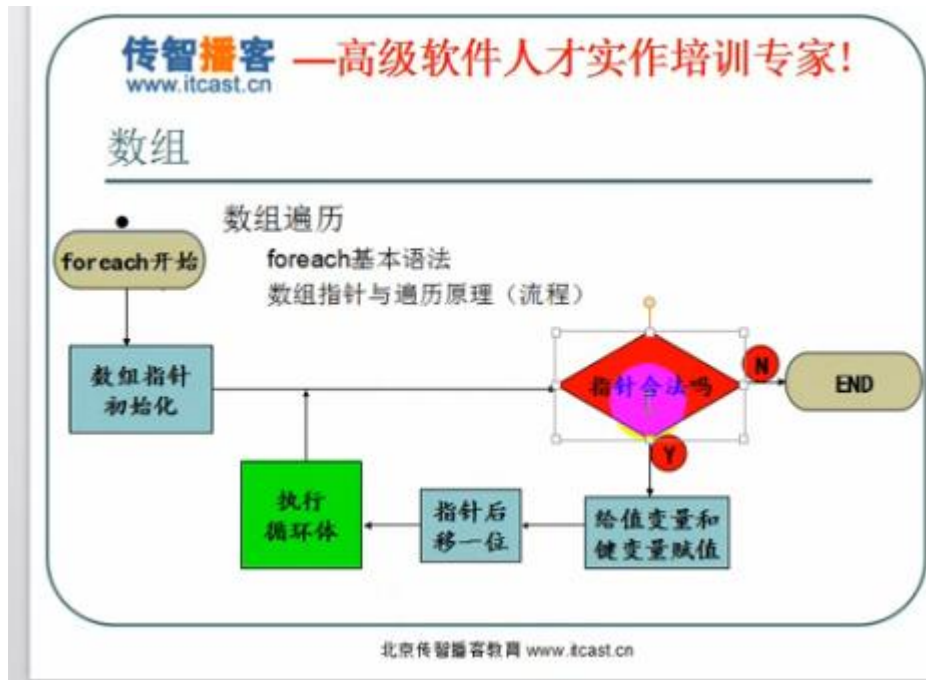


数组遍历原理图



使用 for 和 next 遍历数组

注意:对 php 数组,往往不能单纯使用 for 循环进行遍历.

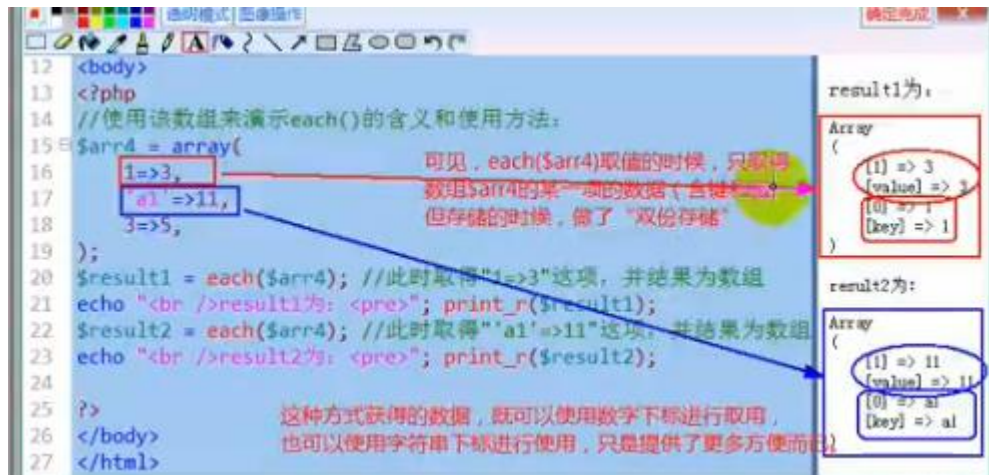
或者说:php 中,使用 for 循环只能循环“下标为连续的纯整数数组”;

Each()函数的使用

Each()函数的作用:先取得一个数组“当前单元”的下标和值(并放入一个数组),然后将指针移动下到下一个单元.

使用形式:

`$a = each($数组名);` //此时\$a 就是一个数组了



List()函数的使用

List()函数用于一次性取得一个数组中从 0 开始的数字下标的多个单元的值

形式:

`List($变量 1,$变量 2,$变量 3) = $数组;`

作用:

上述一行代码相当于如下代码:

\$变量 1=\$数组[0];

\$变量 2=\$数组[1];

\$变量 3=\$数组[2];

注意:这里变量的个数,要跟该数组的数字下标的单元对应,如果某个变量对应的该数字下标不存在,就会报错;

While+each()+list()遍历数组



```
14 //使用while和each和list配合来遍历该数组
15 $arr4 = array(1=>3, 'a1'=>11, 3=>5, "mn"=>18, 88=>2 );
16 while( list($key, $value) = each($arr4) ){//几乎是模式化的写法
17     //这里就可以处理$key和$value;
18     echo "<br />$key = $value";
19 }
20 //对比foreach的原始遍历语法:
21 echo "<br />";
22 $arr4 = array(1=>3, 'a1'=>11, 3=>5, 'value'=>18, 3, 88=>2 );
23 foreach($arr4 as $key => $value){
24     echo "<br />$key = $value";
25 }
```

这里, each(\$arr4)的结果是一个数组, 类似这样:

```
array(
    1=>3,
    0=>1,
    'key'=>1
)
```

1 = 3  
a1 = 11  
3 = 5  
mn = 18  
88 = 2

1 = 3  
a1 = 11  
3 = 5  
mn = 18  
88 = 2