LOGICAL OPERATORS AND COMPOUND CONDITIONS

Ripiu :

Dengan menyusun kondisional di dalam kondisional lain, kita dapat membuat keputusan bercabang.

Operator logika || mengambil dua nilai boolean atau ekspresi boolean berbeda sebagai operand-operandnya dan mengembalikan satu nilai boolean. Ini mengembalikan TRUE jika salah satu dari operand kiri atau operand kanan menghasilkan TRUE.

Operator logika && mengembalikan TRUE hanya jika kedua operandnya menghasilkan TRUE. Ini mengembalikan FALSE jika salah satu atau kedua operandnya menghasilkan FALSE.

Operator logika not (!) hanya mengambil operand kanan. Ini membalik nilai boolean dari operandnya.

Operator logika exclusive or (xor) mengembalikan TRUE hanya jika salah satu operand kiri atau operand kanan menghasilkan TRUE, tetapi bukan keduanya atau keduanya tidak sama-sama TRUE.

PHP menyertakan sintaks alternatif untuk operator || dan &&: kita dapat menggunakan or sebagai pengganti ||, dan kita dapat menggunakan and sebagai pengganti &&. Operator ini berfungsi dengan cara yang hampir sama tetapi memiliki prioritas operator yang berbeda.

Kita dapat menyertakan kode dari satu file ke dalam file lain dengan menggunakan include, yang memungkinkan kita menulis program yang lebih modular.

PHP STRINGS AND VARIABLES

Ripiu:

String adalah kumpulan teks yang diperlakukan oleh komputer sebagai satu kesatuan data.

Sebuah string dapat memiliki panjang apa pun dan berisi huruf, angka, simbol, atau spasi yang dikelilingi oleh tanda kutip.

Untuk menyertakan karakter tertentu di dalam string, kita harus menggunakan urutan escape.

Operator adalah karakter yang melakukan tugas dalam kode kita.

Kita dapat menggunakan operator penggabungan (.) untuk menggabungkan dua string menjadi satu.

Variabel adalah konsep dasar pemrograman yang memungkinkan kita dengan mudah menggunakan kembali data dalam kode kita.

Kita mendeklarasikan variabel dengan menggunakan tanda dolar ($) diikuti oleh nama variabel, dan kemudian menggunakan operator penugasan (=) untuk memberikan nilainya.

PHP memiliki penguraian variabel yang memungkinkan kita menyertakan variabel langsung dalam string kita.

Setelah suatu variabel ditugaskan, kita dapat mengubah nilainya. Ini disebut penugasan ulang.

Kita dapat membuat alias untuk variabel, bukan hanya salinan, dengan menggunakan operator penugasan referensi (=&).

Operasi di sebelah kanan operator penugasan akan dievaluasi sebelum penugasan dilakukan.

Operator penugasan penggabungan (.=) adalah notasi singkat untuk menugaskan ulang variabel string ke nilai saat ini yang ditambahkan dengan nilai string lain.

NUMBERS

Ripiu:

PHP memiliki dua jenis data angka: bilangan bulat dan bilangan desimal

Kita dapat menggunakan operator aritmatika untuk melakukan operasi matematika:

Operation:		Example:
Addition	+	`echo 1 + 4.5; // Prints: 5.5`
Subtraction	-	`echo 9 - 1; // Prints: 8`
Multiplication	*	`echo -1.9 * 2.9; // Prints: -5.51`
Division	/	`echo 9 / 1; // Prints: 9`
Modulo	%	`echo 11 % 3; // Prints: 2`
Exponentiation	**	`echo 8**2; // Prints: 64`

Operasi: Operasi memiliki urutan prioritas, yang berarti bahwa beberapa jenis operasi dalam suatu rangkaian akan dievaluasi sebelum yang lain: pertama-tama dievaluasi akan menjadi operasi yang dibungkus dalam tanda kurung (()), selanjutnya eksponen (**), kemudian perkalian (*) dan pembagian (/), dan akhirnya penambahan (+) dan pengurangan (-). Akronim PEMDAS dapat menjadi cara yang membantu untuk mengingat urutannya.

Kita dapat menetapkan nilai angka ke variabel dan kemudian melakukan operasi numerik dengan mereka.

Kita dapat menggunakan operator penugasan matematika sebagai notasi singkat saat menugaskan ulang variabel angka:

Operation:	Long Syntax:	Short Syntax:
Add	`$x = $x + $y`	`$x += $y`
Subtract	`$x = $x - $y`	`$x -= $y`
Multiply	`$x = $x * $y`	`$x *= $y`
Divide	`$x = $x / $y`	`$x /= $y`
Mod	`$x = $x % $y`	`$x %= $y`

ASSOCIATIVE ARRAYS

Ripiu:

Array asosiatif adalah struktur data di mana kunci string atau bilangan bulat terkait dengan nilai.

Kita menggunakan operator => untuk mengaitkan kunci dengan nilainya.
$my_array = ["panda" => "sangat lucu"]

Untuk mencetak kunci dan nilainya dari suatu array, kita dapat menggunakan fungsi print_r().

Kita mengakses nilai yang terkait dengan suatu kunci dengan menggunakan tanda kurung siku ([ ]). Misalnya: $my_array["panda"] akan mengembalikan "sangat lucu".

Kita dapat menetapkan nilai ke kunci menggunakan sintaks indeks yang sama dan operator penugasan (=): $my_array["anjing"] = "lucu sekali";

Sintaks yang sama ini dapat digunakan untuk mengubah elemen yang sudah ada.
$my_array["anjing"] = "keceriaan maksimal";

Kita dapat menghapus sepasang kunci=>nilai sepenuhnya menggunakan fungsi unset() PHP.

Kunci bisa berupa bilangan bulat. Sebenarnya, array terurut adalah array di mana kunci bilangan bulat telah diberikan kepada nilai secara otomatis.

Dalam PHP, array asosiatif dan array terurut adalah penggunaan yang berbeda dari tipe data yang sama.

Operator gabungan (+) mengambil dua operand array dan mengembalikan array baru dengan kunci unik dari array kedua yang ditambahkan ke array pertama.

Saat menulis fungsi dengan parameter array, kita dapat meneruskan array secara nilai atau dengan referensi tergantung pada niat kita.