在Java中，包是一种将类、接口和其他包分组在一起的机制。包的主要目的是帮助开发者组织代码，防止命名冲突，并控制访问级别。Java使用文件系统的目录作为包的物理表示，每个包对应于一个目录。

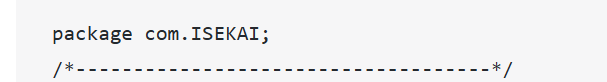
每个 C 程序都有必须命名为 main 的主函数。 main 函数充当程序执行的起点。 它通常通过将调用定向到程序中的其他函数来控制程序执行。

1. 包：为了方便管理Java程序，可以将类组织成包，包是一个命名空间，可以避免不同类的命名冲突。包的名称通常是一个反向的域名，例如com.example.hello，对应的包目录结构应该是com/example/hello。

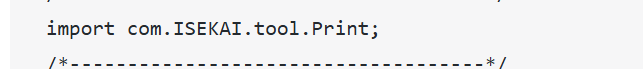
2.导入：完整类名，包名+类名

3.主类：程序入口，字节码文件名只和类名有关，有main函数即从此函数开始

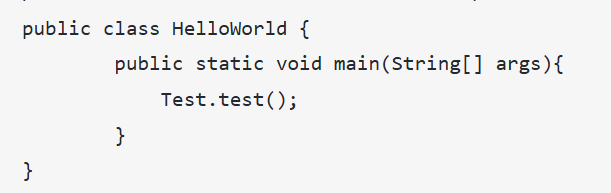
4.辅助类：辅助类包含了其他类用于执行重复任务的方法，这些任务并不是应用程序的核心目的。它们通过提供一些功能来帮助其他类。 它们包含用于实现琐碎和重复任务的方法，使整个代码库具有模块化和可重用性，可在多个类之间进行重用。



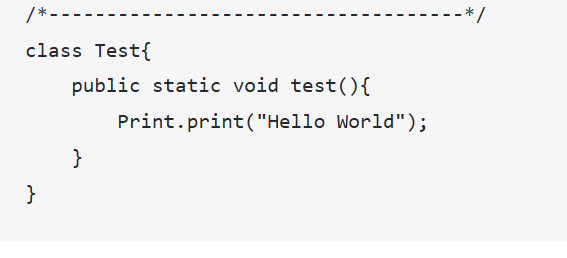
用package标识这个类，避免和类名重复，后面的域名就是包的名字



import就是在java文件开头的地方，先说明会用到那些类别，可以直接使用此类不需再写出完整类名。



Public作为修饰词修饰了class这个关键词，控制了类的访问权限，public修饰的类可以被其他包访问。这部分代码定义了一个名为`HelloWorld`的公共类，它是Java程序的入口点。`main`方法是Java应用程序的主方法，它是程序开始执行的地方。当运行一个Java程序时，JVM会调用这个类的`main`方法。在这个例子中，`main`方法调用了`Test`类的`test`方法。



辅助类定义，这部分代码定义了一个Test类

static静态方法通常用于执行与类相关的操作，而不需要访问或修改特定实例的状态。这个实例中利用这个方法调用了之前导入的`Print`类的`print`方法输出字符串，我们可以直接通过类名来调用，而不需要创建`Test`类的实例。

