1 定义一个人类，此人类有如下成员: 姓名

有两个重载构造函数初始化 姓名；

有个私有静态字段统计人类的数量；

有个静态函数用来初始化静态字段；

有个静态方法获取人类的数量；

最后创建人类对象进行验证。

2.定义一个员工类，继承了人类，员工类有薪水属性

有多个重载构造函数初始化 姓名、薪水，有一个方法输出员工姓名与薪水；

最后创建两个员工类对象进行验证。

3. 定义一个学生类，继承人类，学生类有学号与成绩属性,有个静态成员统计学生个数

有多个重载构造函数初始化属性。最后创建两个学生类对象进行验证，并输出学生人数与人类个数。

public class Person {

// 成员变量

public String name;

// 静态字段

private static int personCount;

// 构造函数1

public Person(String name) {

this.name = name;

personCount++;

}

// 构造函数2

public Person() {

personCount++;

}

// 初始化静态字段的静态方法

public static void initialize() {

personCount = 0;

}

// 获取人类数量的静态方法

public static int getPersonCount() {

return personCount;

}

}

// 员工类，继承人类

class Employee extends Person {

// 薪水属性

public double salary;

// 构造函数1

public Employee(String name, double salary) {

super(name);

this.salary = salary;

}

// 构造函数2

public Employee() {

super();

}

// 输出员工姓名与薪水的方法

public void displayInfo() {

System.out.println("Employee Name: " + name + ", Salary: " + salary);

}

}

// 学生类，继承人类

class Student extends Person {

// 学号与成绩属性

public int studentNumber;

public double grade;

// 静态成员统计学生个数

private static int studentCount;

// 构造函数1

public Student(String name, int studentNumber, double grade) {

super(name);

this.studentNumber = studentNumber;

this.grade = grade;

studentCount++;

}

// 构造函数2

public Student() {

super();

studentCount++;

}

// 输出学生信息的方法

public void displayInfo() {

System.out.println("Student Name: " + name + ", Student Number: " + studentNumber + ", Grade: " + grade);

}

// 获取学生数量的静态方法

public static int getStudentCount() {

return studentCount;

}

}

public class Main {

public static void main(String[] args) {

// 初始化人类数量

Person.initialize();

// 创建人类对象并验证

Person person1 = new Person("John");

Person person2 = new Person("Alice");

System.out.println("Person Count: " + Person.getPersonCount());

// 创建员工类对象并验证

Employee employee1 = new Employee("Bob", 50000.0);

Employee employee2 = new Employee();

employee1.displayInfo();

employee2.displayInfo();

// 创建学生类对象并验证

Student student1 = new Student("Tom", 12345, 85.5);

Student student2 = new Student();

student1.displayInfo();

student2.displayInfo();

System.out.println("Student Count: " + Student.getStudentCount());

System.out.println("Person Count: " + Person.getPersonCount());

}

}