//ini adalah MAP

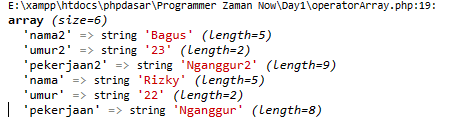


* **OPERATOR ARRAY**

**DEFINISI**

* **$a + $b** , **menggabungkan** array $a dan $b
* **$a == $b** , **true** jika key dan valuenya sama
* **$a === $b** , **true** jika key dan valuenya sama dan posisi sama
* **$a != $b** , true jika $a dan $b tidak sama
* **$a !== $b** , true jika $a dan $b tidak tidak identik





* **KONVERSI**

**DEFINISI**

* **$valueString = (string)100;,** ini akan mengubah integer menjadi string. Kita bisa mengubah menjadi float dan int dengan mengubah yang ada di dalam ( ).

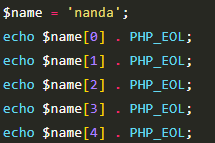




* **SIFAT STRING MENJADI ARRAY**

**DEFINISI**

* Pada php, string itu bersifat seperti array contohnya seperti berikut:

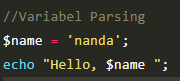




* **VARIABEL PARSING**

**DEFINISI**

* Ini adalah, kita bisa menggabungkan string dengan variable tanpa harus menggunakan titik. Contohnya





* **TERNARY OPERATOR**

**DEFINISI**

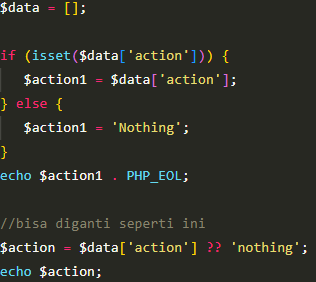
* Ini untuk **mempersingkat if and else** operator



* **COALESCING OPERATOR**

**DEFINISI**

* Ini untuk **mempersingkat** penggunaan **isset**

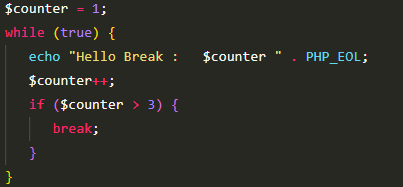


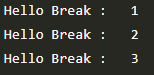


* **BREAK & CONTINUE**

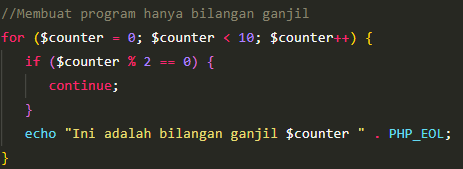
**DEFINISI**

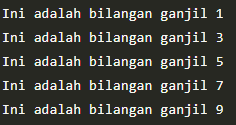
* **Break**, ini untuk **menghentikan** perulangan
* **Continue,** ini untuk **mengskip** perulangan saat ini





**//CONTINUE**

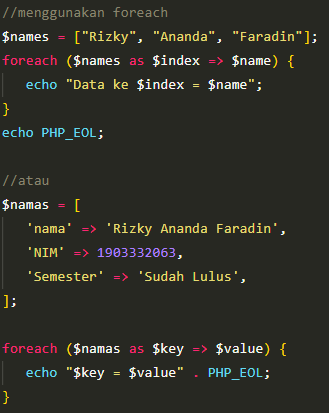


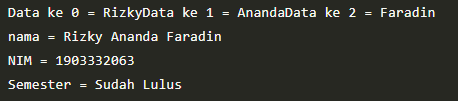


* **FOREACH ARRAY**

**DEFINISI**

* Ini untuk mengambil nilai array dengan foreach





* **DEFAULT PARAMETER**

**DEFINISI**

* Ini untuk memberikan default parameter ketika user tidak memasukkan parameter

//ini tidak boleh, usahakan berikan default parameter di belakang



* **VALID TYPES (TIPE YANG ADA DI DALAM PARAMETER)**

**DEFINISI**

* Ini untuk memberikan tipe data di parameter function, secara default php bisa mengonversi otomatis tipe data, contohnya ketika kita memasukkan argumen int, tapi tipe data dari parameter di function adalah string maka akan otomatis di konversi.
* Beberapa tipe parameter, yaitu

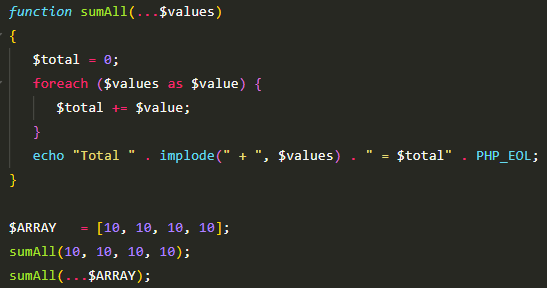
1. **Class / interface** > Parameter harus tipe **class/interface**
2. **Self** > Parameter harus sama dengan **class**, dimana function ini dibuat
3. **Array** > Parameter harus **array**
4. **Callable** > Parameter harus **callable**
5. **Bool** > Parameter harus **Boolean**
6. **Float** > Parameter harus **float**
7. **Int >** Paremeter harus **int**
8. **String >** Parameter harus **String**
9. **Interable >** Parameter harus **array** atau **Tipe Traversable**
10. **Object >** Parameter harus **object**



* **VARIABEL – LENGTH ARGUMENT LIST**

**DEFINISI**

* Ini untuk mengizinkan parameter bisa menerima banyak value.
* Ini akan otomatis membuat argument menjadi array

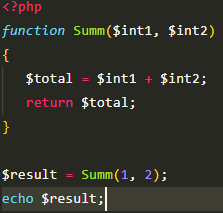




* **FUNCTION RETURN VALUE**

**DEFINISI**

* Secara default function tidak mengembalikan apapun, untuk itu gunakan return
* Kita juga bisa menambahkan **return default**, sehingga kita bisa tau tipe data kembalian dari sebuah function.



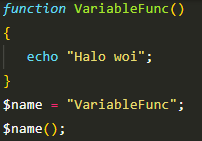
//default return



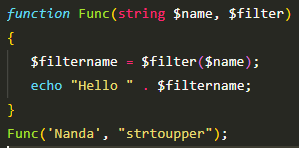
* **VARIABLE FUNCTION**

**DEFINISI**

* Kita bisa membuat function menjadi variable, dengan cara di bawah ini :



Fungsinya untuk memberikan argument berupa function





* **Anonymous Function (Closure)**

**DEFINISI**

* Kita bisa membuat function menjadi variable, dengan cara di bawah ini :

