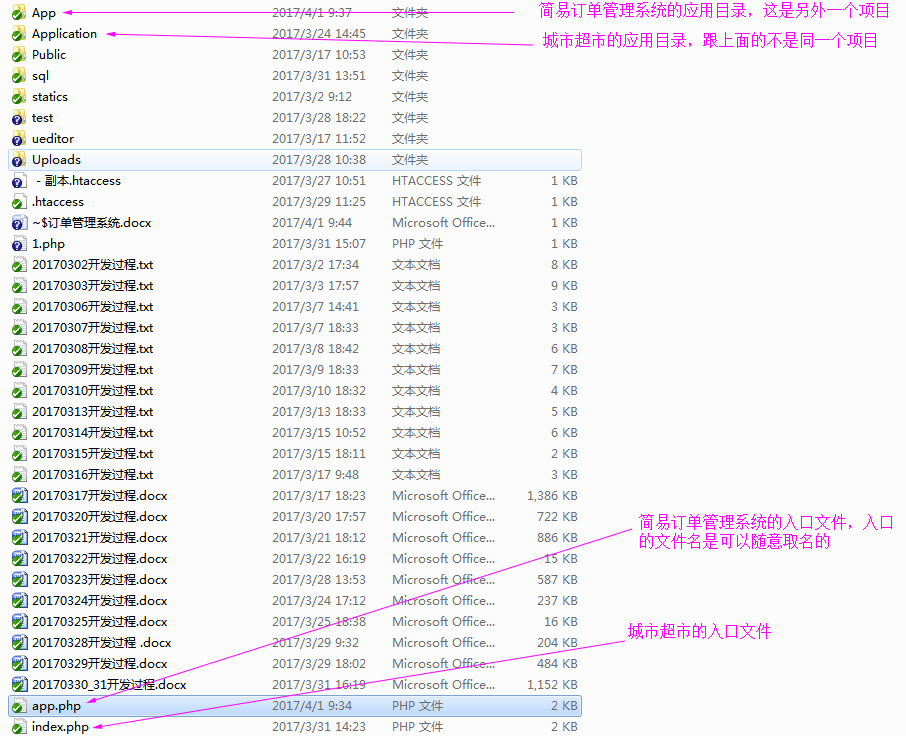
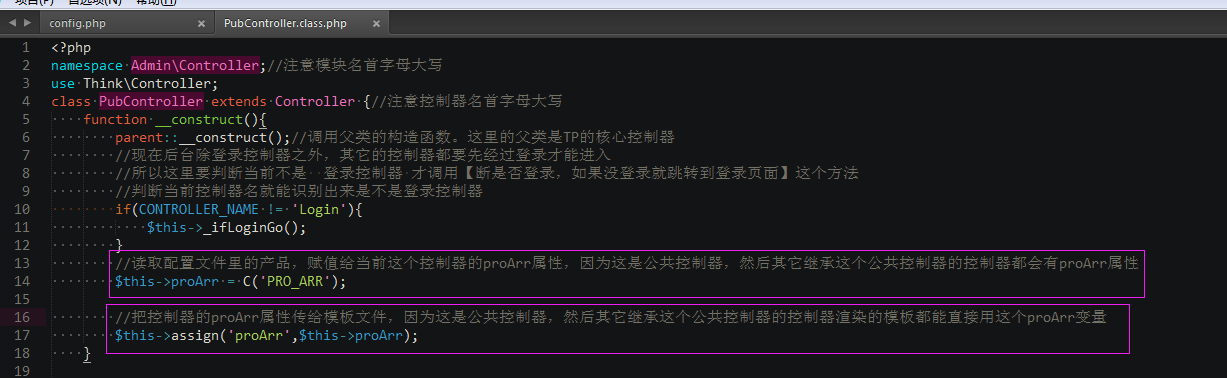
# 应用目录和入口文件

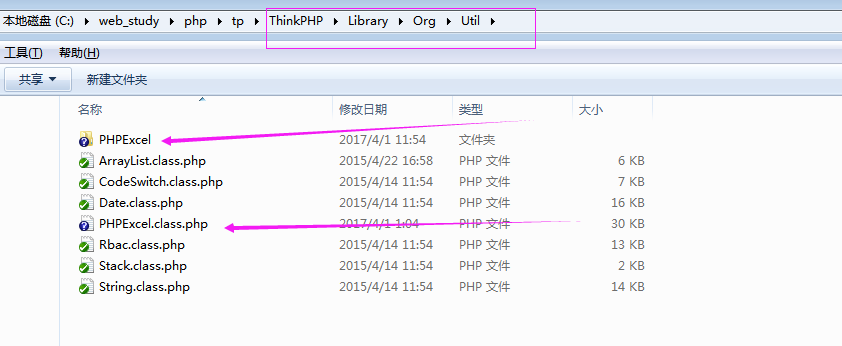


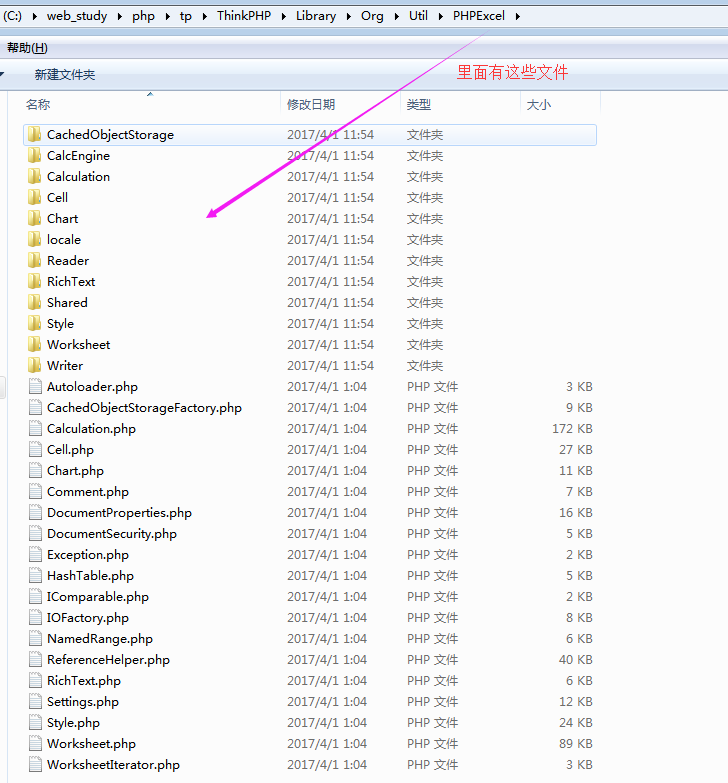
# 后台公用控制器



# 把phpexcel放到TP类库目录

PHPExcel，下载后把里面的PHPExcel.php重命名为PHPExcel.class.php，把PHPExcel.class.php和PHPExcel复制到：ThinkPHP\Library\Org\Util，如下图所示：





# phpexcel在TP的基本用法

PHPExcel的在TP里的基本用法:

public function toExcel(){

import('Org.Util.PHPExcel');//引入PHPExcel库文件

$excel = new \PHPExcel();//创建PHPExcel对象

$excel->setActiveSheetIndex(0)

->setCellValue('A1','hello')

->setCellValue('B2','b2 1234')

->setCellValue('C1','c1 agasfd')

->setCellValue('D2','d2 gg44');

$excel->getActiveSheet()->setCellValue('A8','a8' . "\r\n" . 'abc中文');

//$excel->getActiveSheet()->getRowDimension(8)->setRowHeight(-1);

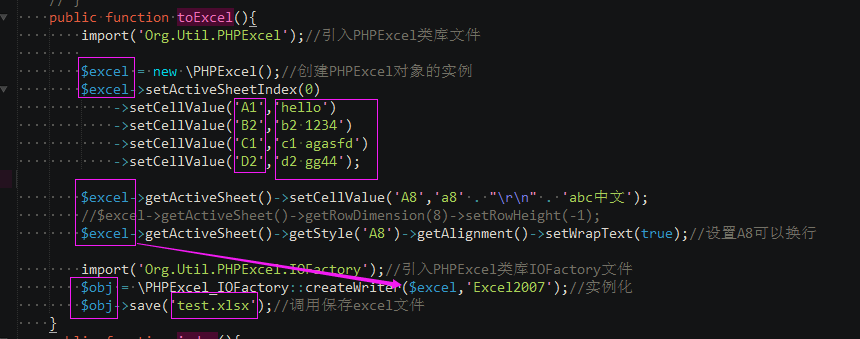
$excel->getActiveSheet()->getStyle('A8')->getAlignment()->setWrapText(true);

import('Org.Util.PHPExcel.IOFactory');

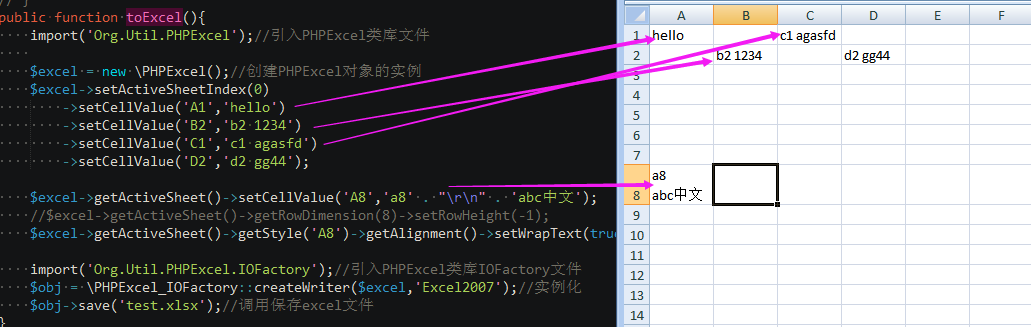
$obj = \PHPExcel\_IOFactory::createWriter($excel,'Excel2007');

$obj->save('test.xlsx');

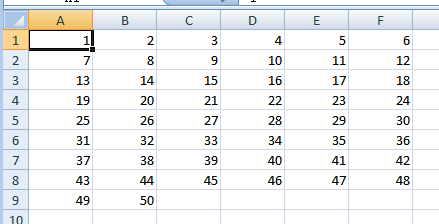
}



上例的操作被执行后页面会显示空白，但会在项目根目录（TP入口文件所在的目录）下生成一个test.xlsx文件，打开这个文件，可以看到是这样的：



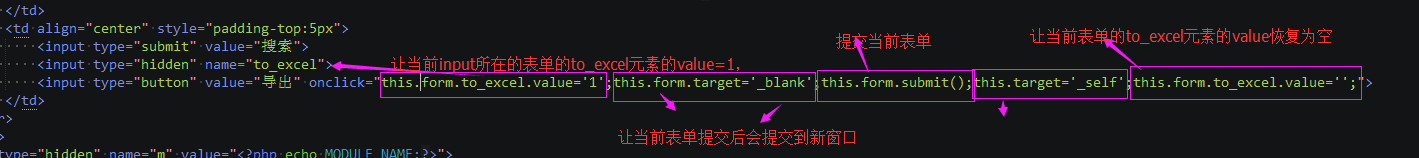
练习：在excel单元格里从1到50显示数字，每个单元格显示一个数字，每行只有前6个单元格显示数字:





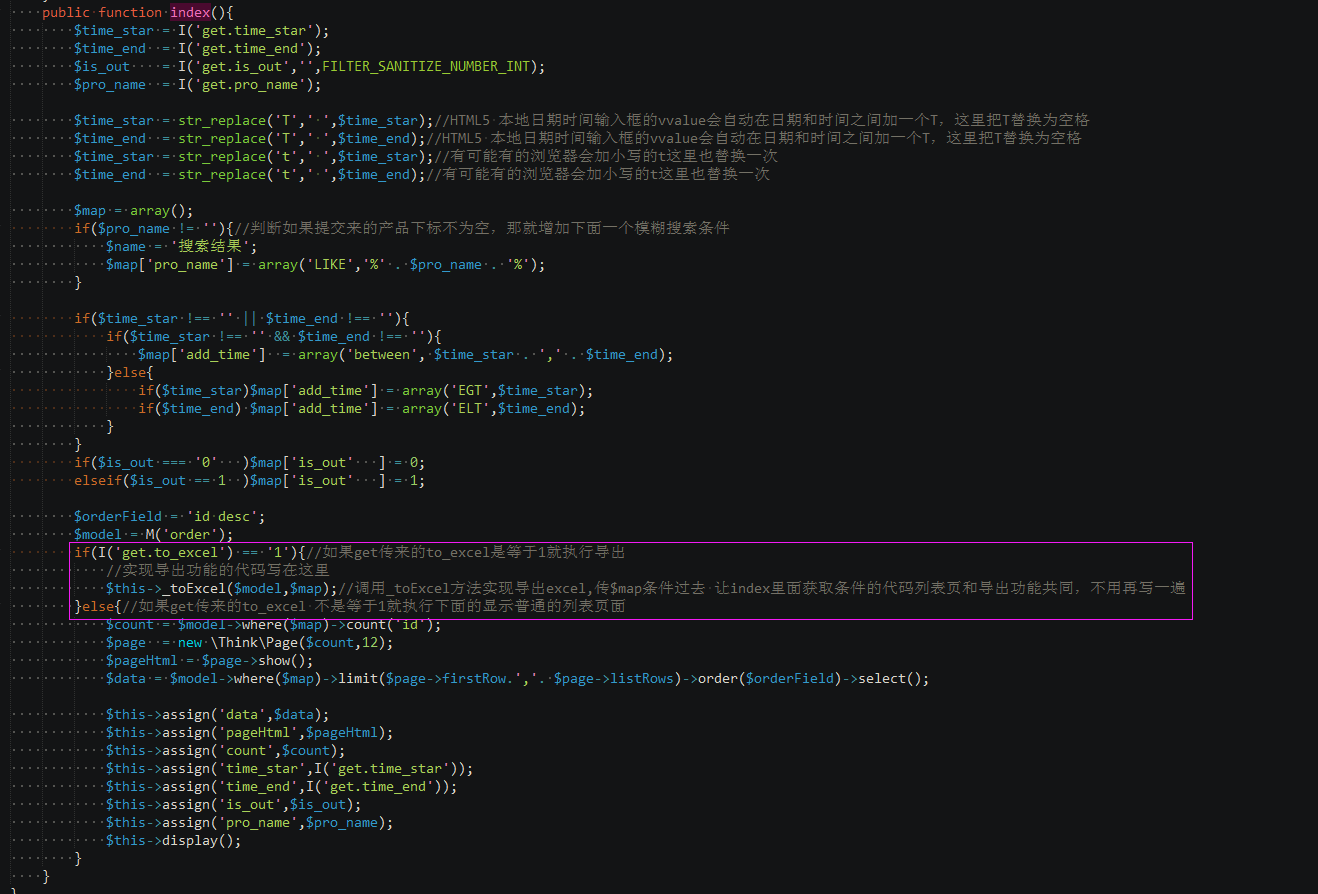
# 后台订单导出

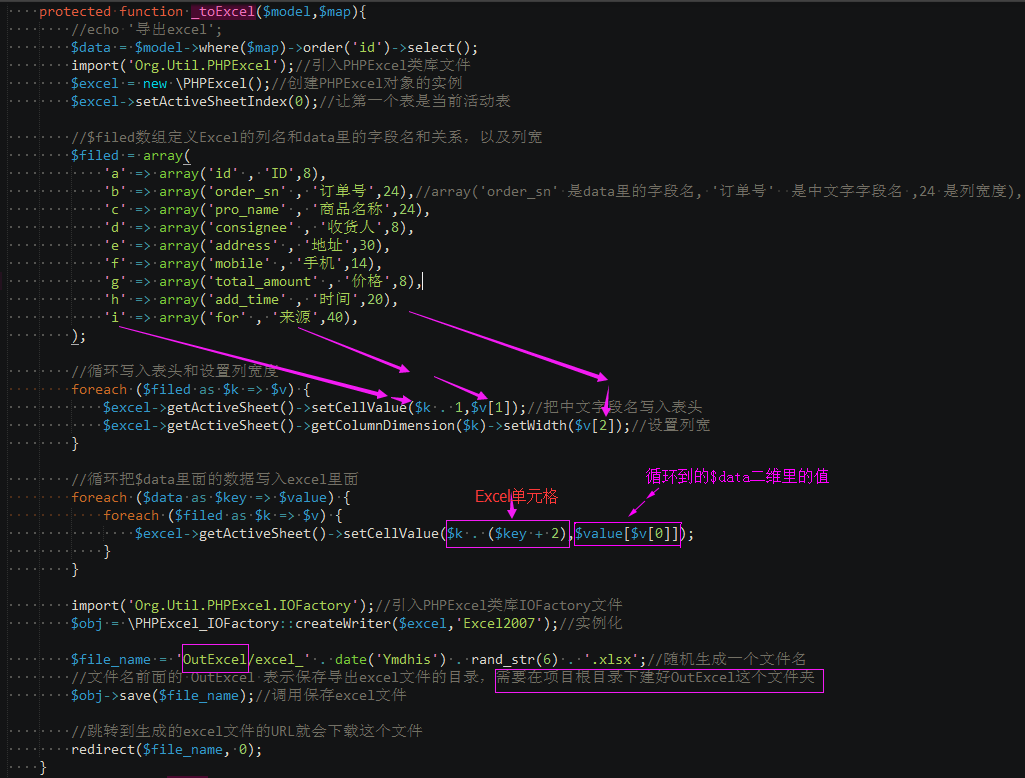
给订单列表的模板页面的导出按钮绑定点击事件：



上面截图里的JS代码应该改为this.form.to\_excel.value='1';this.form.target='\_blank';this.form.submit();this.form.target='\_self';this.form.to\_excel.value='';

控制器：





导出的结果：

