

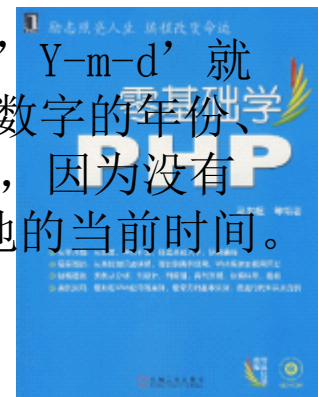
第7章 用PHP处理日期和时间

- 日期和时间的处理是程序开发人员经常要面对的问题。很多编程语言都提供对时间、日期处理机制，但有的语言处理机制比较繁琐，甚至难以理解。幸运的是，PHP程序员不会过多担心对时间、日期的处理，因为PHP提供非常易用的函数来支持程序中对时间和日期的处理。
- 这一章介绍获取时间和检查时间的有关函数及其用法，主要包括如何格式化一个日期和时间、UNIX时间戳的概念、验证日期是否有效等内容。



7.1 获取格式化的日期和时间

- 在PHP中使用函数date()可以得到一个格式化的日期和时间。该函数的语法如下所示。
- `string date(string $format [, int timestamp]);`
- 函数date()有两个参数，第1个参数\$format是一个格式化的字符串，第2个参数是可选的，它表示UNIX的时间戳，关于UNIX时间戳将在下一节介绍。该函数将时间戳\$timestamp按字符串\$format的格式返回，如果没有提供第2个参数\$timestamp，函数将默认使用本地当前时间，即将本地当前时间格式化输出。例如，下面是一行调用函数date()的代码。
- `echo date('Y-m-d');`
- 这段代码输出的结果类似于：2007-12-01。代码中的'Y-m-d'就是一个格式化的字符串，它告诉函数date()，按四位数字的年份、两位数字的月份和两位数字的天数格式化日期和时间，因为没有给函数date()传入第2个参数，该函数将会格式化本地的当前时间。



7.2 处理UNIX时间戳

- 上一小节介绍函数date()时，讲述过其第2个参数是一个UNIX时间戳。大多数UNIX系统用一个32-bit的整数存储当前时间和日期，这个整数是从1970年1月1日零时（按GMT）开始的秒数，这个秒数称为UNIX时间戳。
- 如果希望将一个日期和时间转换成一个UNIX时间戳，可以使用PHP提供的函数mktime()来完成。该函数的语法如下所示。
- `int mktime(int $hour, int $minute, int $second, int $month, int $day, int $year);`
- 函数mktime根据所给参数，返回UNIX时间戳。如果参数非法，函数返回FALSE。函数mktime()的第1个参数是小时数，第2个参数是分钟数，第3个参数是秒数（范围是0-59），第4个参数是月份，第5个参数是天数，第6个参数是年份，年份可以是两位或四位数字，如果是两位数字，0-69对应于2000-2069，70-99对应于1970-1999。该函数的参数从右向左可以省略，省略的参数将以本地日期和时间代替。例如下面的代码，返回当前日期和时间的UNIX时间戳。
- `$cur_timestamp = mktime();`



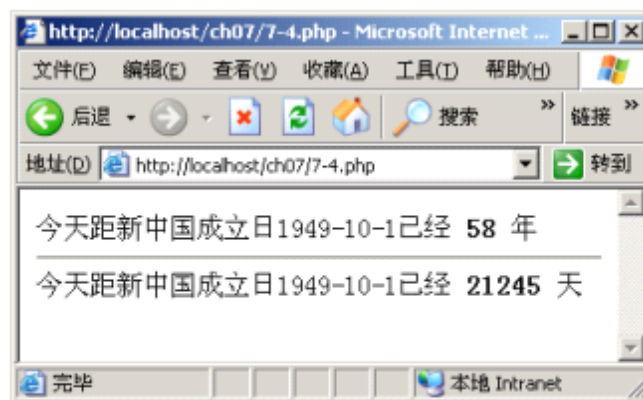
7.3 获取日期和时间的相关信息

- 函数getdate()是一个比较实用的函数，可以通过这个函数取得日期和时间的有关信息。该函数的语法如下所示。
- `array getdate([int $timestamp]);`
- 该函数的参数是一个UNIX时间戳，它是可选的，如果没有给定该参数，PHP默认使用当前本地时间。函数处理成功后，会返回一个关联数组，数组的每个单元存储着日期和时间的相关信息。



7.4 PHP中的日期计算

- 有时会在PHP程序中计算两个日期之间的时间跨度，这可以通过计算两个日期之间的UNIX时间戳之差实现。



PHP 中日期的计算



7.5 检查日期的有效性

- PHP提供函数checkdate()来验证日期是否有效，该函数的语法如下所示。
- `bool checkdate(int $month, int $day, int $year);`
- 函数检查由参数构成的日期，如果给出的日期有效，函数返回TRUE，否则返回FALSE。第1个参数表示月份，它的取值范围是1到12；第二参数是日期，它的取值范围在\$month所具有的有效天数之内，并且闰年已经考虑进去；第3个参数是年份，它的取值范围是1到32,767。



验证日期是否有效



7.6 小结

- 这一章介绍了PHP有关处理时间的函数，包括对日期时间的格式化、验证日期是否有效、日期之间的计算等。具体包括如下函数和内容。
- 处理格式化日期和时间的函数date()。
- 获取UNIX时间戳的函数mktime()和time()。
- 取日期和时间的相关信息的函数getdate()。
- 在PHP程序中实现日期之间跨度的计算。
- 验证日期是否有效的函数checkdate()。

