**实验六**

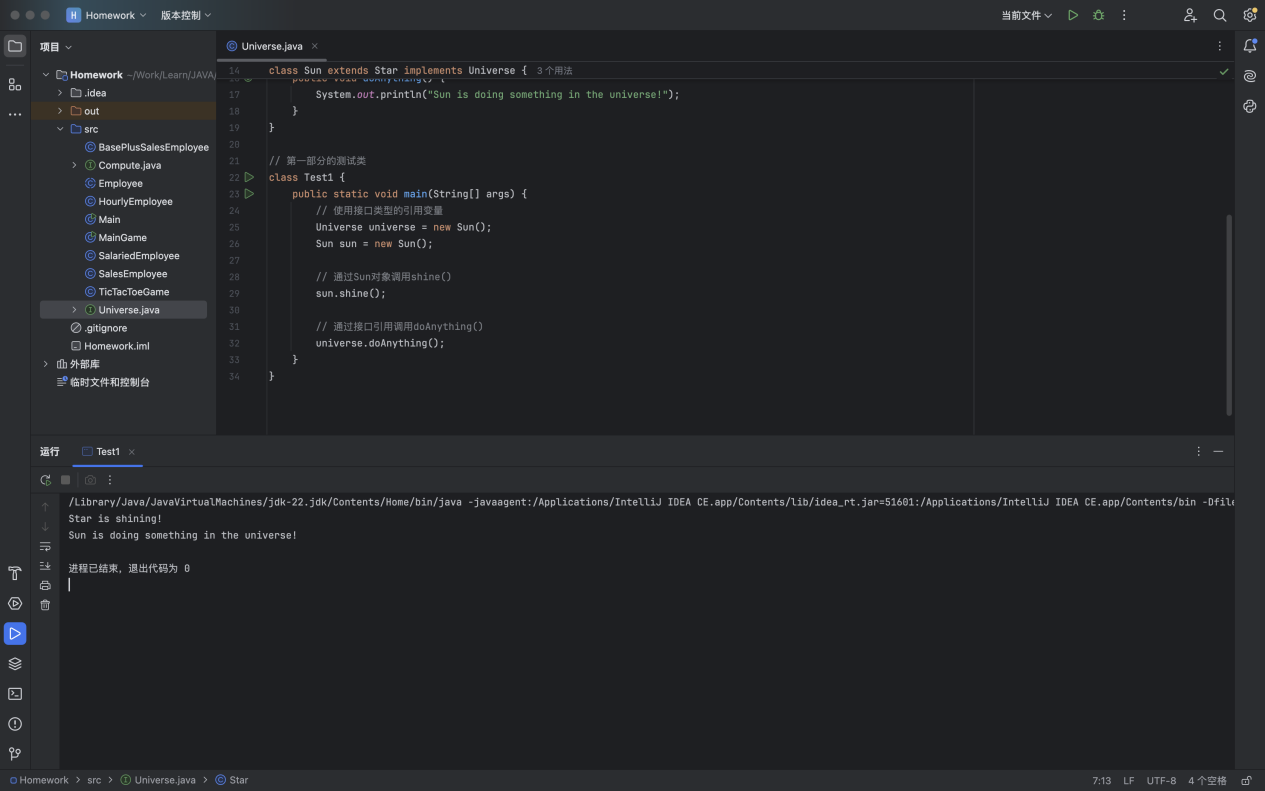
**1.题目：**

|  |
| --- |
| **1、定义接口Universe，含抽象方法doAnything()。类Star：含shine()；派生类Sun，继承Star，分别实现Universe接口。测试类中，实现用接口类型的引用变量去访问shine()和doAnything()。** |

**完整代码：**

|  |
| --- |
| Java // Universe接口定义 interface Universe {  void doAnything(); }  // Star类定义 class Star {  public void shine() {  System.*out*.println("Star is shining!");  } }  // Sun类继承Star并实现Universe接口 class Sun extends Star implements Universe {  @Override  public void doAnything() {  System.*out*.println("Sun is doing something in the universe!");  } }  // 测试类 class Test1 {  public static void main(String[] args) {  // 使用接口类型的引用变量  Universe universe = new Sun();  Sun sun = new Sun();   // 通过Sun对象调用shine()  sun.shine();   // 通过接口引用调用doAnything()  universe.doAnything();  } } |

**代码运行截图：**



**2.题目：**

|  |
| --- |
| 2、利用接口做参数，完成加减乘除运算：  **（1）定义接口Compute，含有一个方法int computer(int n, int m)。（2）四个运算类实现该规则，分别完成加减乘除运算。（3）设计类UseCompute，类中含有方法：**  **public static void useCom(Compute com, int one, int two)，**  **此方法能够用传递过来的对象调用computer方法完成运算，并输出运算的结果。（4）主类Test中，调用UseCompute类的方法useCom来完成加减乘除运算。** |

**完整代码：**

|  |
| --- |
| Java // Compute接口定义 interface Compute {  int computer(int n, int m); }  // 加法实现 class Add implements Compute {  @Override  public int computer(int n, int m) {  return n + m;  } }  // 减法实现 class Subtract implements Compute {  @Override  public int computer(int n, int m) {  return n - m;  } }  // 乘法实现 class Multiply implements Compute {  @Override  public int computer(int n, int m) {  return n \* m;  } }  // 除法实现 class Divide implements Compute {  @Override  public int computer(int n, int m) {  if (m == 0) {  throw new ArithmeticException("除数不能为0");  }  return n / m;  } }  // UseCompute类定义 class UseCompute {  public static void useCom(Compute com, int one, int two) {  int result = com.computer(one, two);  System.*out*.println("计算结果是：" + result);  } }  // 测试类 class Test2 {  public static void main(String[] args) {  // 测试加法  UseCompute.*useCom*(new Add(), 10, 5);   // 测试减法  UseCompute.*useCom*(new Subtract(), 10, 5);   // 测试乘法  UseCompute.*useCom*(new Multiply(), 10, 5);   // 测试除法  UseCompute.*useCom*(new Divide(), 10, 5);  } } |

**代码运行截图：**

