Rahmat Syaparudin

10112747

Atol - 3

PHP REGULAR EXPRESSION

Laporan & Hasil Capture

1. **PENJELASAN REGULAR EXPRESSION (REGEX)**

Regex merupakan singkatan dari Regular Expression, yaitu sebuah metode untuk mencari suatu pola dalam sebuah string. Regex sendiri sebenarnya juga merupakan string. Hanya saja karakter dalam string tersebut diterjemahkan sebagai suatu pola dan aturan tertentu. Misalkan, string yang diawali oleh tanda caping “^” berarti string tersebut harus diawali oleh karakter yang mengikuti tanda caping tersebut.

Syntax regex pada setiap bahasa pemrograman berbeda-beda. Dalam PHP, awalnya dikenal fungsi ereg(), eregi(), ereg\_replace(), dan eregi\_replace(), untuk memanfaatkan regex ini. Namun fungsi-fungsi tersebut sudah tidak dapat digunakan lagi sejak PHP versi 5.3.0 karena sudah diimplementasikan lagi fungsi yang lebih baik lagi.

Saat ini Anda bisa menggunakan fungsi preg\_match() untuk menggunakan regex. Fungsi lainnya yang tersedia adalah preg\_replace(), preg\_filter(), preg\_match\_all(), dan preg\_split(). Semuanya terletak dalam library pcre.php. PCRE adalah singkatan dari Perl Compatible Regular Expression, yang artinya regex ini dapat digunakan pula pada bahasa pemrograman Perl.

1. **PENGGUNAAN REGEX**

Secara default, pola yang akan dicari harus diawali dan diakhiri oleh karakter garis miring (slash) yang berfungsi sebagai delimeter. Contoh Regex paling sederhana untuk mencari apakah ada pola “fgh” dalam string “abcdefghijk” sebagai berikut.

<?php

$string = 'abcdefghijk';

if(preg\_match("/fgh/", $string)) {

echo 'String berisi pola fgh';

} else {

echo 'Tidak ada pola fgh dalam string';

}

?>

Tanda caping yang diikuti karakter abc berarti bahwa string harus berawalan dengan abc. Sedangkan tanda slash berikutnya menandakan bahwa karakter setelahnya tidak perlu dihiraukan.

1. **URUTAN KARAKTER KHUSUS**

Alfabet dan angka merupakan karakter khusus yang berurutan. Cukup merepotkan bukan bila harus menulis [abcd sampai xyz] ataupun [a-z]? Sama seperti format lainnya, dalam regex juga dikenal escape character khusus untuk alfabet, angka, dan linebreak. Berikut ini adalah daftar urutan karakter khusus yang berlaku pada regex.

|  |
| --- |
| \d – Untuk seluruh karakter berupa angka, sama seperti [0-9] |
| \D – Untuk seluruh karakter yang bukan angka, sama seperti [^0-9] |
| \s – Untuk karakter whitespace, sama seperti penulisan [\t\r\n\f\v] |
| \S – Untuk karakter non whitespace, sama seperti penulisan [^\t\r\n\f\v] |
| \w – Untuk semua karakter alpha-numeric, yaitu alfabet dan angka, sama seperti [a-zA-Z0-9\_] |
| \W – Untuk semua karakter non alpha-numeric, sama seperti [^a-zA-Z0-9\_] |

Contoh penggunaannya sebagai berikut. Misalkan Anda sedang mencari string dengan pola yang diawali sebuah angka, kemudian 3 huruf, dan diakhiri oleh sebuah angka maupun huruf. Maka, penulisannya adalah:

<?php

$string = '2hgd3';

if(preg\_match("/^\d[A-Za-z]{3}\w$/", $string)) {

echo 'Pola sesuai';

}

?>

1. **CONTOH PENGGUNAAN REGEX**
2. Validasi Alamat Email

Fungsi di bawah ini digunakan untuk memeriksa kebenaran / validasi alamat email sebelum dikirim:

function cekmail($email){

return filter\_var( $email, FILTER\_VALIDATE\_EMAIL );

}

Cara penggunaanya:

$alamatemail='tujuan@domain.com';

$cek=cekmail($alamatemail);

if($cek){

kirimemail($tujuan,'Nama Pengirim’,'emailpengirim@domain.com’,'Judul email’,'Ini Isi email’);

}

1. Validasi input berupa Integer

Fungsi di bawah ini digunakan untuk memeriksa kebenaran / validasi input yang dimasukkan berupa integer (bilangan bulat)

<?php

$int = 123;

if(!filter\_var($int, FILTER\_VALIDATE\_INT)){

echo(“Integer is not valid”);

}else {

echo(“Integer is valid”);

}

?>

Kode di atas menggunakan “FILTER\_VALIDATE\_INT” filter untuk menyaring variabel. Karena bilangan bulat tersebut valid, output dari kode di atas akan menjadi: “Integer berlaku”.

Jika kita mencoba dengan variabel yang tidak integer (seperti “123abc”), output akan: “Integer tidak sah”.

1. Fungsi strlen()

Fungsi strlen(), fungsi ini digunakan untuk mengetahui panjang karakter, contoh penggunaanya misalkan inputan nama diharuskan lebih dari 3 karakter :

if ((strlen($\_POST['nama'])) < 3){

//keluarkan pesan error

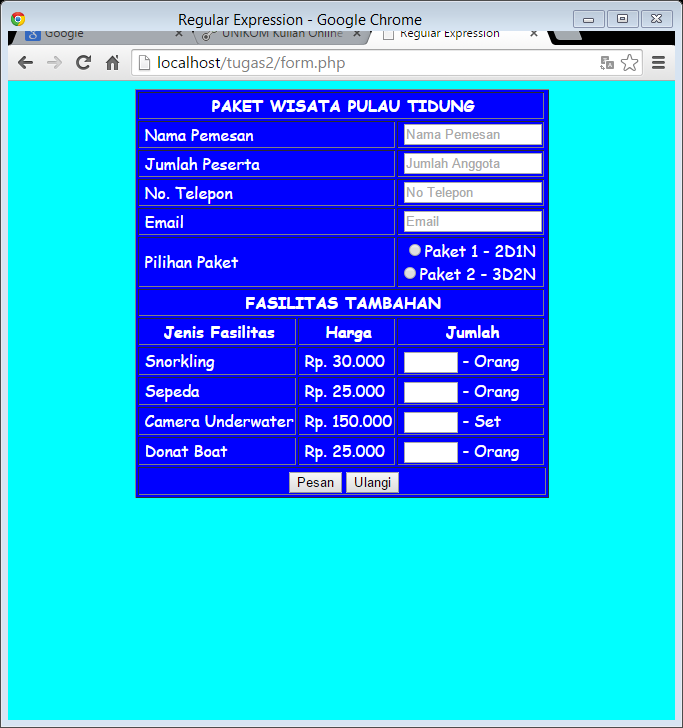
}

else{

//jalankan

}

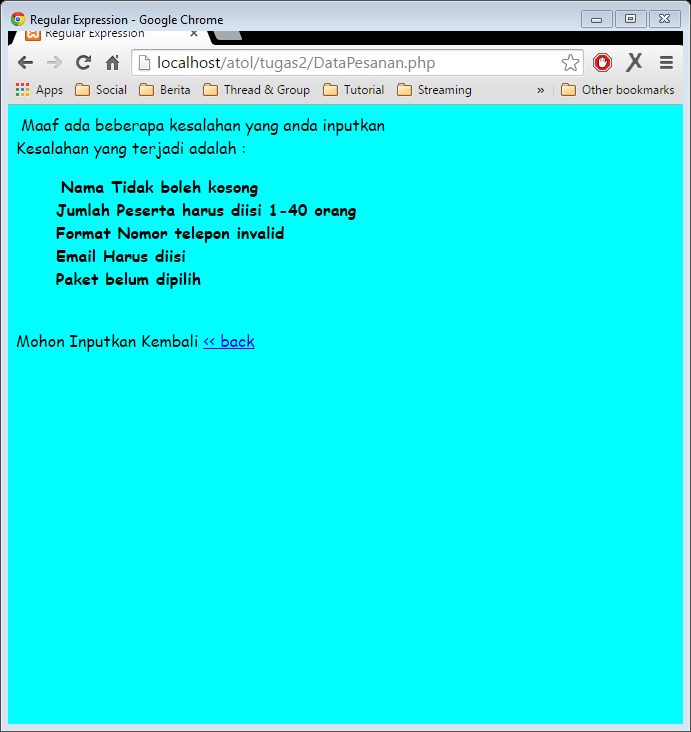
1. **Contoh Implementasi REGEX dalam Form PHP**
2. Tampilan Awal Form Paket Wisata Pulau Tidung



1. Tampilan Output dari Form Pemesanan

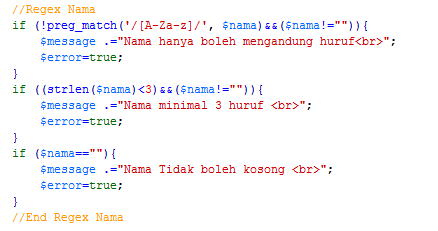


1. Saat kita tidak mengisi form sama sekali, maka akan muncul semua eror ketika menekan tombol Pesan



1. Untuk Nama Pemesan terdapat 3 pesan eror

Pada Textbox Nama Pemesan ini menggunakan fungsi REGEX sebagai berikut:

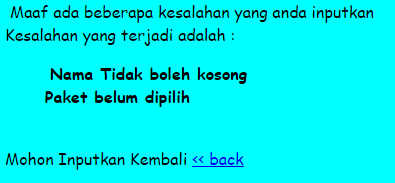


* Untuk pesan eror “nama hanya boleh mengandung huruf” harus berisikan huruf semua tidak boleh mengandung digit
* Untuk pesan eror “Nama minimal 3 huruf” harus berisikan minimal 3 huruf
* Untuk pesan eror “Nama tidak boleh kosong” berarti form nama oemesan tersebut harus diisi dan tidak boleh kosong dengan minimal mengandung 3 huruf

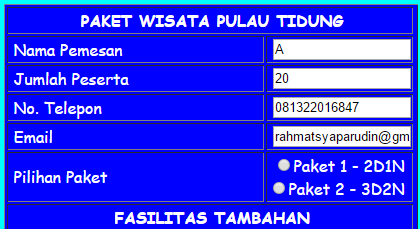
Berikut adalah hasil capture dari pesan kesalahan / eror di atas:

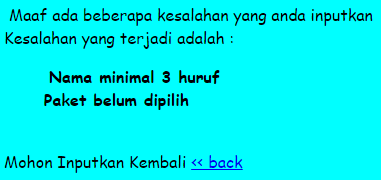
1. Nama Tidak boleh Kosong



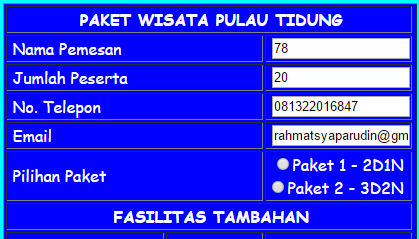


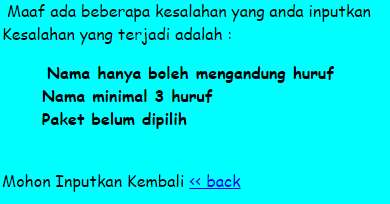
1. Nama minimal 3 huruf





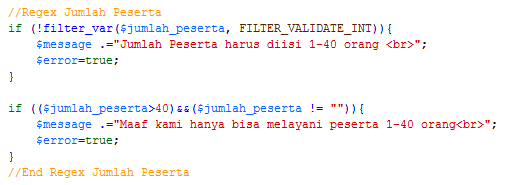
1. Nama hanya mengandung huruf





1. Form Jumlah peserta terdapat 2 validasi eror

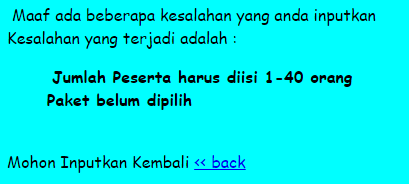
Pada Textbox Jumlah Peserta ini menggunakan fungsi REGEX sebagai berikut:



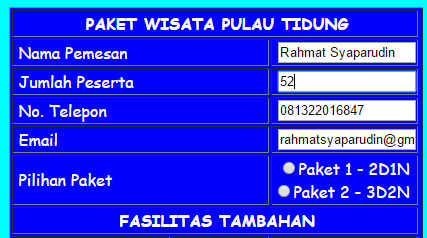
* Fungsi FILTER\_VALIDATE\_INT berfungsi untuk memvalidasi apakah input yang diketik berupa bilangan integer (bilangan bulat). Jika iya, maka akan menampilkan hasilnya, jika salah maka akan emnampilkan pesan eror.
* Jika menginputkan huruf, simbol, atau gabungan keduanya maka akan menampilkan pesan “jumlah peserta harus diisi antara 1 samapi 40 orang”
* Jika input yang diketik melebihi dari 40 orang, maka akan muncul pesan eror “Maaf kami hanya bisa melayani peserta 1-40 orang”

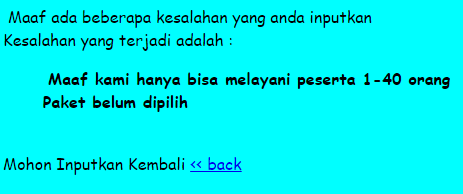
1. Jumlah peserta harus diisi





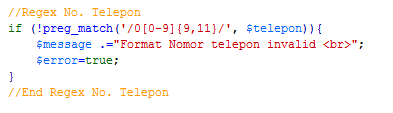
1. Jumlah peserta harus diisi antara 1 sampai 40 orang





1. Form Nomor Telepon terdapat 1 validasi eror

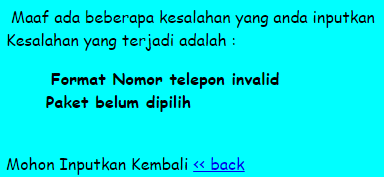
Pada Textbox Nomor Telepon ini menggunakan fungsi REGEX sebagai berikut:



* Validasi nomor telepon menggunakan fungsi pres\_match() yang berarti mencocokan inputan yang dimasukkan dengan pola yang sudah ditentukan.
* Inputan harus diawali dengan angka ‘0’ dan diikuti angka setelahnya sebanyak 9 sampai 11 angka
* Inputan nomor telepon harus berjumlah 10 sampai 12 digit

1. Format nomor telepon harus diawali dengan 0 dan berukuran 10 -12 digit

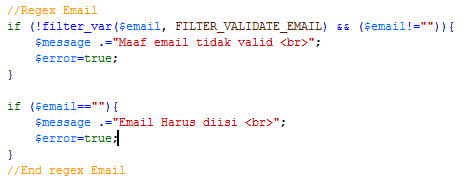




1. Form Email

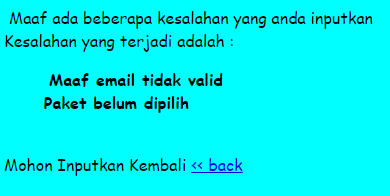
Validasi email menggunakan FILTER\_VARIDATE\_EMAIL. Input Email boleh terdiri dari karakter dan angka, akan tetapi dengan ketentuan sebagai berikut:

* Boleh diawali dengan karakter angka atau huruf
* Jumlah huruf sebelum tanda ‘@’ minimal 1 karakter
* Angka atau huruf yang dimasukkan boleh berulang
* Karakter ‘@’ harus ada
* Tanda titik ‘.’ Boleh ada tetapi di antara karakter huruf atau angka
* Untuk akhiran karakter setelah ‘@’ harus dengan titik ‘.’ (dot) karakter huruf minimal 2 dan maksimal 6 digit huruf



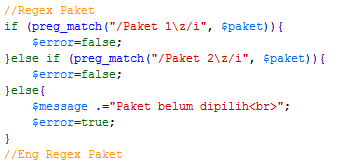
1. Input Email yang tidak sesuai aturan





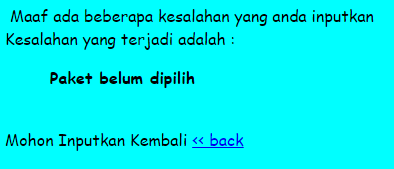
1. Form Pilih salah satu Paket

Validasi Pilih Paket menggunakan pres\_match( ). Karena pilih paket ini menggunakan radio Buton, maka harus dipilih alah satunya. Jika tidak maka akan menampilkan eror “Paket belum dipilih”



1. Paket yang belum dipilih





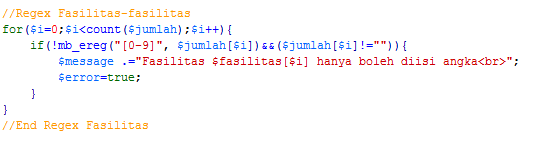
1. Ketika semua data pemesanan paket lengkap maka tidak akan ada eror lagi, jika tidak menambahkan fasilitas tambahan apapun.



1. Fasilitas Tambahan

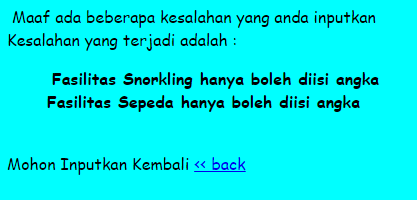
Pada for fasilitas tambahan ini, mempunyai ketentuan sebagai berikut

* Kita hanya boleh menginputkan angka saja
* Boleh dikosongkan atau dalam artisan tidak diisi satu sama sekali
* Boleh diisi sebagian saja, misalnya hanya snorkeling dan sepeda saja.
* Jika terdapat eror, maka akan menampilkan pesan eror “Fasilitas (nama\_fasilitas) hanya boleh diisi angka.



1. Fasilitas dengan inputan huruf





1. Jika input data pemesanan dan fasilitas tambahan benar, maka tidak akan muncul eror

