

PERTEMUAN 17

ERROR HANDLING

A. CAPAIAN PEMBELAJARAN

Capaian pembelajaran dari pertemuan ini diharapkan mahasiswa mampu menjelaskan terjadinya pesan kesalahan dan solusi memperbaiki kesalahan.

B. URAIAN MATERI

1. Pesan Kesalahan dan Solusi Perbaikan

Beberapa pesan kesalahan yang terjadi :

Pesan Error:

view sourceprint?

1.Warning: mysql_fetch_array(): supplied argument is not

2.a valid MySQL result resource in

atau

view sourceprint?

1.Warning: mysql_num_rows(): supplied argument is not

2.a valid MySQL result resource in ...

Penyebab:

Error di atas disebabkan ada query SQL yang salah. Bisa jadi karena sintaks perintah query SQL nya salah, misalnya kurang tanda petik untuk menyatakan string. Selain itu bisa juga terjadi karena nama field dan tabel yang salah penulisannya.

Solusi:

Coba cek kembali statement query SQL. Pastikan tidak ada lagi kesalahan dalam penulisannya. Untuk proses debugging query SQL (mengetahui letak kesalahannya), bisa menambahkan statement die(mysql_error()) pada perintah mysql_query() nya. Contoh:

[view sourceprint?](#)

- 1.\$query = ".....";
- 2.mysql_query(\$query) or die(mysql_error());
- 3..

Pesan Error:

[view sourceprint?](#)

- 1.Warning: session_start() [function.session-start]:
- 2.Cannot send session cookie - headers already sent by
- 3.(output started at ...) in ... on line ...

atau

[view sourceprint?](#)

- 1.Warning: session_start() [function.session-start]:
- 2.Cannot send session cache limiter - headers already sent
- 3.(output started at ...) in ... on line ...

Penyebab:

Error di atas biasanya terjadi pada penggunaan session. Sewaktu memberikan perintah session_start(), terdapat baris perintah di atasnya yang menampilkan string atau kode html ke browser. Sebagai contoh, perhatikan script berikut ini:

[view sourceprint?](#)

- 1.<?php
- 2.echo "Hallo...";
- 3.session_start();
- 4.?>

atau

[view sourceprint?](#)

- 1.<html>

```
2.<head>
3.  <title>....</title>
4.</head>
5.<?php
6.session_start();
7.?>
```

Perhatikan script di atas, maka terdapat string yang di-echo-kan atau terdapat kode html yang diletakkan sebelum session_start(). Nah... inilah yang menyebabkan error. Selain contoh di atas, error juga bisa muncul dari script yang berbentuk seperti ini

[view sourceprint?](#)

```
1.<?php
2.include "header.php";
3.session_start();
4.?>
```

Meskipun sebelum session_start() tidak ada echo atau kode html, namun error akan tetap muncul apabila isi dari script header.php terdapat echo atau kode html juga, misalnya

header.php

[view sourceprint?](#)

```
1.<?php
2.echo "hallo";
3.?>
```

Solusi:

Pastikan sebelum perintah session_start() tidak terdapat perintah echo atau kode html apapun.

Pesan Error:

[view sourceprint?](#)

1.Warning: Cannot modify header information – headers already sent by

2.(output started at ...) in ...

Penyebab:

Peringatan di atas biasanya muncul ketika melakukan redirecting menggunakan perintah

[view sourceprint?](#)

```
1.header("Location: ...");
```

Seharusnya, sebelum perintah header("Location: ...") tidak boleh ada string apapun yang ditampilkan di browser, meskipun itu kode HTML (sama seperti penyebab error di session_start() sebelumnya). Sebagai contoh perhatikan script berikut ini:

[view sourceprint?](#)

```
1.<?php
```

```
2.echo "<p>Hallo Apa kabar?</p>";
```

```
3.header("Location: test.php");
```

```
4.?>
```

Pada script di atas, sebelum perintah header() terdapat string yang ditampilkan ke browser. Inilah yang mengakibatkan muncul peringatan tersebut. Bagaimana dengan script berikut ini?

[view sourceprint?](#)

```
1.<?php
```

```
2.$a1 = 10;
```

```
3.if ($a1 < 0) echo "Nilai A negatif";
```

```
4.else header("Location: test.php");
```

```
5.?>
```

Apakah muncul Warning atau tidak? Jika di coba, maka tidak muncul Warning nya meskipun sebelum header() terdapat string yang ditampilkan ke browser. Mengapa demikian? meskipun di situ terdapat perintah untuk menampilkan string

ke browser, namun perintah tersebut tidak pernah dijalankan karena nilai \$a1 nya positif.

Solusi:

Pastikan sebelum perintah header("Location: ...") tidak ada kode HTML atau perintah echo apapun. Jika meng-include-kan suatu script di atas header("Location: ...") pastikan dalam script yang di-include-kan tersebut tidak ada kode HTML atau echo.

Pesan Error:

[view sourceprint?](#)

- 1.Warning: mysql_connect() [function.mysql-connect]:
- 2.Access denied for user 'root'@'localhost' (using password: YES) in ...

Penyebab:

Error terjadi disebabkan koneksi yang gagal ke MySQL. Koneksi gagal disebabkan nama user atau password yang salah.

Solusi:

Yakinkan username dan password koneksi ke MySQL sudah benar ketika menggunakan perintah koneksi `mysql_connect("namahost", "namauser", "passworduser");`

Pesan Error:

[view sourceprint?](#)

- 1.Parse error: syntax error, unexpected T_STRING,
- 2.expecting T_OLD_FUNCTION or T_FUNCTION or T_VAR or '}' in ...

Penyebab:

Biasanya ini terjadi karena penggunaan versi PHP yang rendah (versi di bawah PHP 5) untuk menjalankan script PHP yang menggunakan OOP.

Solusi:

Melakukan upgrade versi PHP, atau gunakan versi PHP yang lebih tinggi

Pesan Error:

[view sourceprint?](#)

1. Parse error: syntax error, unexpected T_ELSE in ...

Penyebab:

Error terjadi karena penulisan sintaks IF ... ELSE yang tidak sesuai aturan. Perhatikan contoh penggunaan IF ELSE yang salah di bawah ini

[view sourceprint?](#)

```
1.<?php
2.$a1 = 5;
3.if ($a1 > 0) $status1 = "A lebih besar dari 0"
4.else ($a1 < 0) $status1 = "A lebih kecil dari 0"
5.else $status1 = "A sama dengan 0"
6.?>
```

Contoh di atas terdapat kesalahan yaitu setiap akhir statement tidak diberikan tanda titik koma (;), dan pada bagian else (\$a1 < 0) kurang IF, seharusnya

[view sourceprint?](#)

```
1.<?php
2.$a1 = 5;
3.if ($a1 > 0) $status1 = "A lebih besar dari 0";
4.else if ($a1 < 0) $status1 = "A lebih kecil dari 0";
5.else $status1 = "A sama dengan 0";
6.?>
```

Solusi:

Pastikan struktur statement yang digunakan berbentuk seperti ini

[view sourceprint?](#)

```
1.<?php
2.if (kondisi1) perintah1;
3.else if (kondisi2) perintah2;
4.else if (kondisi3) perintah3;
5..
```

6..

7.else perintahX;

8.?>

atau

[view sourceprint?](#)

01.<?php

02.if (kondisi1)

03.{

04. perintah;

05. perintah;

06. .

07. .

08.}

09.else if (kondisi2)

10.{

11. perintah;

12. perintah;

13. .

14. .

15.}

16..

17..

18.else

19.{

20. perintah;

21. perintah;

22. .

23. .

24.}

25.?>

Pesan Error:

[view sourceprint?](#)

1.Fatal error: Call to undefined function: ...

Penyebab:

Error tersebut disebabkan karena nama function yang digunakan tidak ada atau tidak dikenal oleh PHP. Jika function tersebut adalah built in function, misalnya `str_split()` bisa jadi hal ini disebabkan pada versi PHP yang digunakan belum ada function tersebut. Jika function tersebut adalah function buatan sendiri, bisa jadi lupa menyisipkan ke dalam script atau meng-include-kannya.

Solusi:

Jika function yang tidak dikenal adalah built in function, maka lakukan upgrade versi PHP yang lebih tinggi. Sedangkan jika function yang tidak dikenal adalah buatan sendiri, pastikan sudah meng-includekan di dalam scriptnya sehingga bisa dikenal. Contohnya:

function.php

[view sourceprint?](#)

1.<?php

2.function jumlah(\$a, \$b)

3.{

4. return \$a + \$b;

5.}

6.?>

script.php

[view sourceprint?](#)


```
1.<?php
2.include "function.php";
3.echo jumlah(9, 10); // akan menghasilkan 19
4.?>
atau
```

script.php[view sourceprint?](#)

```
1.<?php
2.function jumlah($a, $b)
3.{
4.  return $a + $b;
5.}
6.
7.echo jumlah(9, 10); // akan menghasilkan 19
8.?>
```

Pesan Error:[view sourceprint?](#)

```
1.Parse error: parse error, unexpected $end
```

Penyebab:

Error disebabkan karena kurangnya tanda penutup kurung kurawal. Contohnya:

[view sourceprint?](#)

```
1.<?php
2.if ($a1 > 1)
3.{
4.  echo "Nilai A lebih dari 1";
```

5. ?>

atau

[view sourceprint?](#)

1. <?php

2. while (\$a1 > 1)

3. {

4. .

5. .

6. .

7. ?>

Selain itu, bisa pula error tersebut disebabkan karena efek perbedaan penggunaan tanda pembuka PHP <?php dan <?. Contohnya :

[view sourceprint?](#)

01. <?php

02. if (kondisi)

03. {

04. statement1;

05. statement2;

06. ?>

07.

08. <?

09. statement3;

10. statement4;

11. }

12. ?>

Perhatikan contoh script di atas, bahwa pada blok statement IF dipecah menjadi 2 blok, namun perhatikan bahwa untuk blok kedua, tanda pembukanya menggunakan <? Saja, terkadang suatu server hanya membolehkan tanda

pembuka script PHP menggunakan `<?php`, sehingga untuk blok kedua tidak dianggap sebagai script, dan.. akibatnya statement IF nya dianggap tidak ada tanda `}` penutup.

Solusi:

Pastikan tanda kurung kurawal penutup tidak lupa menuliskannya. Jika ternyata kasus yang di hadapi sama seperti contoh kasus terakhir (perbedaan tanda pembuka script PHP), konsistenkan penulisan tanda pembukanya menggunakan `<?php` supaya aman.

Pesan Error:

[view sourceprint?](#)

1.Notice: Undefined index: ...

atau

[view sourceprint?](#)

1.Notice: Undefined variable: ...

Penyebab:

Sebenarnya itu bukan error, namun sekedar peringatan saja bahwa ada variabel atau index array yang belum terdefinisi sebelumnya. Di beberapa versi PHP peringatan tersebut muncul, namun di versi yang lain tidak muncul. Berikut ini contohnya:

[view sourceprint?](#)

1.<?php

2.if (\$a > 0)

3.{

4.

5.}

6.?>

Perhatikan script di atas, variabel \$a langsung diberikan statement pengecekan IF, padahal nilai \$a ini tidak ada sebelumnya, di beberapa versi PHP, akan muncul peringatan

[view sourceprint?](#)

1. Notice: Undefined variable: a

Namun, bila nilai \$a ini sudah didefinisikan sebelumnya misalnya

[view sourceprint?](#)

```
1.<?php
2.$a = 10;
3.if ($a > 0)
4.{
5. ....
6.}
7.?>
```

maka peringatan di atas tidak akan muncul lagi. Peringatan yang hampir sama juga bisa muncul untuk contoh berikut ini

[view sourceprint?](#)

```
1.<?php
2.if ($_GET['test1'] == 0)
3.{
4. ....
5.}
6.?>
```

yang dikarenakan nilai \$_GET['test1'] belum terdefinisi sebelumnya atau belum ada nilainya (belum ada request GET untuk parameter 'test1').

Solusi:

Sebenarnya peringatan di atas bisa saja diabaikan karena tidak berpengaruh pada kinerja script atau proses yang terjadi di dalamnya. Namun, jika ingin menghilangkan peringatan tersebut caranya adalah dengan memodifikasi parameter

[view sourceprint?](#)

```
1.error_reporting = ...
```

pada file konfigurasi **php.ini**

Tambahkan nilai "& ~E_NOTICE" pada nilai parameternya, contoh:

[view sourceprint?](#)

```
1.error_reporting = E_ALL & ~E_NOTICE
```

Maksud dari parameter di atas adalah, PHP akan menampilkan Pesan error kecuali NOTICE. Jika sudah, silakan merestart Apache nya.

Pesan Error:

[view sourceprint?](#)

```
1.Fatal error: Maximum execution time of ... seconds exceeded in ...
```

Penyebab:

Error di atas disebabkan karena lama eksekusi sebuah script PHP telah mencapai batas maksimumnya. Jika menggunakan AppServ atau XAMPP, default *maximum execution time* nya diset 30 detik. Jika lama eksekusi script yang dibuat melebihi 30 detik, maka akan muncul error di atas.

Solusi:

Untuk meningkatkan atau menambah max execution time pada file php.ini, dengan cara carilah parameter

[view sourceprint?](#)

```
1.max_execution_time = ...
```

Tambahlah nilainya, misalnya menjadi 9999 detik, sehingga nilai parameternya menjadi

[view sourceprint?](#)

```
1.max_execution_time = 9999
```

Selanjutnya, simpan kembali php.ini nya, lalu restart Apachenya. Kalau perlu, lakukan analisis algoritma pada script apabila dirasa tidak efisien sehingga mengakibatkan running time yang lama.

Pesan Error:

Bukan Error , tapi kegagalan dalam proses upload file ketika membuat script upload file.

Penyebab:

Kegagalan dalam proses upload file itu disebabkan oleh ukuran file yang terlalu besar melebihi batas maksimum yang diperbolehkan oleh PHP sesuai di file php.ini.

Solusi:

Ubah batas maksimum file size untuk upload ini dengan mengubah parameter

[view sourceprint?](#)

```
1.upload_max_filesize = ...
```

dan

[view sourceprint?](#)

```
1.post_max_size = ...
```

pada file php.ini

Contoh untuk mengubah batas maksimum upload file hingga 90 Mb.

[view sourceprint?](#)

```
1.upload_max_filesize = 90M
```

dan

[view sourceprint?](#)

```
1.post_max_size = 80M
```

Setelah melakukan perubahan, lakukan restart Apache nya, jangan lupa pula untuk menambah komponen input

[view sourceprint?](#)

```
1.<input type="hidden" name="MAX_FILE_SIZE" value="80000000" >
```

pada form upload nya.

Solusi di atas untuk dapat menyelesaikan permasalahan upload jika file diupload ke folder atau direktori tertentu yang ada di server. Namun, jika file diupload ke database server, maka solusi di atas tidak menyelesaikan masalah karena adanya batasan limit file size pada tipe data yang memungkinkan untuk menyimpan data file, misalnya tipe data blob, medium blob, atau long blob.

C. LATIHAN SOAL

1. Tuliskan 3 pesan kesalahan apa yang sering anda temui ketika membuat program, solusi apa yang anda lakukan berdasarkan pesan kesalahan yang tampil, jelaskan dalam bentuk script!
2. Tuliskan 3 buah pesan kesalahan yang terkait pembuatan database, solusi apa yang anda lakukan untuk memperbaiki pesan kesalahan tersebut, jelaskan dalam bentuk script!
3. Tuliskan 3 pesan kesalahan yang tampil ketika melakukan penyimpanan database atau proses perhitungan dalam database dan bagaimana solusi memperbaikinya, jelaskan dalam bentuk script!

D. DAFTAR PUSTAKA

Abdul Kadir, 2002, Dasar Pemrograman Web Dinamis Menggunakan PHP, Andi Yogyakarta.

<http://www.w3schools.com>