

**Jobsheet 4 – PHP**  
**Pemrograman Web**



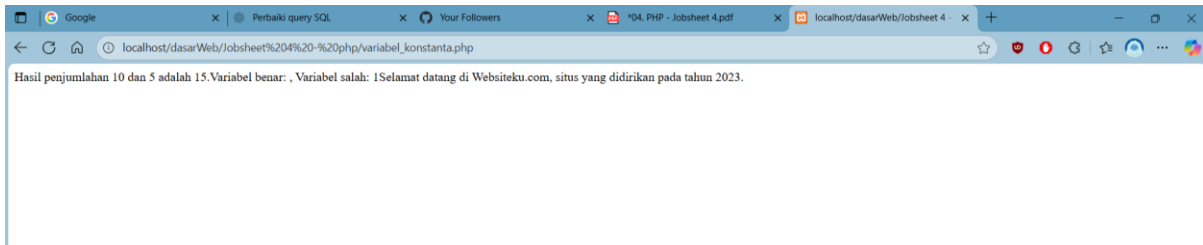
**244107060144**

**Maria Savira**

**D-IV Sistem Informasi Bisnis / 2C**

## Praktikum 1 – Variabel dan Konstanta

1. Deklarasi variabel menggunakan tanda \$ sedangkan konstanta menggunakan define(). Lalu variabel dapat dirubah isinya, sedangkan konstanta tidak bisa. Keduanya dapat memuat berbagai tipe data, termasuk String, boolean, hingga integer. Boolean di dalam PHP outputnya bukan berupa true atau false, melainkan 1 dan 0.



## Praktikum 2 – Penggunaan Tipe Data

2. Tipe data dapat diberlakukan operasi seperti tambah, kali, dan kurang. Untuk mengecek tipe data maupun diterimanya suatu data atau tidak, dapat menggunakan var\_dump(\$variabel). Selain itu, variabel di dalam PHP merupakan loosely typed. Yaitu variabel yang dapat mendeteksi tipe datanya sendiri berdasarkan value tanpa harus mendeklarasikan tipe data.

Dalam memanggil maupun menginisialisasi tipe data String, dapat menggunakan “...” maupun ‘...’. Terakhir, tipe data Array dibaca dari indeks ke-0, bukan ke-1.

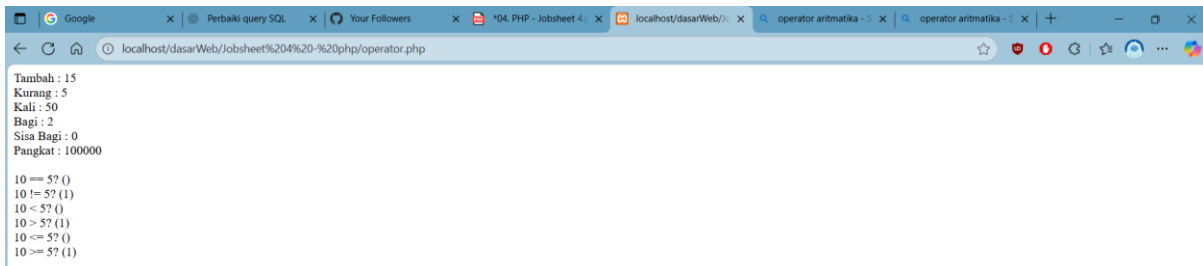


## Praktikum 3 – Penggunaan Operator PHP

- 3.1. Dalam PHP, terdapat operator yang dapat digunakan untuk melakukan operasi pada dua atau lebih variabel untuk melakukan perhitungan. Operator seperti tambah (+), kali (x), kurang (-), hingga bagi (/) dan sebagainya merupakan operator aritmatika.



3.2. Terdapat juga operator perbandingan dalam PHP yang dapat digunakan untuk membandingkan 2 variabel. Seperti mana variabel yang lebih besar hingga apakah value dari kedua variabel tersebut sama. Operator perbandingan akan menghasilkan nilai boolean yaitu true (1) or false (0).

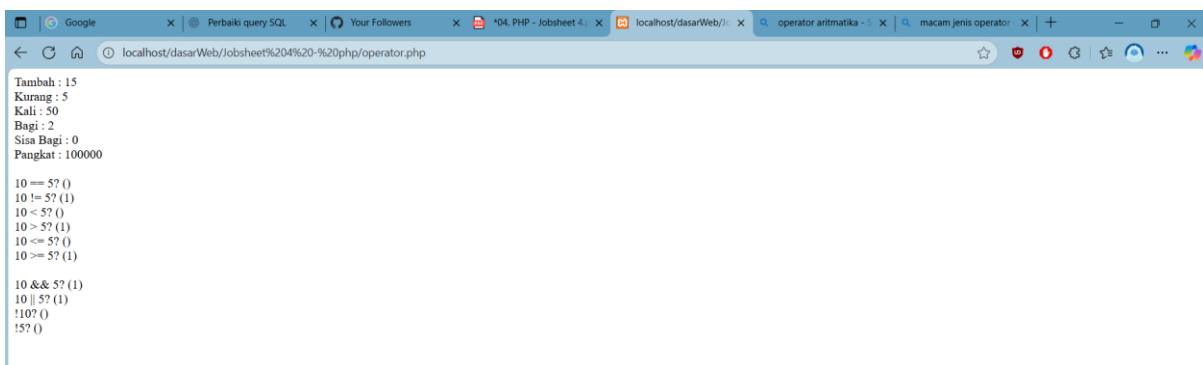


The screenshot shows a web browser window with the URL `localhost/dasarWeb/jobsheet%204%20-%20php/operator.php`. The page displays the following PHP code and its output:

```
Tambah : 15
Kurang : 5
Kali : 50
Bagi : 2
Sisa Bagi : 0
Pangkat : 100000

10 == 5? (0)
10 != 5? (1)
10 < 5? (0)
10 > 5? (1)
10 <= 5? (0)
10 >= 5? (1)
```

3.3. Terdapat juga operator logika seperti AND, OR, dan NOT. Operator logika digunakan untuk menggabungkan beberapa kondisi.



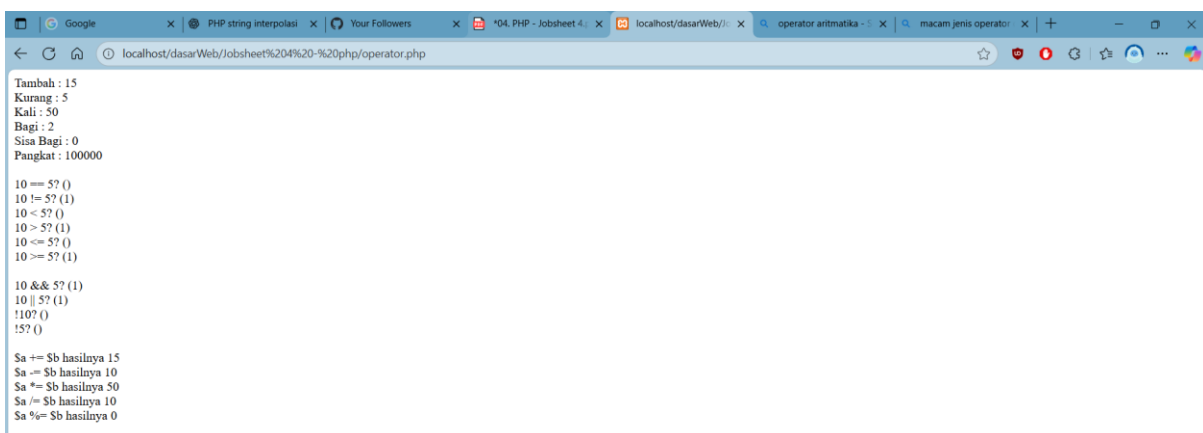
The screenshot shows a web browser window with the URL `localhost/dasarWeb/jobsheet%204%20-%20php/operator.php`. The page displays the following PHP code and its output:

```
Tambah : 15
Kurang : 5
Kali : 50
Bagi : 2
Sisa Bagi : 0
Pangkat : 100000

10 == 5? (0)
10 != 5? (1)
10 < 5? (0)
10 > 5? (1)
10 <= 5? (0)
10 >= 5? (1)

10 && 5? (1)
10 || 5? (1)
!10? (0)
!5? (0)
```

3.4. Merupakan contoh penggunaan operator assignment gabungan (`+=`, `-=`, `*=`, `/=`, `%=`) pada PHP, di mana setiap operator tidak hanya melakukan perhitungan tetapi juga langsung memperbarui nilai variabel \$a. Akibatnya, hasil yang ditampilkan bersifat berurutan karena nilai \$a berubah di setiap langkah, misalnya `$a += $b` membuat \$a bertambah, lalu hasil perubahan itu menjadi dasar untuk operasi berikutnya hingga menghasilkan keluaran akhir sesuai urutan eksekusi.



The screenshot shows a web browser window with the URL `localhost/dasarWeb/jobsheet%204%20-%20php/operator.php`. The page displays the following PHP code and its output:

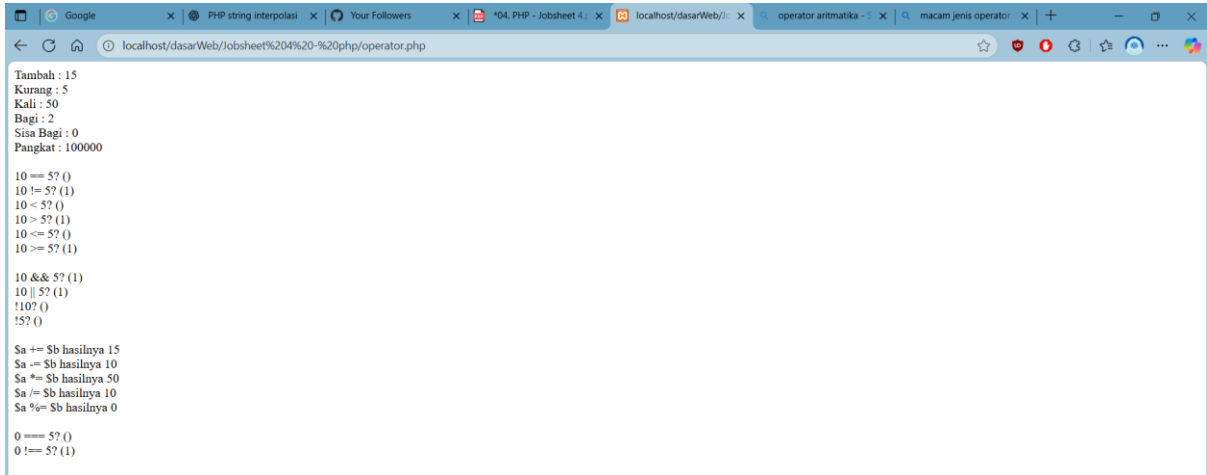
```
Tambah : 15
Kurang : 5
Kali : 50
Bagi : 2
Sisa Bagi : 0
Pangkat : 100000

10 == 5? (0)
10 != 5? (1)
10 < 5? (0)
10 > 5? (1)
10 <= 5? (0)
10 >= 5? (1)

10 && 5? (1)
10 || 5? (1)
!10? (0)
!5? (0)

$a += $b hasilnya 15
$a -= $b hasilnya 10
$a *= $b hasilnya 50
$a /= $b hasilnya 10
$a %= $b hasilnya 0
```

3.5. Kode tersebut mendemonstrasikan penggunaan operator identik (===) dan ketidak-identikan (!==) pada PHP, yang berfungsi membandingkan nilai sekaligus tipe data dari dua variabel. Pada \$a = 0 dan \$b = 5, ekspresi \$a === \$b bernilai false (ditampilkan sebagai 0) karena meskipun keduanya bertipe sama (integer), nilainya berbeda. Sebaliknya, ekspresi \$a !== \$b bernilai true (ditampilkan sebagai 1) karena memang \$a dan \$b tidak identik.



```

Tambah : 15
Kurang : 5
Kali : 50
Bagi : 2
Sisa Bagi : 0
Pangkat : 100000

10 == 5? (0)
10 != 5? (1)
10 < 5? (0)
10 > 5? (1)
10 <= 5? (0)
10 >= 5? (1)

10 && 5? (1)
10 || 5? (1)
!10? (0)
!5? (0)

Sa += Sb hasilnya 15
Sa -= Sb hasilnya 10
Sa *= Sb hasilnya 50
Sa /= Sb hasilnya 10
Sa %= Sb hasilnya 0

0 === 5? (0)
0 !== 5? (1)

```

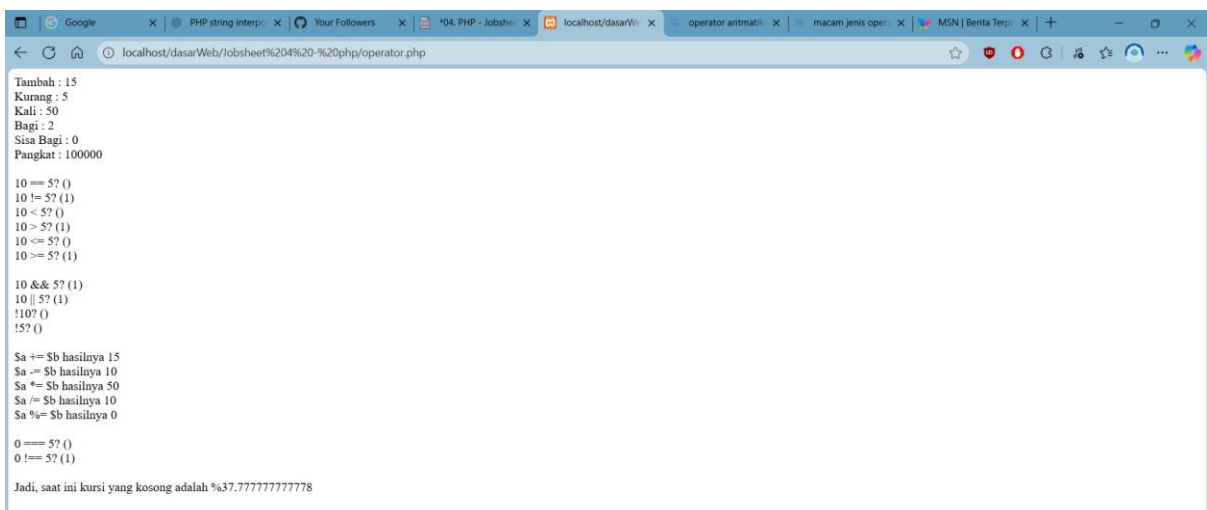
### 3.6. Kode Program :

```

61  $totalKursi = 45;
62  $kursiDitempati = 28;
63  $kursiKosong = ($totalKursi - $kursiDitempati) / $totalKursi * 100;
64
65  echo "<br>Jadi, saat ini kursi yang kosong adalah %$kursiKosong";
66  ?>

```

Output :



```

Tambah : 15
Kurang : 5
Kali : 50
Bagi : 2
Sisa Bagi : 0
Pangkat : 100000

10 == 5? (0)
10 != 5? (1)
10 < 5? (0)
10 > 5? (1)
10 <= 5? (0)
10 >= 5? (1)

10 && 5? (1)
10 || 5? (1)
!10? (0)
!5? (0)

Sa += Sb hasilnya 15
Sa -= Sb hasilnya 10
Sa *= Sb hasilnya 50
Sa /= Sb hasilnya 10
Sa %= Sb hasilnya 0

0 === 5? (0)
0 !== 5? (1)

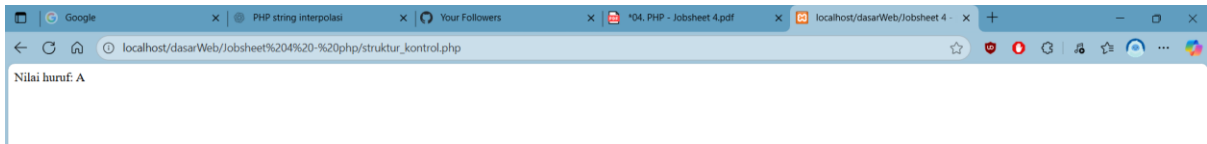
Jadi, saat ini kursi yang kosong adalah %37.777777777778

```

## Praktikum 4 – Penggunaan Struktur Kontrol Pada PHP

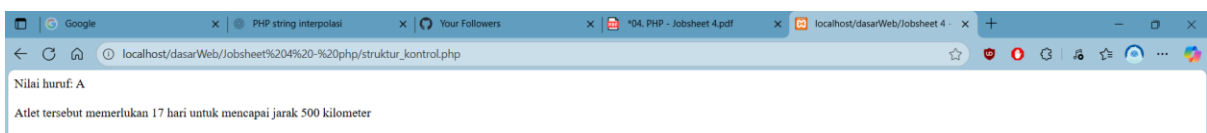
- 4.1. Outputnya adalah nilai huruf: A karena memenuhi persyaratan if yang pertama. Yaitu value nilaiNumerik lebih besar atau sama dengan 90 dan lebih kecil atau sama dengan 100.

Apabila syarat pemilihan pertama terpenuhi, maka kode program pertama yang akan dijalankan.



- 4.2. While merupakan kode program untuk perulangan yang tidak diketahui jumlah pengulangannya. Yang mana pada kasus ini, perulangan akan berhenti jika value dari jarakSaatIni lebih besar daripada jarakTarget. Setiap kali mengulang, maka hari akan bertambah 1.

Diperoleh perulangan sebanyak 17 kali supaya atlet dapat mencapai jarak 500 meter.



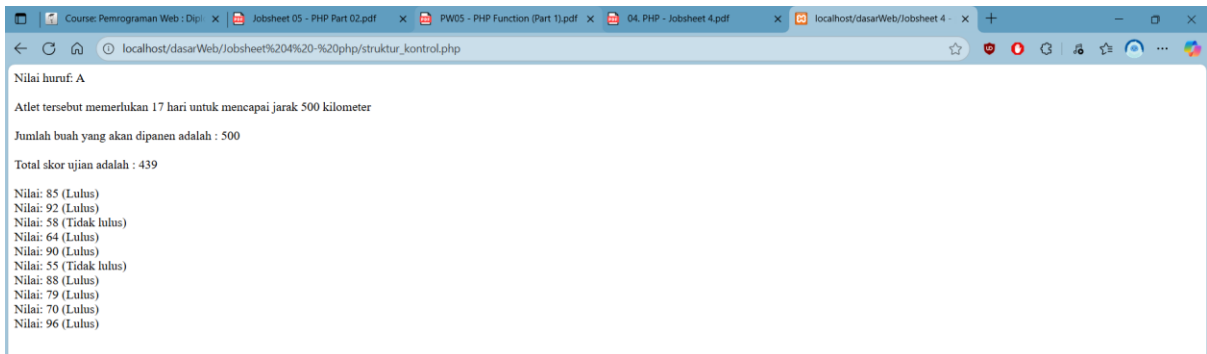
- 4.3. Program kode tersebut bertujuan untuk menghitung jumlahBuah yang didapat dari tanamanPerLahan dikali buahPerTanaman, yang mana rumus ini akan dilakukan berulang kali sebanyak jumlahLahan. Dan perulangan yang digunakan adalah perulangan for karena sudah jelas dan diketahui jumlah perulangannya. Maka didapat hasil jumlah buahnya adalah 500.



- 4.4. Program kode tersebut bertujuan untuk menghitung totalSkor dengan menjumlahkan variabel tersebut dengan setiap skor yang ada. Yakini menggunakan perulangan foreach, dengan foreach maka iterasi akan dilakukan dengan memanggil semua elemen satu-persatu hingga semua elemen tidak bersisa. Hasil total skor adalah 439.



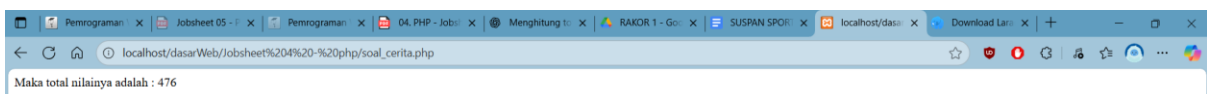
4.5. Melakukan perulangan terhadap setiap nilaiSiswa dengan pemilihan apabila nilaiSiswa dibawah 60 maka akan dinyatakan tidak lulus. Hal ini membuktikan bahwa perulangan dengan pemilihan bisa untuk dilakukan.



4.6. Kode program :

```
1  <?php
2  #variabel yg dipakai untuk soal cerita 4.6
3  $nilaiSiswa = [85, 92, 58, 64, 90, 55, 88, 79, 70, 96];
4
5  #soal cerita 4.6
6  sort($nilaiSiswa);
7  $totalNilai = 0;
8
9  for ($i = 0; $i < count($nilaiSiswa); $i++){
10     if ($i == 0 || $i == 1 || $i == 8 || $i == 9){
11         continue;
12     }
13     else {
14         $totalNilai += $nilaiSiswa[$i];
15     }
16 }
17
18 echo "Maka total nilainya adalah : $totalNilai";
19
```

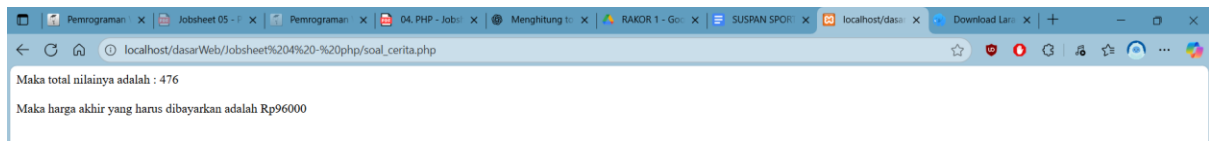
Output :



4.7. Kode Program :

```
20 #soal cerita 4.7
21 $hargaProduk = 120000;
22 $diskon = 0.2;
23 $hargaAkhir = 0;
24
25 if ($hargaProduk > 100000){
26     $hargaAkhir = $hargaProduk - ($hargaProduk * $diskon);
27 }
28
29 echo "<br><br>Maka harga akhir yang harus dibayarkan adalah Rp$hargaAkhir";
```

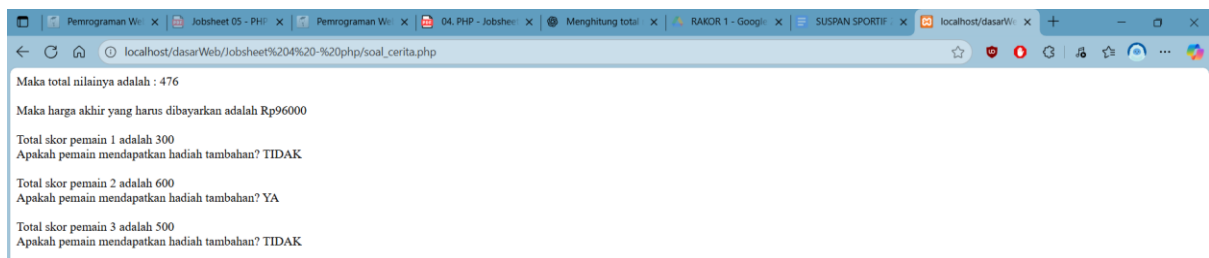
Output :



#### 4.8. Kode Program :

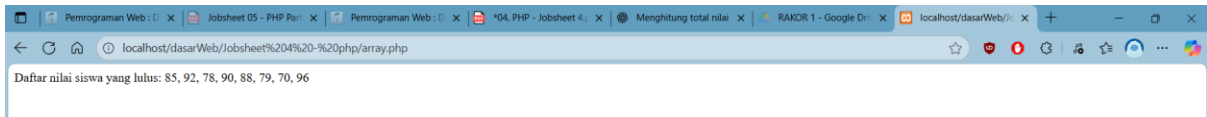
```
31 #soal_cerita 4.8
32 $totalSkor = [300, 600, 500];
33
34 echo "<br><br>";
35
36 $i = 0;
37 while ($i < count($totalSkor)){
38     $skor = $totalSkor[$i];
39     $hadiah = ($skor > 500) ? "YA" : "TIDAK";
40
41     echo "Total skor pemain " . ($i + 1) . " adalah $skor<br>";
42     echo "Apakah pemain mendapatkan hadiah tambahan? $hadiah<br><br>";
43
44     $i++;
45 }
46 ?>
```

Output :



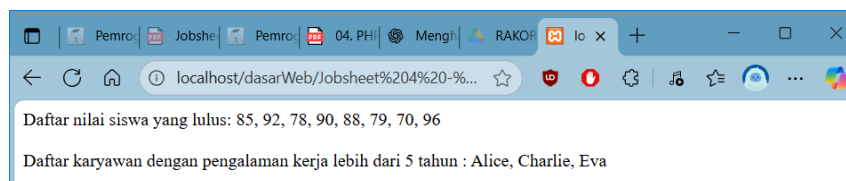
## Praktikum 5 - Penggunaan Array Pada PHP

- 5.1. Kode program tersebut merupakan implementasi dari array 1 dimensi. Melakukan perulangan foreach terhadap semua nilaiSiswa yang ada, dan pada setiap iterasinya akan dicek, apakah nilaiSiswa tersebut 70 atau lebih dari 70. Jika iya, maka nilai tersebut akan masuk ke dalam array nilaiLulus. Lalu dalam outputnya, terdapat implode yaitu sintaks pemisah tiap variabel array.



- 5.2. Kode program tersebut merupakan implementasi dari array 2 dimensi. Melakukan perulangan foreach pada setiap elemen daftarKaryawan, yang mana pada setiap iterasinya akan dicek apakah kolom ke-1 (pengalaman kerja) pada setiap karyawan valuenya lebih dari 5 atau tidak. Jika iya, maka kolom ke-0 (nama karyawan) akan dimasukkan ke dalam array karyawanPengalamanLimaTahun.

Kemudian array karyawanPengalamanLimaTahun dipanggil dengan implode untuk mengatur format bagaimana setiap elemen dipanggil.



- 5.3. Kode program tersebut merupakan implementasi dari array asosiatif, yang mana nama dari array tersebut adalah daftarNilai, dihubungkan dengan Matematika, Fisika, dan Kimia. Yang mana masing-masing dari mata kuliah tersebut memiliki nilai yang merujuk kepada seseorang.

Pada kasus kali ini, yang dipanggil adalah daftarNilai dari mataKuliah Fisika.





#### 5.4. Kode Program :

```
47 #soal cerita 5.4
48 $ujianMatematika = [
49     ['Alice', 85],
50     ['Bob', 92],
51     ['Charlie', 78],
52     ['David', 64],
53     ['Eva', 90],
54 ];
55
56 echo "Daftar nilai ujian matematika : <br>";
57
58 foreach ($ujianMatematika as $nilai){
59     echo "Nama : $nilai[0], Nilai : $nilai[1]<br>";
60 };
61 ?>
```

#### Output :

