**Jobsheet-4: PHP**

**Web Design and Programming Courses**

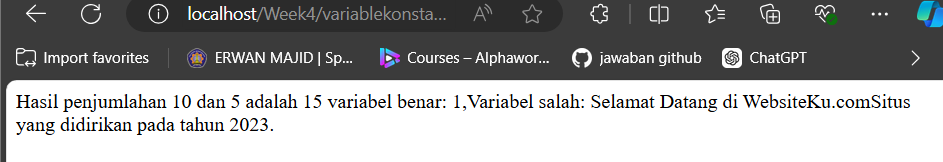
Erwan Majid/08/2i

Link github: <https://github.com/Majid5654/Semester-3/tree/master/Week4>

* **Practical Section 1. Variables and Constants**

1. What do you understand from the use of variables on the file? Record your understanding

below. (Question No. 1)



-first is we declare angka1 = 10,angka 2 = 5

-Their sum is stored in the variable $hasil, and then printed using echo

-The boolean variables $benar and $salah are set to true and false

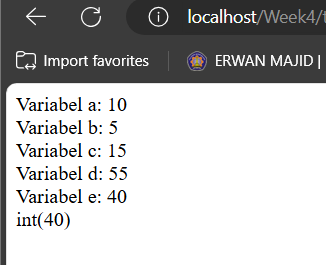
-define("NAMA\_SITUS", "WebsiteKu.com");

define("TAHUN\_PENDIRIAN",2023);

then it print Selamat Datang di " .NAMA\_SITUS. And print websiteku.com

"Situs yang didirikan pada tahun " .TAHUN\_PENDIRIAN. "."; print 2023

* **Practicum 2: Use of Data Types**

1. Explain what you understand from the code below. (Question No. 2)

-$a = 10

$b = 5

$c = $a + 5 results in $c = 15.

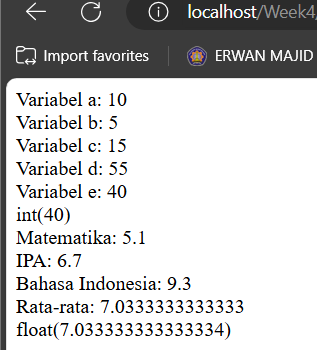
$d = $b + (10 \* 5) calculates $d = 5 + 50 = 55.

$e = $d - $c calculates $e = 55 - 15 = 40.

Each variable is printed using echo

The var\_dump($e) function is used to display the data type and value of $e,

1. Explain what you understand from the code below. (Question No. 3)



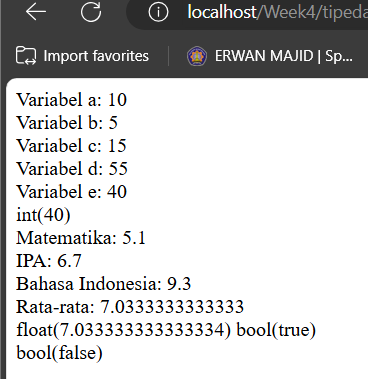
-Scores for three subjects ($nilaiMatematika, $nilaiIPA, and $nilaiBahasaIndonesia) are initialized.

-The average of these three scores is calculated (sum 3 subject and divided by 3) and stored in $rataRata.

-The subject scores and the average are printed.

-var\_dump($rataRata) shows the data type and value of $rataRata

1. Explain what you understand from the code below. (Question No. 4)



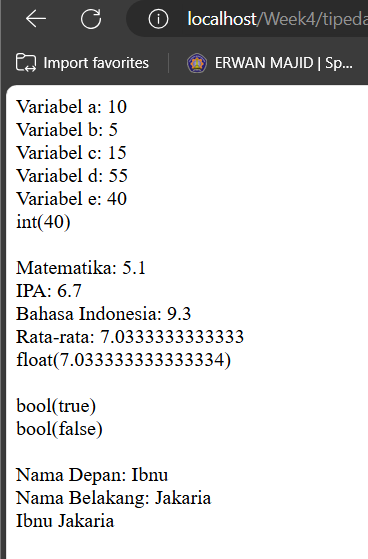
-$apakahSiswaLulus = true; → Represents that the student has passed.

-$apakahSiswaSudahUjian = false; → Represents that the student has not yet taken the exam.

-var\_dump($apakahSiswaLulus); displays the data type and value of $apakahSiswaLulus as:true

-var\_dump($apakahSiswaSudahUjian); displays the data type and value of $apakahSiswaSudahUjian as:false

1. Explain what you understand from the code below. (Question No. 5)



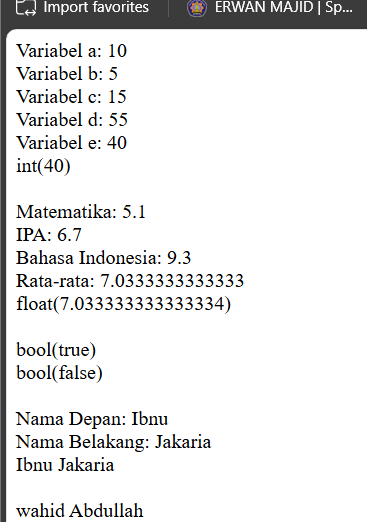
-$namaDepan = "Ibnu"; → Represents the first name.

-$namaBelakang = "Jakaria"; → Represents the last name.

-$namaLengkap = "{$namaDepan} {$namaBelakang}"; → Uses double quotes with interpolation to combine the first and last names with a space in between.

-$namaLengkap2 = $namaDepan .' '. $namaBelakang; combines them using concatenation (.) with single quotes

1. Explain what you understand from the code below. (Question No. 6)

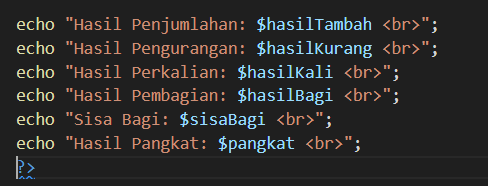


-$listMahasiswa = ["wahid Abdullah", "Elmo bachtiar", "Lendis Fabri"]; initializes an array containing three names

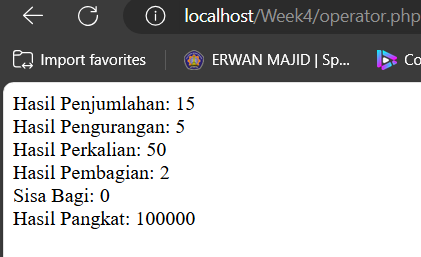
-echo $listMahasiswa[0]; prints the first element of the array, which is "wahid Abdullah"

**Practical Section 3: The Use of PHP Operators**

1. Complete the program code above so that it can display the results and be neat. (Question No.7)

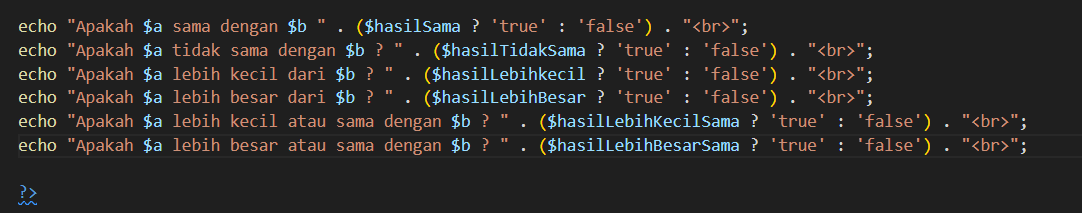


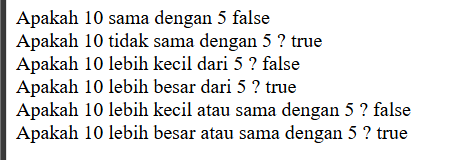
1. Save the file, then open a browser and run localhost/week4/operator.php . Observe what is happening and explain what you understand. (Question No. 8)



-To output the results of the calculations using echo,

1. Complete the program code above so that it can display the results and be neat. Save the file, then open a browser and run /refresh localhost/week4/operator.php Observe what is happening and explain what you understand. (Question No. 9)

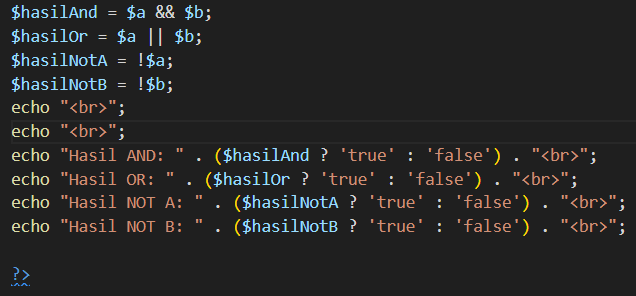


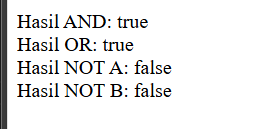


-echo "Apakah $a sama dengan $b? " . ($hasilSama ? 'true' : 'false') . "<br>";

This checks if $hasilSama is true or false and appends "true" or "false" to the output,and so on

1. Complete the program code above so that it can display the results and be neat. Save the file, then open a browser and run /refresh localhost/week4/operator.php Observe what is happening and explain what you understand. (Question No. 10)





-$hasilAnd = $a && $b; checks if both $a and $b are true.

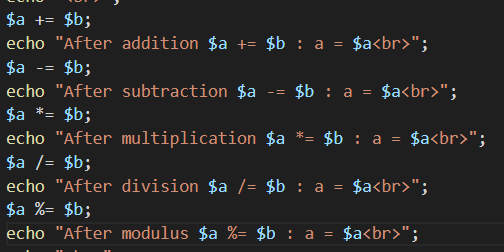
$hasilOr = $a || $b; checks if at least one of $a or $b is true.

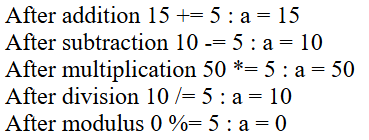
$hasilNotA = !$a; negates the value of $a.

$hasilNotB = !$b; negates the value of $b.

Output:

Each result is displayed using echo, converting boolean values to "true" or "false" for readability.

1. Complete the program code above so that it can display the results and be neat. Save the file, then open a browser and run /refresh localhost/week4/operator.php Observe what is happening and explain what you understand. (Question No. 11)



-Addition ($a += $b): Adds $b to $a.

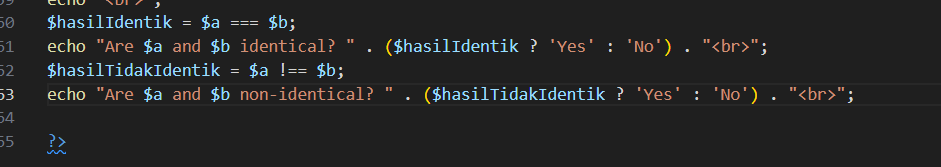
-Subtraction ($a -= $b): Subtracts $b from $a.

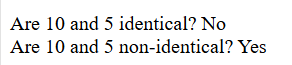
-Multiplication ($a \*= $b): Multiplies $a by $b.

-Division ($a /= $b): Divides $a by $b.

-Modulus ($a %= $b): Sets $a to the remainder when divided by $b.

1. Complete the program code above so that it can display the results and be neat. Save the file, then open a browser and run /refresh localhost/week4/operator.php.Observe what is happening and explain what you understand. (Question No. 12)





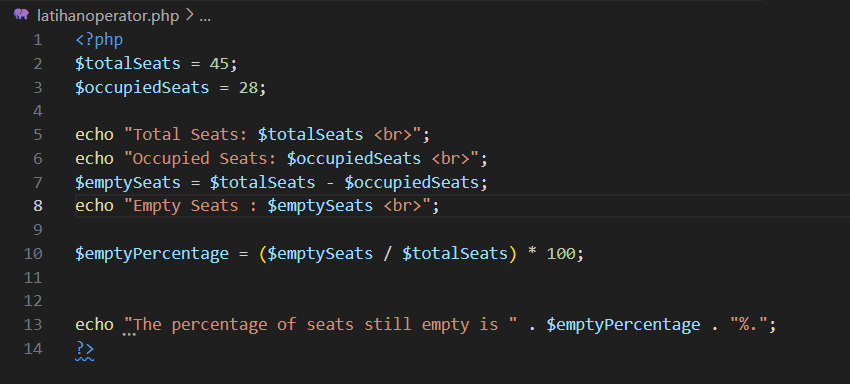
- $hasilIdentik = $a === $b; checks if $a and $b have the same value and type.

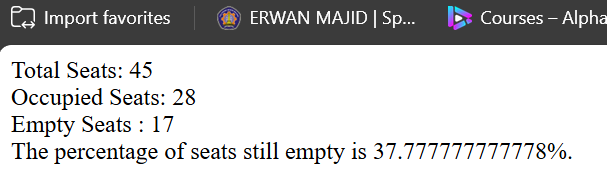
Displays Yes if they are identical, otherwise No.

-$hasilTidakIdentik = $a !== $b; checks if $a and $b are not the same in value or type.

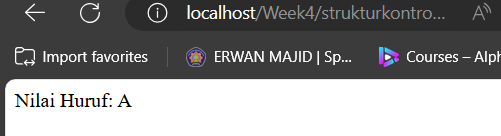
Displays Yes if they are non-identical, otherwise No

1. Create a file with named latihanoperator.php. Write down the program code for step 16 and display the result below along with the program code (Question No. 13)





* **Practical Section 4: The Use of Control Structures in PHP**

1. Note here what you observe from the addition of the program code above. (Question No. 14)

-If $nilaiNumerik is between 90 and 100, it outputs "Nilai Huruf: A".

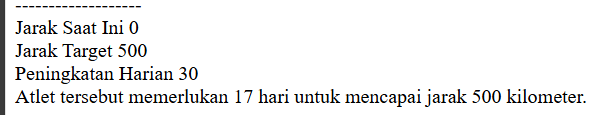
If $nilaiNumerik is between 80 and 89, it outputs "Nilai Huruf: B".

If $nilaiNumerik is between 70 and 79, it outputs "Nilai Huruf: C".

If $nilaiNumerik is below 70, it outputs "Nilai Huruf: D".

Because in the first set $nilaiNumerik is 92, so it prints "Nilai Huruf: A".

1. Note here what you observe from the addition of the program code above. (Question No. 15)



-$jarakSaatIni starts at 0 km.

$jarakTarget is set to 500 km, the goal distance.

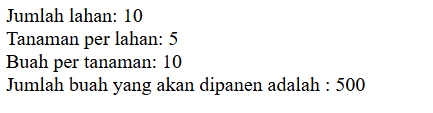
$peningkatanHarian is 30 km, meaning the athlete increases the distance by 30 km per day.

$hari counts the number of days required.

-The while loop will keep adding the daily progress ($peningkatanHarian) to $jarakSaatIni and incrementing $hari until the total distance reaches or exceeds 500 km.

-The expected result is that the athlete will need 17 days to reach 500 km

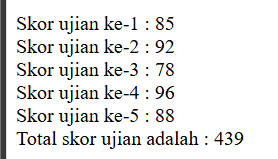
1. Note here what you observe from the addition of the program code above. (Question No. 16)



-The for loop runs for each field ($i from 1 to 10). In each iteration, it adds the product of $tanamanPerLahan and $buahPerTanaman to $jumlahBuah, calculating the total number of fruits.

In this case, the total number of fruits will be 10×5×10=500

1. Note here what you observe from the addition of the program code above. (Question No. 17)



-$skorUjian: An array containing the exam scores [85, 92, 78, 96, 88].

$totalSkor: Initialized to 0, used to accumulate the total score.

$index: Initialized to 0 to track the current exam number.

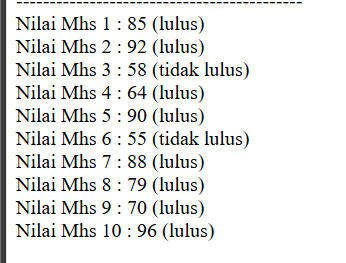
The foreach loop iterates over each score in $skorUjian:

$totalSkor is updated by adding each exam score.

$index is incremented to show the current exam number.

Each score is displayed with its corresponding exam number using echo

1. Note here what you observe from the addition of the program code above. (Question No. 18)



-$nilaiSiswa: An array containing student scores.

$indexx: A counter initialized at 0 to keep track of each student's number.

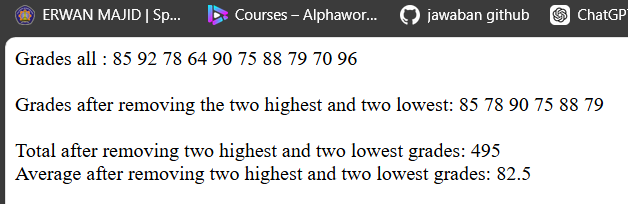
The foreach loop iterates over each score in $nilaiSiswa:

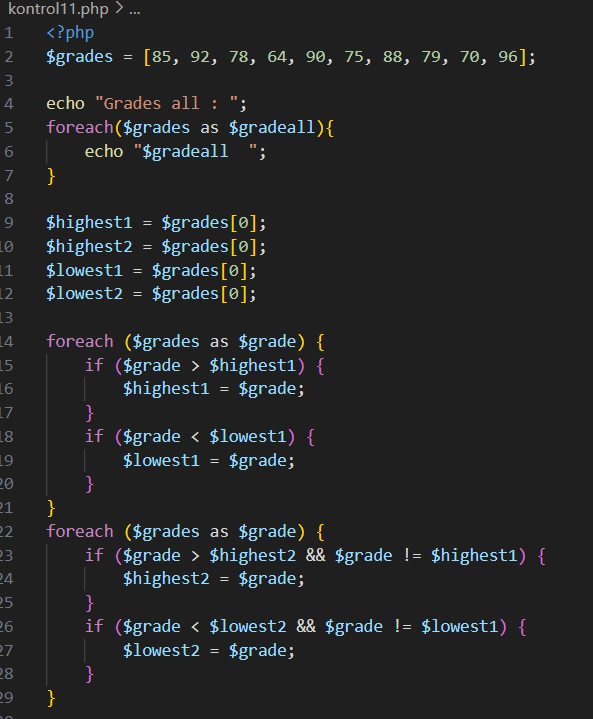
$indexx++: Increments to track the student's number.

if ($nilai < 60): Checks if the score is below 60. If so, it prints the student's score with the label "tidak lulus" (failed) and uses continue to skip to the next student.

If the score is 60 or higher, it prints the score with the label "lulus" (passed).

1. Create a file with the name kontrol1.php. Create the program code for step 21 and display the result below along with the program code (Question No. 19)

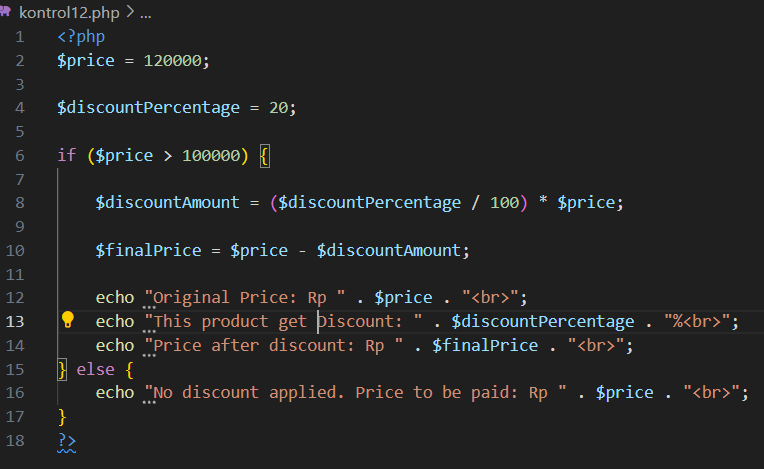




-Full code on github.

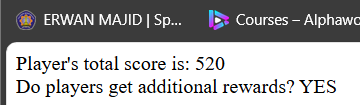
1. Create a file with the name kontrol2.php. Create the program code for step 23 and display the result below along with the program code (Question No. 20)

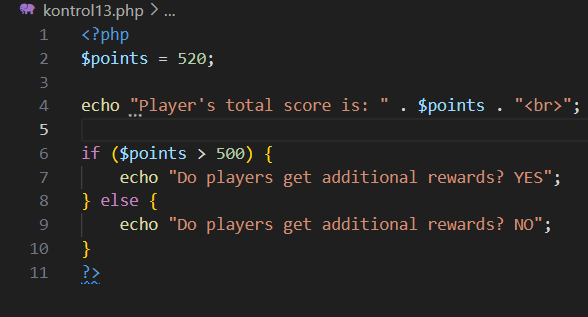




1. Create a file with the name kontrol3.php. Create the program code for step 25 show the

result below along with the program code (Question No. 21)





* **Practical Section 5: Using Arrays in PHP**

1. Note here what you observe from adding the code above. (Question No. 22)



$nilaiSiswa contains an array of student scores.

$nilaiLulus is an empty array that will store the scores of students who passed (scores >= 70).

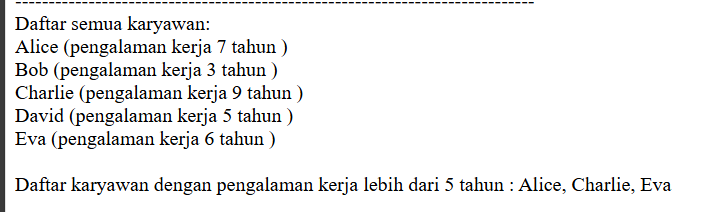
The foreach loop goes through each score in $nilaiSiswa.

If the score is 70 or higher, it is added to the $nilaiLulus array using $nilaiLulus[] = $nilai;.

Display Passing Scores:

Finally, implode(', ', $nilaiLulus) converts the $nilaiLulus array into a comma-separated string, which is displayed using echo.

1. Note here what you observe from the addition of the code above. (Question No. 23)



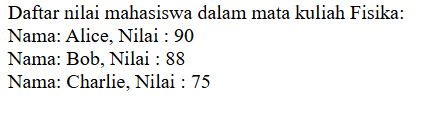
-$daftarKaryawan is a multidimensional array containing employee names and their years of experience

-The first foreach loop creates an array $employeeDetails, where each entry combines the employee's name and their experience in a formatted string (e.g., "Alice (pengalaman kerja 7 tahun)"

-The implode(' ', $employeeDetails) function joins all formatted strings with a space, and the result is displayed as a list of all employees.

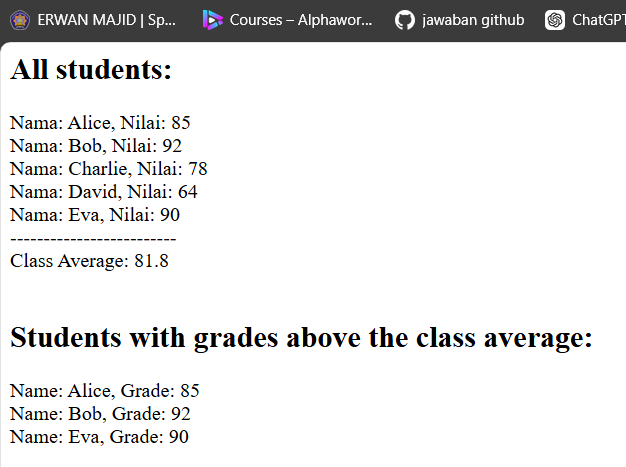
-The second foreach loop checks each employee's experience. If it’s greater than 5 years, the employee's name is added to the $karyawanPengalamanLimaTahun array.

-Finally, it uses implode(', ', $karyawanPengalamanLimaTahun) to create a comma-separated string of employees who have more than 5 years of experience and displays it.

1. Note here what you observe from the addition of the program code above. (Question No. 24)

-array $daftarNilai containing student scores for three subjects: Matematika, Fisika, and Kimia. Each subject is an array of student names paired with their scores. It then sets the variable $mataKuliah to 'Fisika' and prints a header indicating the subject. A foreach loop iterates through the scores for 'Fisika', printing each student's name and score in a formatted manner. Overall, the code displays students' scores specifically for the subject 'Fisika'.

1. Create the code for step 13 with a two-dimensional array and display the result below along with the program code (Question No. 25)





-first displays all students and their grades, then calculates the total grades and the class average by summing the grades and dividing by the number of students. After calculating the average, it prints the average value and then lists the students whose grades exceed this average.

For calculate total grades : foreach ($students as $student) {

$totalGrades += $student[1];

}

-$classAverage = $totalGrades / $numStudents;

-Calculates the total grades and the class average:

echo "Class Average: $classAverage<br><br>";

-Prints the class average:

echo " <h2> Students with grades above the class average:</h2>";

foreach ($students as $student) {

if ($student[1] > $classAverage) {

echo "Name: {$student[0]}, Grade: {$student[1]}<br>";

}

}