http://www.helloweba.com/view-blog-204.html

在一些复杂的系统中，要求对信息栏目进行无限级的分类，以增强系统的灵活性。那么PHP是如何实现无限级分类的呢？我们在本文中使用递归算法并结合mysql数据表实现无限级分类。

递归，简单的说就是一段程序代码的重复调用，当把代码写到一个自定义函数中，将参数等变量保存，函数中重复调用函数，直到达到某个条件才跳出，返回相应的数据。

**Mysql**

首先我们准备一张数据表class，记录商品分类信息。表中有三个字段，id：分类编号，主键自增长；title：分类名称；pid：所属上级分类id。

class表结构：

CREATE TABLE IF NOT EXISTS `class` (   
  `id` mediumint(6) NOT NULL AUTO\_INCREMENT,   
  `title` varchar(30) NOT NULL,   
  `pid` mediumint(6) NOT NULL DEFAULT '0',   
  PRIMARY KEY (`id`)   
) ENGINE=MyISAM  DEFAULT CHARSET=utf8;

插入数据后，如图：



**PHP**

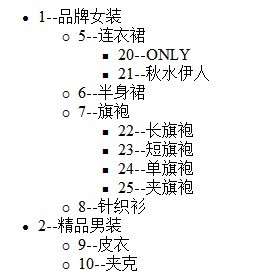
根据不同的需求，我们提供两种不同格式的自定义函数，一种是返回字符串，一种是返回数组，两种函数都使用了递归方法。先看返回字符串格式的函数：

function get\_str($id = 0) {   
    global $str;   
    $sql = "select id,title from class where pid= $id";    
    $result = mysql\_query($sql);//查询pid的子类的分类   
    if($result && mysql\_affected\_rows()){//如果有子类   
        $str .= '<ul>';   
        while ($row = mysql\_fetch\_array($result)) { //循环记录集   
            $str .= "<li>" . $row['id'] . "--" . $row['title'] . "</li>"; //构建字符串   
            get\_str($row['id']); //调用get\_str()，将记录集中的id参数传入函数中，继续查询下级   
        }   
        $str .= '</ul>';   
    }   
    return $str;   
}

以上函数get\_str()通过递归，不断查询下级分类，并最终返回字符串，大家可以根据项目需求修改其中的str，最终生成一个无限分级列表：

include\_once('connect.php'); //连接数据库，connect.php文件自己写一个啊   
echo get\_str(0); //输出无限级分类

效果如：



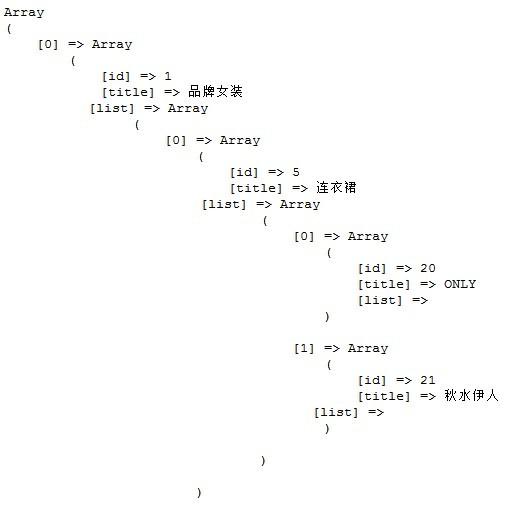
接着我们来看返回数组格式的函数，一样要使用递归：

function get\_array($id=0){   
    $sql = "select id,title from class where pid= $id";   
    $result = mysql\_query($sql);//查询子类   
    $arr = array();   
    if($result && mysql\_affected\_rows()){//如果有子类   
        while($rows=mysql\_fetch\_assoc($result)){ //循环记录集   
            $rows['list'] = get\_array($rows['id']); //调用函数，传入参数，继续查询下级   
            $arr[] = $rows; //组合数组   
        }   
        return $arr;   
    }   
}

函数get\_array()返回了数组，这是我们期待的，所以笔者推荐使用get\_array()得到数组，这样一来，我们可以对数组进行任意操作，比如我们可以将数组转换成json格式的数据传给前端页面，前端页面可以通过解析json数据灵活展示分类信息。比如树形结构的分类列表，下拉分类列表等。

include\_once('connect.php'); //连接数据库   
$list = get\_array(0); //调用函数   
print\_r($list); //输出数组

输出效果如：



如果要输出json格式的数据，则可使用：

echo json\_encode($list);