

## BAB 5 LARIK atau ARRAY

taryanarx@email.unikom.ac.id

### 5.1 Pengertian Array

**Array** bukan merupakan tipe data dasar seperti **integer** atau **boolean**, **Array** merupakan sebuah tipe data bentukan yang terdiri dari kumpulan tipe data lainnya. Dengan Menggunakan array akan memudahkan dalam membuat kelompok data, serta menghemat penulisan dan penggunaan **variabel**.

PHP Memiliki 3 Tipe Array:

- **Indexed arrays** – Array dengan index numerik
- **Associative arrays** – Array dengan Nama Kunci
- **Multidimensional arrays** – Array yang memiliki satu atau lebih array

### 5.2 Index Array

PHP mendukung beberapa **cara penulisan array**, salah satunya dengan menggunakan **konstruktor array PHP** (*array language construct*) sebagai berikut:

```
$nama_variabel = array(  
Key1 => value1,  
key2 => value2,  
key3 => value3,  
...  
)
```

Komponen **array** terdiri dari pasangan kunci (**key**) dan nilai (**value**). **Key** adalah penunjuk posisi dimana **value** disimpan. Perhatikan juga bahwa PHP menggunakan *tanda panah* (=>) untuk memberikan nilai kepada key.

```
$nama = array (  
0=>"Budi",  
1=>"Sari",  
2=>"Gaga",  
3=>"Reni",  
4=>"Fahra");
```

Atau seperti ini

```
$nama []="Budi";  
$nama []="Sari";  
$nama []="Gaga";  
$nama []="Reni";  
$nama []="Fahra";
```

### 5.2.1 Membaca Array

Dalam mengakses nilai dari **array**, bisa menggunakan kombinasi **\$nama\_variabel** dan nilai **key**-nya, dengan penulisan sebagai berikut:

**\$nama\_variabel[key];**

Berikut adalah contoh pengaksesan array dalam PHP:

```
echo $nama[1]; //Sari  
echo $nama[2]; //Gaga  
echo $nama[3]; //Reni
```

Selain mendefinisikan key secara langsung, PHP juga memperbolehkan penulisan array tanpa key, dan key itu secara otomatis akan diurutkan dari nilai 0, 1, 2, dst.

Berikut adalah contoh pendefinisian array tanpa key:

```
$nama = array("Andri", "Joko", "Fahra", "Rina", "Gaga");
```

```
$nama = array("Andri", "Joko", "Fahra", "Rina", "Gaga");  
// pengaksesan array  
echo $nama[1]; //Joko  
echo $nama[2]; //Fahra  
echo $nama[3]; //Rina
```

Selain menggunakan angka, **key** dalam PHP dapat berisi **string** atau **boolean**. Sedangkan untuk value dapat menyimpan berbagai tipe data seperti *integer*, *float*, *string*, *boolean*, bahkan *array* lainnya

Berikut adalah contoh array dengan menggunakan berbagai tipe data untuk key dan value, yaitu dengan tipe data integer, string, dan boolean. Namun jika key di definisikan dengan tipe data boolean maka secara otomatis PHP akan mengkonversinya menjadi 1.

NamaFile:array51.php

```
<?php
$coba = array (2=>"Andri","dua"=>"2",'tiga'=>3,true=>true,
5=>"sembilan");
echo $coba[2]; //Andri
echo "<br />";
echo $coba["dua"]; //2
echo "<br />";
echo $coba['tiga']; //3
echo "<br />";
echo $coba[true]; //1 (true di konversi menjadi 1)
echo "<br />";
echo $coba[5]; // sembilan
?>
```

Untuk PHP versi 5.4.x keatas, PHP menyediakan cara yang lebih singkat dalam pembuatan array, atau disebut dengan '*short syntax array*'.

### 5.2.2. Menampilkan Array dengan Perulangan

Apabila data array yang akan ditampilkan memiliki elemen yang banyak, maka anda dapat menggunakan perulangan.

#### 1. Contoh Menggunakan for

```
for($i=0;$i<count($gorengan);$i++)
echo "Indek $i : $gorengan[$i]";
```

## 2. Menggunakan foreach

```
foreach($gorengan as $nama)
echo "$nama<br>";
```

Berikut contoh penggunaannya:

NamaFile:array52.php

```
<?php
// pembuatan array
$gorengan = ["Bala Bala", "Gehu", "Cireng", "Comro", "Pisang Aroma"];
// pengaksesan array
echo $gorengan[1]; //Gehu
echo "<br />";
echo $gorengan[2]; //Cireng
echo "<br />";
echo $gorengan[3]; //Comro

echo "<br>";
//menampilkan indek dan nama
echo count($gorengan);
for($i=0;$i<count($gorengan);$i++)
echo "Indek $i : $gorengan[$i]<br>";
echo "<br>";
foreach($gorengan as $nama)
echo "$nama<br>";

?>
```

Perhatikan bahwa tidak perlu membuat keyword 'array', tapi langsung membuat tanda kurung siku. Dan, seperti biasa, index key array dimulai dari 0. Sehingga pemanggilan \$gorengan[1] menghasilkan "Gehu", bukan "Bala Bala".

Untuk menghitung berapa jumlah array dapat digunakan fungsi **count**, Contoh diatas:

```
count($gorengan) // 5
```

### 5.3 Associative Array

Array asosiatif adalah array yang menggunakan indek berupa string, Pengindeksan array dengan string dapat sangat berguna dalam kasus dimana anda harus menyimpan nama elemen dan nilai elemennya.

#### 1. Menggunakan constructor array()

```
<?php
```

```
$harga = array("Gehu"=>"1000", "Bala Bala"=>"500",  
"Batagor"=>"2500", "Kopi"=>"3000");  
//menampilkan asosiatif array  
echo $harga['Gehu']; //1000  
echo $harga['Kopi']; //3000  
echo "<hr>";  
//Menampilkan seluruh elemen array  
foreach($harga as $elemen)  
echo "$elemen<br>";  
echo "<hr>";  
//Menampilkan seluruh data dengan nama dan harganya  
foreach($harga as $nama => $hargaGorengan)  
echo "$nama : $hargaGorengan<br>";  
?>
```

Hasil koding di atas adalah seperti berikut:

```
10003000
```

```
1000  
500  
2500  
3000
```

```
Gehu : 1000  
Bala Bala : 500  
Batagor : 2500  
Kopi : 3000
```

**Contoh Lainnya:**

```
$ibukota=array("Jabar"=>"Bandung", "Jatim"=>"Surabaya",  
"Jateng"=>"Semarang",  
"Bali"=>"Denpasar");  
echo $ibukota["Jabar"]; //maka yang tampil adalah Bandung
```

## 2. Menggunakan tanda kurung siku

```
$ibukota["Jabar"]="Bandung";  
$ibukota["Jatim"]="Surabaya";  
$ibukota["Jateng"]="Semarang";  
$ibukota["Bali"]="Denpasar";  
echo $ibukota["Jateng"]; //maka yang tampil adalah Semarang  
$harga["Gehu"]=1000;  
$harga["Bala Bala"]=500;  
$harga["Batagor"]=2500;  
$harga["Kopi"]=3000;  
echo $harga["Batagor"]; //maka akan tampil 2500
```

#### 5.4. Array Multi Dimensi

Array multidimensi adalah array yang berisi satu atau lebih array. PHP mendukung array multidimensi yang terdiri dari dua, tiga, empat, lima, atau lebih tingkatan. Namun, array lebih dari tiga tingkat sulit dikelola bagi kebanyakan orang, Array dua dimensi adalah array dalam array, array tiga dimensi adalah array dalam array dalam array.

Tabel 5.1. Penjualan Mobil

MOBIL	STOK	TERJUAL
Volvo	22	18
BMW	15	13
Saab	5	2
Land Rover	17	15

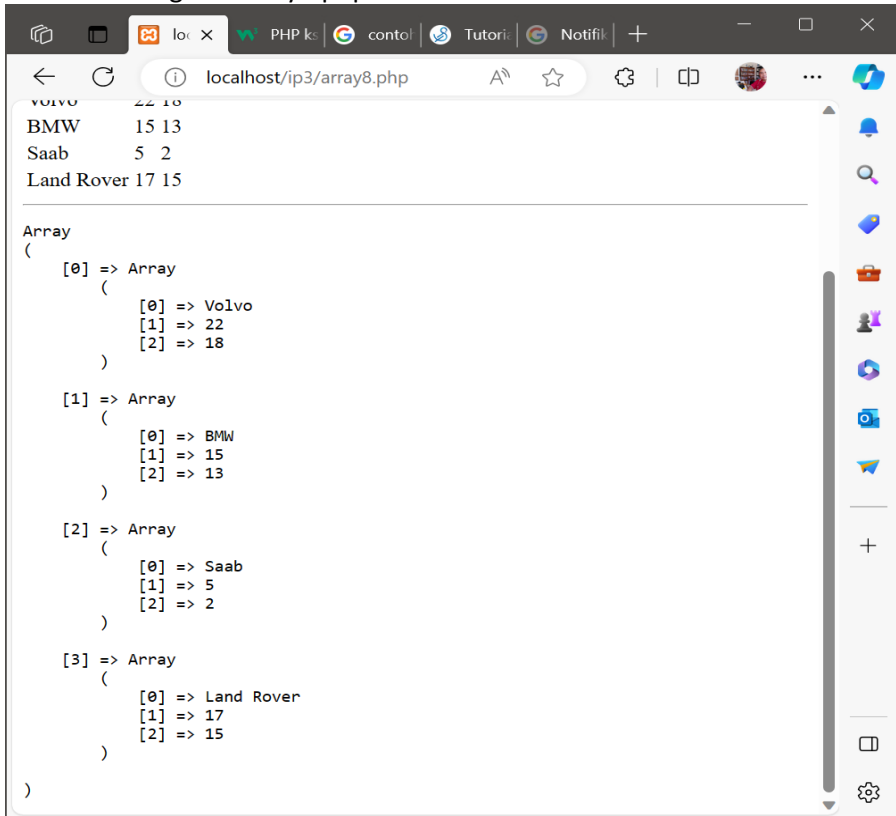
#### Contoh Array 2 Dimensi – array8.php

```
<?php
$mobil = array (
array("Volvo",22,18),
array("BMW",15,13),
array("Saab",5,2),
array("Land Rover",15,15));

echo "<table>";
echo "<th>MOBIL<th>STOK<th>TERJUAL";
echo "<tr><td>".$cars[0][0]."<td>".$cars[0][1]."<td>".$cars[0][2];
echo "<tr><td>".$cars[1][0]."<td>".$cars[1][1]."<td>".$cars[1][2];
echo "<tr><td>".$cars[2][0]."<td>".$cars[2][1]."<td>".$cars[2][2];
echo "<tr><td>".$cars[3][0]."<td>".$cars[3][1]."<td>".$cars[3][2];
echo "</table><hr>";
//menampilkan array 2 dimensi
echo "<table>";
for ($row = 0; $row < 4; $row++) {
    echo "<tr>";
    for ($col = 0; $col < 3; $col++) {
        echo "<td>".$cars[$row][$col];
    }
}
echo "</table><hr>";
echo "<pre>";
    print_r($cars);
```

```
?> echo "</pre>"
```

### Hasil Kode Program array8.php



Gambar 5.1. Menampilkan isi Array

### 5.5. Menambahkan Nilai Array

Untuk menambahkan nilai array kita dapat menggunakan tanda [] tanpa index contoh \$nama[]=nilai, sesuaikan dengan dimensi array yang akan ditambahkan nilai contoh:

```
$buku=array("Pascal","Delphi","Java");  
//Menambah data kedalam array 1 dimensi  
$buku []="HTML";
```

```

$daftar_harga_buku=array(
    array("nama_buku"=>"Pascal","harga"=>100000),
    array("nama_buku"=>"Delphi","harga"=>50000),
    array("nama_buku"=>"Java","harga"=>60000),
);
//Menambah data kedalam array 2 dimensi
$daftar_harga_buku[]=array("nama_buku"=>"HTML","harga"=>50000);

```

Kode Program Lengkapnya adalah sebagai berikut:

NamaFile:array12.php

```

<?php
echo "<h1>Menambahkan Nilai Ke Array</h1>";
echo "<h2>Array Satu Dimensi</h2>";
$buku=array("Pascal","Delphi","Java");
$buku[]="Kimia"; //Menambah data ke array 1 dimensi
foreach ($buku as $indek => $namabuku) {
echo "Buku $namabuku indek_ke $indek <br/>";
}
echo "<h2>Array Dua Dimensi</h2>";
$daftar_harga_buku=array(
    array("nama_buku"=>"Pascal","harga"=>100000),
    array("nama_buku"=>"Delphi","harga"=>50000),
    array("nama_buku"=>"Java","harga"=>60000),
);
$daftar_harga_buku[]=array("nama_buku"=>"HTML","harga"=>50000);
//Menambah data ke array 2 dimensi
foreach ($daftar_harga_buku as $data) {
echo "Buku ".$data["nama_buku"]." harganya = ".$data["harga"]." <br/>";
}
?>

```



Gambar 5.2. Array 2 Dimensi

## 5.6. Menghapus Array

Apabila ingin menghapus satu nilai atau keseluruhan dari array maka dapat menggunakan fungsi `unset()`.

```
$buku=array("Pascal","Delphi","Java");
unset($buku[1]); //delphi dihapus
$daftar_harga_buku=array(
    array("nama_buku"=>"Pascal","harga"=>100000),
    array("nama_buku"=>"Delphi","harga"=>50000),
    array("nama_buku"=>"Java","harga"=>60000),
);
unset($daftar_harga_buku[1]); Delphi di Hapus
```

## 5.5 Fungsi Bawaan Array

PHP menyediakan sejumlah fungsi bawaan yang dapat digunakan untuk melakukan operasi pada array. Berikut adalah beberapa fungsi bawaan yang sering digunakan untuk melakukan operasi pada array di PHP:

1. `count()`: Mengembalikan jumlah elemen dalam sebuah array.

```
$arr = [1, 2, 3, 4, 5];  
echo count($arr); // Output: 5
```

## 2. array\_push(): Menambahkan satu atau lebih elemen ke akhir array.

```
$minuman = ["Kopi", "Teh", "Soda"];  
array_push($minuman, "Air Putih", "Cendol");  
print_r($minuman);  
//Array ( [0] => Kopi [1] => Teh [2] => Soda [3] => Air Putih [4] => Cendol )
```

## 3. array\_pop(): Digunakan untuk menghapus elemen terakhir dari sebuah array dan mengembalikan nilai elemen tersebut. Berikut adalah contoh penggunaannya:

```
$buah = array("Apel", "Pisang", "Jeruk");  
$buahTerakhir = array_pop($buah);  
print_r($buah); // Array sekarang hanya berisi "Apel" dan "Pisang"
```

Dalam contoh di atas, `array_pop($buah)` akan menghapus "Jeruk" dari array `$buah` dan mengembalikan nilai "Jeruk". Setelah pemanggilan fungsi, array `$buah` hanya akan berisi Apel dan Pisang

## 4. array\_shift(): Menghapus dan mengembalikan elemen pertama dari array.

```
$arr = [1, 2, 3];  
$firstElement = array_shift($arr);  
echo $firstElement; // Output: 1
```

## 5. array\_unshift(): Menambahkan satu atau lebih elemen ke awal array.

```
$arr = [2, 3];  
array_unshift($arr, 1);  
print_r($arr); // Output: Array ( [0] => 1 [1] => 2 [2] => 3 )
```

## 6. array\_slice(): Mengembalikan sebagian dari array.

```
$arr = [1, 2, 3, 4, 5];  
$slicedArray = array_slice($arr, 2); // Mengambil elemen dari indeks ke-2  
sampai akhir array  
print_r($slicedArray); // Output: Array ( [0] => 3 [1] => 4 [2] => 5 )
```

5. `array_splice()`: Menghapus dan/atau mengganti elemen dari array.

```
$arr = [1, 2, 3, 4, 5];  
array_splice($arr, 2, 2); // Menghapus 2 elemen dari indeks ke-2  
print_r($arr); // Output: Array ( [0] => 1 [1] => 2 [2] => 5 )
```

Ini hanya beberapa contoh fungsi bawaan yang sering digunakan dalam PHP untuk melakukan operasi pada array. PHP memiliki banyak fungsi bawaan lainnya yang dapat Anda gunakan sesuai dengan kebutuhan Anda.

## Contoh Penggunaan Array dalam Program

```
<?php
// pembuatan array
$namabarang = ["Komputer","Laptop","Printer","Mouse","Keyboard"];
$harga = [1000,2000,3000,4000,5000];
$jumlah = [3,2,6,4,5];

// pengaksesan array
echo $namabarang[1];
echo "<br />";
echo $namabarang[2];
echo "<br />";
echo $namabarang[3];
echo "<br />";
echo $harga[0];
echo "<br />";
echo "<table>";
echo "<tr><th>NO<th>NAMA BARANG<th>HARGA<th>JUMLAH<th>TOTAL";
for($i=0;$i<4;$i++)
{
echo "<tr><td>",$i+1,"<td>$namabarang[$i]<td>$harga[$i]<td>$jumlah[$i]<td>";
$total=$harga[$i]*$jumlah[$i];
echo $total;
}
?>
```

NO	NAMA BARANG	HARGA	JUMLAH	TOTAL
1	Komputer	1000	3	3000
2	Laptop	2000	2	4000
3	Printer	3000	6	18000
4	Mouse	4000	4	16000

Gambar 5.3. Menampilkan array

## 5.8. Mengurutkan Array

Elemen-elemen yang ada dalam array dapat diurutkan berdasarkan abjad atau berdasarkan angka, baik diurutkan secara menurun atau menaik.

Pada bab ini, kita akan membahas fungsi pengurutan array PHP berikut:

- **sort()**- mengurutkan array dalam urutan menaik
- **rsort()**- mengurutkan array dalam urutan menurun
- **asort()**- mengurutkan array asosiatif dalam urutan menaik, berdasarkan nilainya
- **ksort()**- mengurutkan array asosiatif dalam urutan menaik, berdasarkan kuncinya
- **arsort()**- mengurutkan array asosiatif dalam urutan menurun, berdasarkan nilainya
- **krsort()**- mengurutkan array asosiatif dalam urutan menurun, berdasarkan kuncinya

NamaFile:arraySort1.php

```
<?php
echo "<b>Fungsi sort() </b><hr>";
$kota = array("Cimahi","Tasikmalaya","Bandung", "Garut" );
sort($kota);
$i = 0;
do {
    echo $i+1 . ".$kota[$i] <br>";
    $i++;
} while ($i < 4);
echo "<b><hr>Fungsi rsort() </b><br>";
rsort($kota);
$i = 0;
do {
    echo $i+1 . ".$kota[$i] <br>";
    $i++;
} while ($i < 4);

echo "<b><hr>Fungsi asort() </b><br>";
$usia = array("Ridwan"=>"15", "Eddi"=>"25",
    "Eni"=>"23", "Gaga"=>"11");
asort($usia);
foreach($usia as $nama => $x_value) {
    echo "Key=" . $nama . ", Value=" . $x_value;
    echo "<br>";
}

echo "<b><hr>Fungsi ksort() </b><br>";
ksort($usia);
foreach($usia as $nama => $x_value) {
```

```
    echo "Key=" . $nama . ", Value=" . $x_value;
    echo "<br>";
}

echo "<b><hr>Fungsi arsort() </b><br>";
arsort($usia);
foreach($usia as $nama => $x_value) {
    echo "Key=" . $nama . ", Value=" . $x_value;
    echo "<br>";
}

echo "<b><hr>Fungsi krsort() </b><br>";
krsort($usia);
foreach($usia as $nama => $x_value) {
    echo "Key=" . $nama . ", Value=" . $x_value;
    echo "<br>";
}
?>
```

**Fungsi sort()**

---

1. Bandung  
2. Cimahi  
3. Garut  
4. Tasikmalaya

---

**Fungsi rsort()**

1. Tasikmalaya  
2. Garut  
3. Cimahi  
4. Bandung

---

**Fungsi asort()**

Key=Gaga, Value=11  
Key=Ridwan, Value=15  
Key=Eni, Value=23  
Key=Eddi, Value=25

---

**Fungsi ksort()**

Key=Eddi, Value=25  
Key=Eni, Value=23  
Key=Gaga, Value=11  
Key=Ridwan, Value=15

---

**Fungsi arsort()**

Key=Eddi, Value=25  
Key=Eni, Value=23  
Key=Ridwan, Value=15  
Key=Gaga, Value=11

---

**Fungsi krsort()**

Key=Ridwan, Value=15  
Key=Gaga, Value=11

Key=Eni, Value=23  
Key=Eddi, Value=25

## DAFTAR PUSTAKA

1. Suryana, Taryana (2024) [Pengantar Pemrograman WEB \(HTML, CSS, PHP, MYSQL\)](#) xviii, 181 hlm Deepublish, ISBN:978-623-02-8181-5
2. Suryana, Taryana; Koesheryatin (2014), (2023), [Aplikasi Internet Menggunakan HTML, CSS & Java Script](#), Elexmedia Komputindo, Jakarta
3. Suryana, Taryana (2010) [Membuat Web Pribadi dan Bisnis dengan HTML.](#), Gava Media, Yogyakarta
4. Suryana, Taryana; Sarwono, Jonathan (2007) [E-COMMERCE MENGGUNAKAN PHP DAN MYSQL](#), Graha Ilmu, Yogyakarta
5. Wong Jony. 2010. *Internet Marketing for Beginners*. Jakarta: PT Elex Media Komputindo.
6. Kumar, Anil and Manoj Kumar Dash. 2013. *Constructing a Measurement in Service Quality for Indian Banks: Structural Equation Modeling Approach*. *Journal of Internet Banking and Commerce*, April 2013, vol. 18, no.1
7. Siregar, Riki R. 2010. *Strategi Meningkatkan Persaingan Bisnis Perusahaan dengan Penerapan E-commerce*
8. <https://www.php.net/manual/en/book.mysql.php>
9. <https://www.w3schools.com>
10. <https://tekno.kompas.com/read/2022/09/26/10150007/pengertian-e-commerce-beserta-jenis-contoh-dan-manfaatnya>
11. <https://meenta.net/pengertian-e-commerce/>
12. <https://www.rumahweb.com/journal/session-adalah/>
13. <https://rizkypratama.id/blog/2019/11/01/membuat-session-login-pada-php/>
14. <https://blog.skillacademy.com/supplier-adalah>
15. <https://chat.openai.com/>
16. <http://www.grahailmu.co.id/previewpdf/978-979-756-182-6-231.pdf>
17. <https://www.w3schools.com/html>, September, 2025
18. [https://repository.dinus.ac.id/docs/ajar/Pert\\_5\\_Percabangan\\_Bersarang.pptx](https://repository.dinus.ac.id/docs/ajar/Pert_5_Percabangan_Bersarang.pptx)

