**信息科学与工程学院**

**2020－2021学年第一学期**

实 验 报 告

课程名称： Java编程技术

实验名称： 实验一

专 业 班 级 通信工程 二班

学 生 学 号 201800121050

学 生 姓 名 孟麟芝

实 验 时 间 2020年9月11日

实验报告

【实验目的】

1.掌握安装SDK，安装Eclipse、IDEA等集成开发环境和安装EditPlus、VSCODE等编辑软件的方法。

2.掌握设置程序运行环境的方法。

3.掌握编写与运行程序的方法。

4.理解面向对象的编程思想。

【实验要求】

1.编写一个简单的控制台应用程序，该程序在命令行窗口输出两行文字：“Hello World!”和“We are students.”要求分别使用集成开发环境和命令行方式实现

2.在集成开发环境软件的练习中思考回答下面问题：

1) 编译器怎样提示丢失大括号的错误？

2) 编译器怎样提示语句丢失分号的错误？

3) 编译器怎样提示将System写成system这一错误？

【实验具体内容】

【第一个实验】

1.实验源代码（粘贴源代码）：

public class App {

    public static void main(String[] args) {

        System.out.println("Hello, World!");

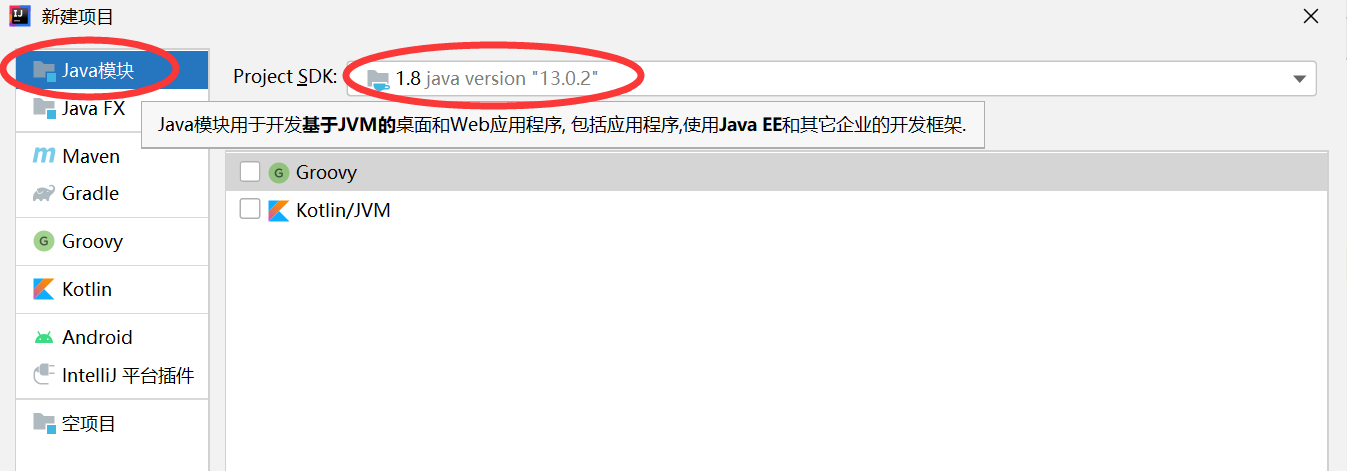
        System.out.println("We are students.");

    }

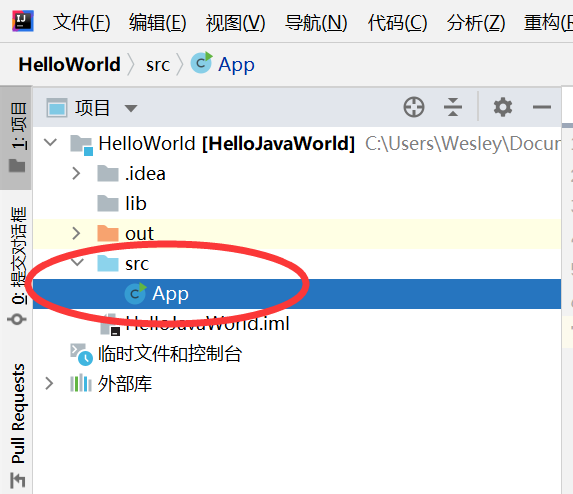
}

2.实验记录与说明分析：

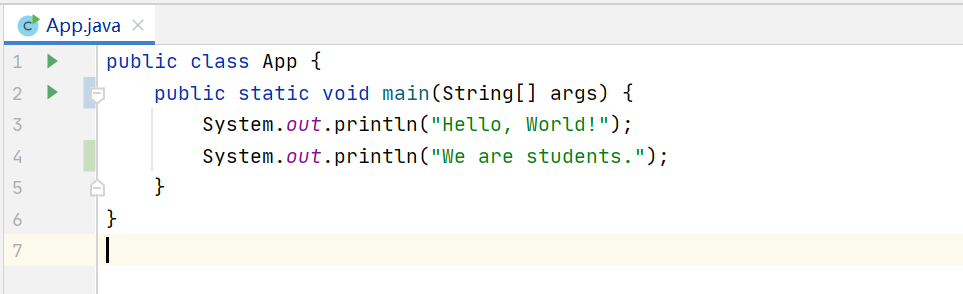
（1）先使用集成开发环境实现。在IDEA中新建一个项目，对于本实验的HelloWorld程序，选择建立JAVA模块即可（如图下所示），且无需基于任何现成模板。本机的JDK版本为13.0.2，并导入了完整的JRE



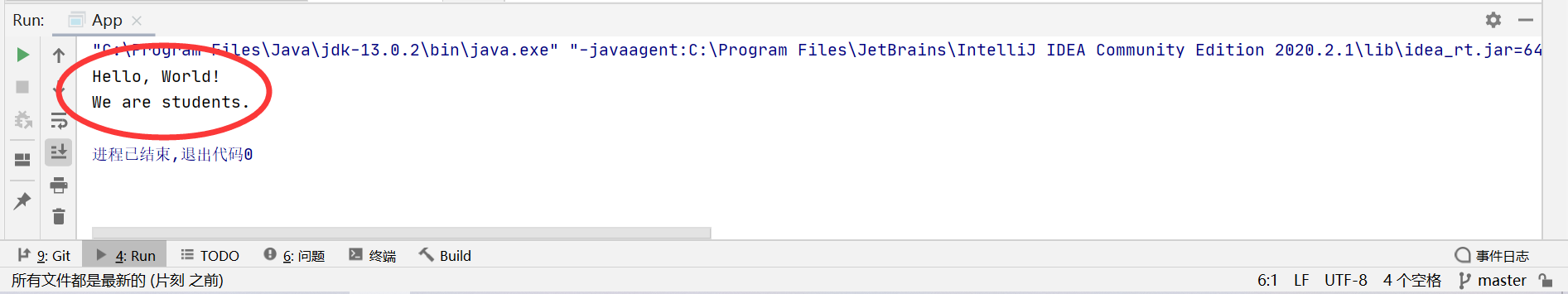
（2）在项目文件夹下的src文件夹根目录中新建一个App.java文件作为源文件，如图所示



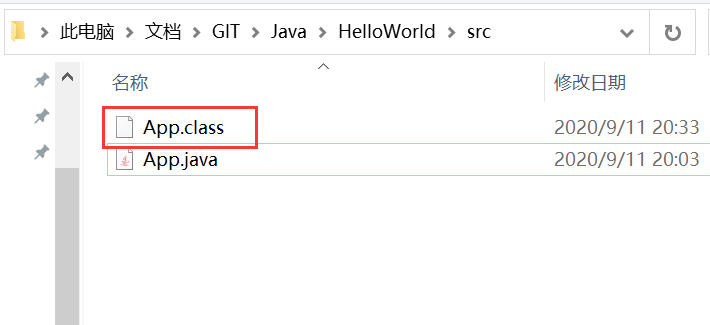
（3）在App.java中写入HelloWorld源代码，如图所示：



（4）编译无误后运行程序，可见如下结果：



（5）再使用命令行的方式实现，使用cd命令进入源文件所在文件夹，再使用javac编译，操作完成后可见目录下多出了一个.class文件



（6）再使用java指令运行.class文件，可见得到了所需要的结果



【第二个实验】

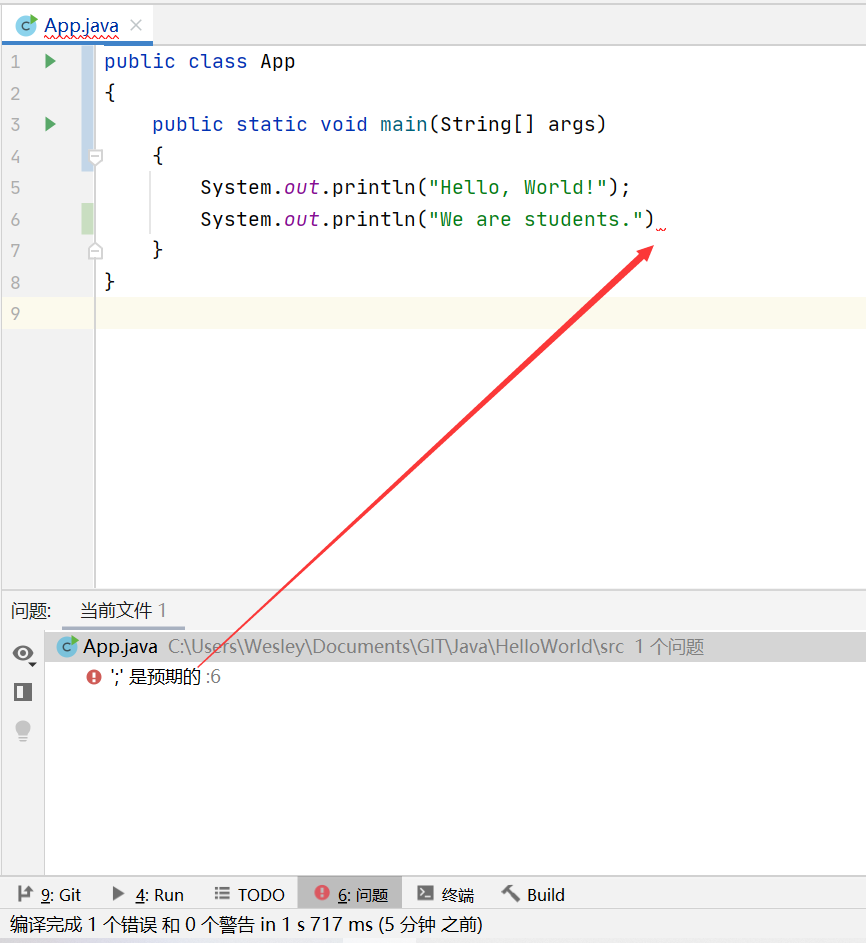
1) 编译器怎样提示丢失大括号的错误？

答：编译器在缺少大括号处用红色的波浪线标出，并在问题提示窗口写出了“预期}”及缺少大括号的代码所在行数



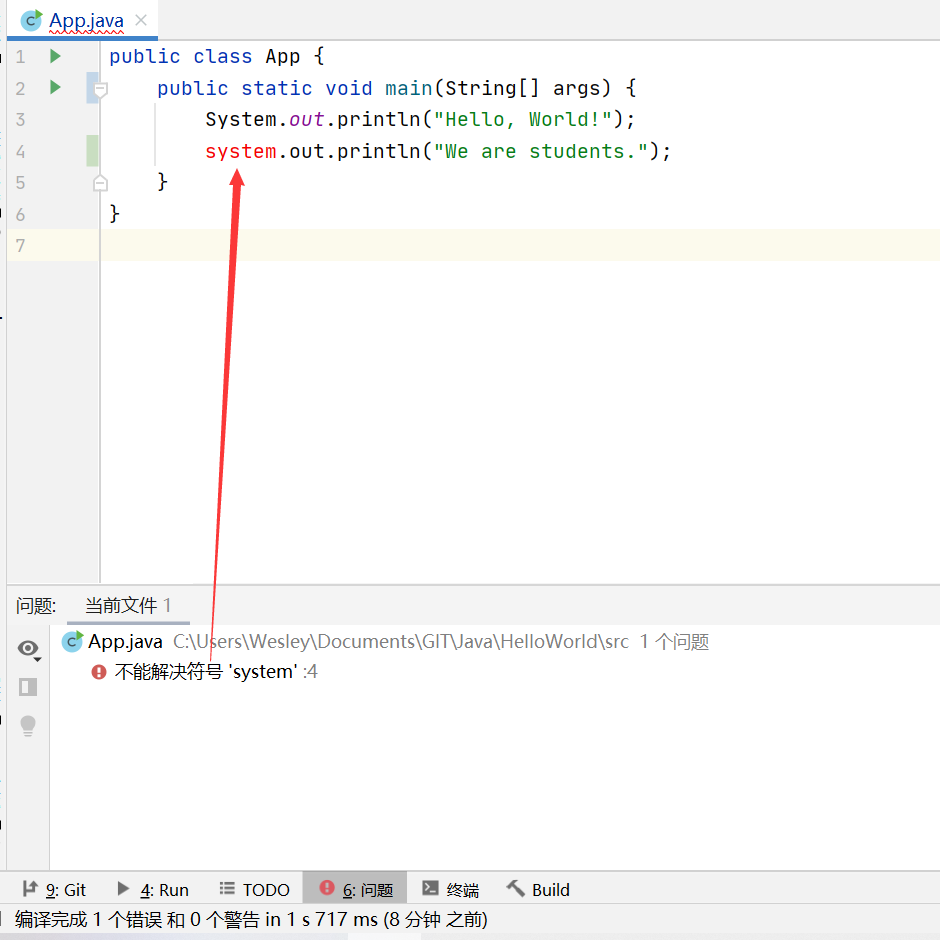
2) 编译器怎样提示语句丢失分号的错误？

答：提示方式与丢失大括号类似，也是用红色波浪线标出，并说明了缺少的符号和所在行数

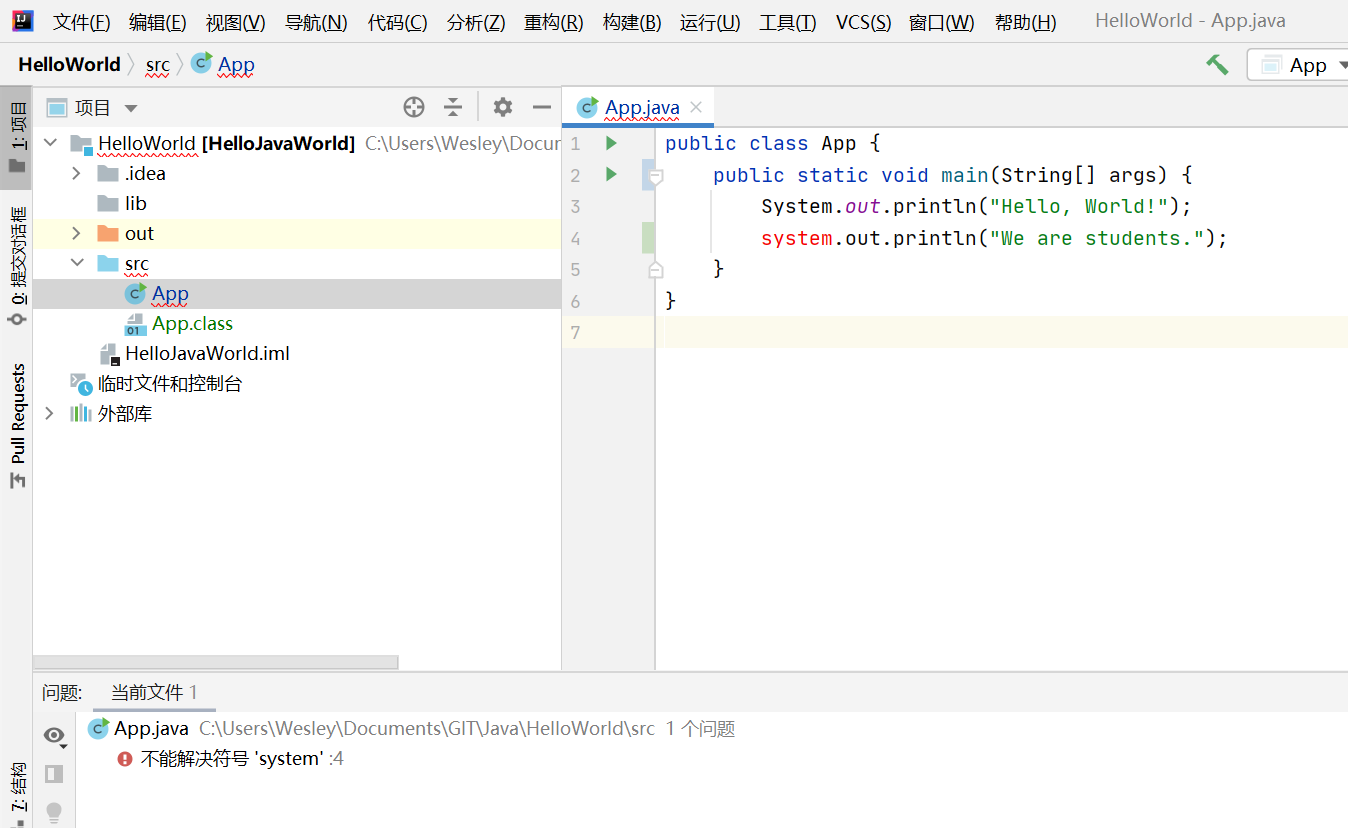


3) 编译器怎样提示将System写成system这一错误？

答：编译器直接将system标红，并在问题窗口作出提示，表示无法识别system这一符号



4) 另外，只要项目中的源文件出错，项目、出错文件所在文件夹、出错文件均会被用红色波浪线标出



【思考题】

1.什么是面向对象，什么是面向过程？

答：

从定义上讲，面向过程是一种以过程为中心 的编程思想。这些都是以什么正在发生为主要目标进行编程，不同于面向对象的是谁在受影响。

面向对象则是超越了程序设计的范畴，延伸到各个领域，它是一种对现实世界理解和抽象的方法，是计算机编程技术发展到一定阶段后的产物。面向对象是把构成问题事务分解成各个对象，建立对象的目的不是为了完成一个步骤，而是为了描叙某个事物在整个解决问题的步骤中的行为。面向对象在程序设计中一般是指“继承、封装、多态”的设计方法。

2.面向对象和面向过程之间的差异，试找一个例子进行解释？

答：

简单地进行理解，面向过程的程序要将每个细化的“操作”单列出来，可以作为主程序中的一行，也可作为一个子程序出现，如要实现“打篮球”，那么在主程序中，就要调用“投篮”、“走步”、“传球”等多个子程序（类），它们之间的耦合度很高，互相配合才能产生有意义的作用，且过程稍变，可能各个类都需要改变，扩展性并不好。但在较小的、不常更改的程序中，面向过程的设计是较为简洁直白的思路。

面向对象则不然，它站在更宏观的层次上，让每个类具有较为独立的功能，成为一个个独立的对象存在，互相之间耦合性较低。在大型项目中，代码量增多，问题的复杂程度变高，这时，面向对象的编程发挥着巨大的作用。

面向对象和面向过程的差异还可以通过这样一个例子来理解：

若进行各种数的加减乘除运算，一个面向过程的程序可能是这样呈现的：

1. **void** 整数加
2. {
3. ……
4. }
5. **void** 整数乘
6. {
7. ……
8. }
9. **void** 实数加
10. {
11. ……
12. }
13. **void** 实数乘
14. {
15. ……
16. }
17. ……

若是面向对象的程序，则可能是下面这样呈现的，可以看到，整数、实数、复数等成为了独立的对象，可以独立的完成加减乘除操作，互相之间耦合度很低。

1. **void** 整数
2. {
3. 加减乘除
4. }
5. **void** 实数
6. {
7. 加减乘除
8. }
9. **void** 复数
10. {
11. 加减乘除
12. }
13. ……

如果再考虑继承的概念，那么这个面向对象的程序可能会呈现出下面的效果：

1. **void** 整数
2. {
3. 加减乘除
4. }
5. **void** 实数
6. {
7. 沿用上面的设计
8. }
9. **void** 复数
10. {
11. 沿用上面的设计
12. }
13. ……

3.简单说说面向对象的基本概念和面向对象程序设计的特点和优点？

答：

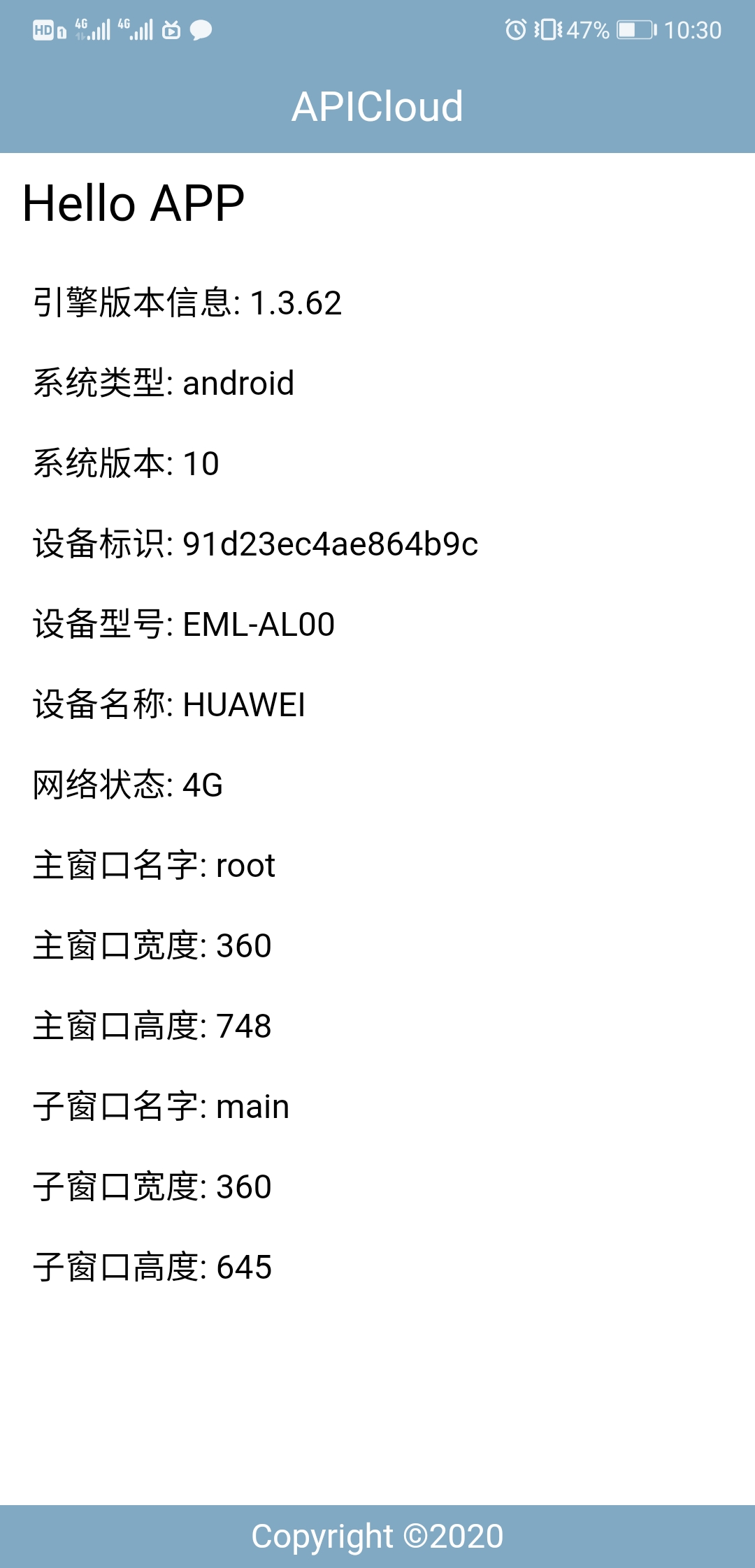
面向对象是把构成问题事务分解成各个对象，建立对象的目的不是为了完成一个步骤，而是为了描叙某个事物在整个解决问题的步骤中的行为。面向对象的概念早已超越了程序设计的范畴，延伸到各个领域。而面向对象在程序设计中一般是指“继承、封装、多态”的设计方法。

其优点在于，减小了模块之间的耦合，方便了模块的转移和互用，所以每个类几乎都可以做到“一次编写，处处运行”。项目体量很大、复杂程度较高时，面向对象的程序设计思路发挥着巨大作用。

4.尝试使用APICloud或DCCloud编写简单的APP？

答：

由于未曾学习JS，CSS，HTML等相关知识，直接按网站教程就可生成一个默认程序，进行在线云编译，即可下载到手机进行安装运行。



【实验心得】

工欲善其事必先利其器，程序开发环境的配置是开始编程的重要一环。

在这学期之前，我们已经学习或了解过了C、汇编、Matlab、VHDL、Python等语言，这学期更是C#、Java双管齐下。在学习过程中，我们开始逐渐意识到了它们之间配置的相同之处与不同之处。使用任何一种语言进行开发，都离不开环境的配置。理论上，安装配置好了SDK，再加一个记事本软件，便已经可以写一些简短的程序了。不过集成开发环境（IDE）和一些编辑器（Notepad++、VScode）则提供了更友好的界面，更方便的项目管理方式。另外，基本现今的各种IDE、编辑器都可以配合GIT使用，这使项目版本管理变得十分方便，也可以方便的在github上存放自己的代码。

虽然，这些方便快捷的工具对于初学似乎无用，但在今后的学习、工作中，希望它们会派上用场。

老师课上的讲解，也让我们意识到了编程需要遵守一些约定俗成的命名规则，加以注意，可以提高自己代码的规范性。

面向对象、面向过程的概念、区别已经是老生常谈，这一次实验学习中，结合老师所举例子以及网上搜集的资料，又对它们有了进一步的理解。但缺乏实际程序设计，对它们仍然没有切身的感受，似乎面前还有一层窗纱，希望在这一学期的学习中可以“拨云见日”。