# 简介

在 Yii 中，创建新对象和初始化已存在对象时广泛使用配置。

配置通常包含被创建对象的类名和一组将要赋值给对象 [属性](https://www.yiichina.com/doc/guide/2.0/concept-properties)的初始值。 还可能包含一组将被附加到对象[事件](https://www.yiichina.com/doc/guide/2.0/concept-events)上的句柄。 和一组将被附加到对象上的[行为](https://www.yiichina.com/doc/guide/2.0/concept-behaviors)。

以下代码中的配置被用来创建并初始化一个数据库连接：



[Yii::createObject()](https://www.yiichina.com/doc/api/2.0/yii-baseyii" \l "createObject()-detail) 方法接受一个配置数组并根据数组中指定的类名创建对象。

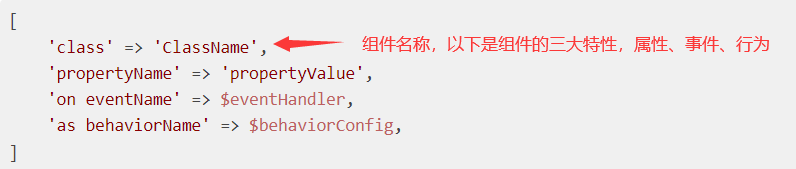
对象实例化后，剩余的参数被用来初始化对象的属性， 事件处理和行为。

对于已存在的对象，可以使用 [Yii::configure()](https://www.yiichina.com/doc/api/2.0/yii-baseyii" \l "configure()-detail) 方法根据配置去初始化其属性， 就像这样：

Yii::configure($object, $config);

# 配置的格式（Configuration Format）

一个配置的格式可以描述为以下形式：



其中

* class 元素指定了将要创建的对象的完全限定类名。
* propertyName 元素指定了对象属性的初始值。键名是属性名，值是该属性对应的初始值。 只有公共成员变量以及通过 getter/setter 定义的 [属性](https://www.yiichina.com/doc/guide/2.0/concept-properties)可以被配置。
* on eventName 元素指定了附加到对象[事件](https://www.yiichina.com/doc/guide/2.0/concept-events)上的句柄是什么。 请注意，数组的键名由 on 前缀加事件名组成。 请参考[事件](https://www.yiichina.com/doc/guide/2.0/concept-events)章节了解事件句柄格式。
* as behaviorName 元素指定了附加到对象的[行为](https://www.yiichina.com/doc/guide/2.0/concept-behaviors)。 请注意，数组的键名由 as 前缀加行为名组成。$behaviorConfig 值表示创建行为的配置信息，格式与我们之前描述的配置格式一样。

下面是一个配置了初始化属性值，事件句柄和行为的示例：



# 使用配置（Using Configurations）

Yii 中的配置可以用在很多场景。本章开头我们展示了如何使用 Yii::creatObject() 根据配置信息创建对象。本小节将介绍配置的两种 主要用法 —— 配置应用与配置小部件。

## 应用的配置（Application Configurations）

[应用](https://www.yiichina.com/doc/guide/2.0/structure-applications)的配置可能是最复杂的配置之一。

因为 [application](https://www.yiichina.com/doc/api/2.0/yii-web-application) 类拥有很多可配置的属性和事件。 更重要的是它的 [components](https://www.yiichina.com/doc/api/2.0/yii-di-servicelocator" \l "$components-detail) 属性可以接收配置数组并通过应用注册为组件。

以下是一个针对[基础应用模板](https://www.yiichina.com/doc/guide/2.0/start-installation)的应用配置概要：



配置中没有 class 键的原因是这段配置应用在下面的入口脚本中， 类名已经指定了。

(**new** yii\web\Application($config))->run();

更多关于应用 components 属性配置的信息可以查阅[应用](https://www.yiichina.com/doc/guide/2.0/structure-applications) 以及[服务定位器](https://www.yiichina.com/doc/guide/2.0/concept-service-locator)章节。

自版本 2.0.11 开始，系统配置支持使用 container 属性来配置[依赖注入容器](https://www.yiichina.com/doc/guide/2.0/concept-di-container) 例如：



请参考 [依赖注入容器](https://www.yiichina.com/doc/guide/2.0/concept-di-container) 下面的 [高级应用实例](https://www.yiichina.com/doc/guide/2.0/concept-di-container" \l "advanced-practical-usage) 获取更多 definitions 和 singletons 配置项和实际使用的例子。

## 小部件的配置（Widget Configurations）

使用[小部件](https://www.yiichina.com/doc/guide/2.0/structure-widgets)时，常常需要配置以便自定义其属性。

[yii\base\Widget::widget()](https://www.yiichina.com/doc/api/2.0/yii-base-widget" \l "widget()-detail) 和 [yii\base\Widget::begin()](https://www.yiichina.com/doc/api/2.0/yii-base-widget" \l "begin()-detail) 方法都可以用来创建小部件。 它们可以接受配置数组：

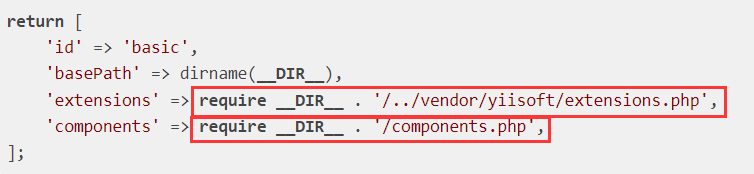


上述代码创建了一个小部件 Menu 并将其 activateItems 属性初始化为 false。 item 属性也配置成了将要显示的菜单条目。

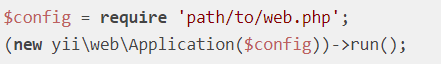
# 配置文件（Configuration Files）

当配置的内容十分复杂，通用做法是将其存储在一或多个 PHP 文件中， 这些文件被称为配置文件。

一个配置文件返回的是 PHP 数组。 例如，像这样把应用配置信息存储在名为 web.php 的文件中：



仅仅需要 “require”，就可以取得一个配置文件的配置内容，像这样：

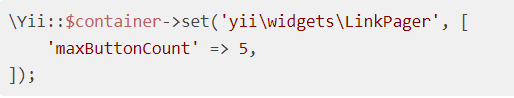


# 默认配置（Default Configurations）

[Yii::createObject()](https://www.yiichina.com/doc/api/2.0/yii-baseyii" \l "createObject()-detail) 方法基于[依赖注入容器](https://www.yiichina.com/doc/guide/2.0/concept-di-container)实现。

使用 Yii::creatObject() 创建对象时，可以附加一系列****默认配置****到指定类的任何实例。 默认配置还可以在[入口脚本](https://www.yiichina.com/doc/guide/2.0/runtime-bootstrapping) 中调用 Yii::$container->set() 来定义。

例如，如果你想自定义 [yii\widgets\LinkPager](https://www.yiichina.com/doc/api/2.0/yii-widgets-linkpager) 小部件，以便让分页器最多只显示 5 个翻页按钮（默认是 10 个）， 你可以用下述代码实现：



不使用默认配置的话，你就得在任何使用分页器的地方， 都配置 maxButtonCount 的值。

# 环境常量（Environment Constants）

配置经常要随着应用运行的不同环境更改。

例如在开发环境中， 你可能使用名为 mydb\_dev 的数据库， 而生产环境则使用 mydb\_prod 数据库。 为了便于切换使用环境，Yii 提供了一个定义在入口脚本中的 YII\_ENV 常量。 如下：

defined('YII\_ENV') **or** define('YII\_ENV', 'dev');

你可以把 YII\_ENV 定义成以下任何一种值：

* prod：生产环境。常量 YII\_ENV\_PROD 将被看作 true。 如果你没修改过，这就是 YII\_ENV 的默认值。
* dev：开发环境。常量 YII\_ENV\_DEV 将被看作 true。
* test：测试环境。常量 YII\_ENV\_TEST 将被看作 true。

有了这些环境常量，你就可以根据当下应用运行环境的不同，进行差异化配置。 例如，应用可以包含下述代码只在开发环境中开启 [调试工具](https://www.yiichina.com/doc/guide/2.0/tool-debugger)。

