

# بسم الله الرحمن الرحيم

## آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

### آموزش Cache کردن Recordset

مدرس : مهندس افشین رفوآ

## آموزش Cache کردن Recordset

### آموزش برنامه نویسی : آموزش برنامه نویسی PHP

یک ویژگی ظریف در **ADODB**، **Cache** کردن **Recordset** است. این کار باعث سریع شدن نتایج پرس و جو می شود.

از چندین متد می توان برای **Cache** کردن **Recordset** استفاده کرد.

CacheExecute()

CachePageExecute()

CacheSelectLimit()

#### CacheExecute()

مانند **Execute** است، فقط **recordset** برای **cache2secs** ثانیه، در دایرکتوری **ADODB\_CACHE\_DIR**، **cache** می شود. اگر **CacheExecute()** دوباره با همان پارامترها، همان پایگاه داده، همان نام کاربری و همان پسورد، فراخوانی شود و **Recordset**، کش (**cache**) شده منقضی نشده باشد، همان **recordset** کش شده برگردانده می شود.

```
include('adodb.inc.php');
include('tohtml.inc.php');
$ADODB_CACHE_DIR = '/usr/local/ADODbcache';
$conn = &ADONewConnection('mysql');
$conn->
>PConnect('localhost','userid','password','database');
$rs = $conn->CacheExecute(15, 'select * from table');
# cache 15 secs
```

#### CachePageExecute()

فرمول

```
CachePageExecute($secs2cache, $sql, $nrows, $page,  
$inputarr=false)
```

برای صفحه بندی ( **pagination** یا صفحه صفحه کردن) **recordset** استفاده می شود، صفحه ها از 1 شروع می شود نه صفر.

```
include_once('adodb.inc.php');  
include_once('adodb-pager.inc.php');  
session_start();  
$db = NewADOConnection('mysql');  
$db->Connect('localhost','root','','xphplens');  
$sql = "select * from adoxyz ";  
$pager = new ADODB_Pager($db,$sql);  
$pager->Render($rows_per_page=5);
```

## CacheSelectLimit()

فرمول

```
CacheSelectLimit([$secs2cache,] $sql, $numrows=-1,$offset=-  
1,$inputarr=false)
```

مانند **SelectLimit** است، فقط **recordset** بازگشتی برای **\$cache2secs** ثانیه، در دایرکتوری **\$ADODB\_CACHE\_DIR**، **cache** می شود.

```
$conn->Connect(...);  
$conn->cacheSecs = 3600*24; // cache 24 hours  
$rs = $conn->CacheSelectLimit('select * from table',10);
```

بازدهی این 3 متد انقدر متفاوت است که باید درباره زمان مناسب برای کش کردن صحبت کنیم. هنگامی که سرور پایگاه داده شما خیلی از وب سرور شما کند تر است، و یا پایگاه داده ترافیک شدید دارد، کش کردن مناسب است چون بار روی پایگاه داده را کم می کند. اگر پایگاه داده شما بار چندانی ندارد و خیلی از وب سرورتان سریع تر است، کش کردن کارایی و بازدهی را کاهش می دهد.