//打开文件，rb是二进制模式读文件  
$file\_content = *fopen*($\_FILES['file']['tmp\_name'], 'rb');

//读取两字节  
$file\_bin = *fread*($file\_content, 2);  
*fclose*($file\_content);  
//解析文件类型  
$binarr = *unpack*('C2chars', $file\_bin);  
$code\_arr = array(  
 '.jpg' => 255216,  
 '.png' => 13780,  
 '.gif' => 7173,  
 '.bmp' => 6677  
);  
if (!*in\_array*($binarr ['chars1'] . $binarr ['chars2'], $code\_arr)) {  
 SmartyAdapter::display(array("error\_info" => '错误的文件格式'));  
 exit;  
}

**fopen()** 将 filename指定的名字资源绑定到一个流上。

feof — 测试文件指针是否到了文件结束的位置

fgets — 从文件指针中读取一行

string **fgets** ( resource $handle [, int file — 把整个文件读入一个数组中$length ] )

从 handle指向的文件中读取一行并返回长度最多为 length - 1 字节的字符串。碰到换行符（包括在返回值中）、EOF 或者已经读取了 length - 1 字节后停止（看先碰到那一种情况）。如果没有指定 length，则默认为 1K，或者说 1024 字节。

**Note**:

从 PHP 4.3 开始，忽略掉 length 则行的长度被假定为 1024，将继续从流中读取数据直到行结束。如果文件中的大多数行都大于 8KB，则在脚本中指定最大行的长度在利用资源上更为有效

从指针 handle 指向的文件中读取了 length - 1 字节后返回字符串。 如果文件指针中没有更多的数据了则返回 **FALSE**。

file — 把整个文件读入一个数组中

file\_exists — 检查文件或目录是否存在

//获取文件内容  
$handle = *fopen*($file['tmp\_name'], 'r');

if ($handle) {  
 while (!*feof*($handle)) {  
 $buffer = *fgets*($handle, 4097);//读取一行不超过4M数据

fclose — 关闭一个已打开的文件指针

/\*\*  
 \* 下载模板文件  
 \*/  
public function downLoadModelFileAction()  
{  
 $filename = *ROOT\_PATH* . '/mis/static/txt/platformCardOrder.csv';  
 $fileinfo = *pathinfo*($filename);  
 *header*('Content-type: application/x-' . $fileinfo['extension']);  
 *header*('Content-Disposition: attachment; filename=' . $fileinfo['basename']);  
 *header*('Content-Length: ' . *filesize*($filename));  
 *readfile*($filename);  
 exit();  
}

**optimize**

award.php、186行上传文件