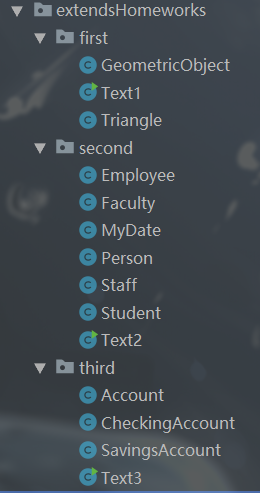
**作业目录结构**

**11.1** (The Triangle class) Design a class named Triangle that extends GeometricObject. The class contains:

(三角形类 Triangle) 设计一个名为 Triangle 的类来扩展 GeometricObject 类。该类包括：

■ Three doubledata fields named side1, side2, and side3with default values 1.0 to denote three sides of the triangle.

三个名为 sidel1、side2 和 side3 的 double 数据域表示这个三角形的三条边，它们的默认 值是 1.0

■ A no-arg constructor that creates a default triangle.

一个无参构造方法创建默认的三角形

■ A constructor that creates a triangle with the specified side1, side2, and side3.

一个能创建带指定 side1、side2 和 side3 的三角形的构造方法。

■ The accessor methods for all three data fields.

所有数据域的访问器方法

■ A method named getArea()that returns the area of this triangle.

一个名为 getArea()的方法返回这个三角形的面积。

■ A method named getPerimeter()that returns the perimeter of this triangle.

一个名为 getPerimeter() 的方法返回这个三角形的周长。

■ A method named toString()that returns a string description for the triangle.

一个名为 toString（） 的方法返回这个三角形的字符串描述。

For the formula to compute the area of a triangle, see Programming Exercise 2.19.

The toString()method is implemented as follows:

return "Triangle: side1 = "+ side1 + " side2 = "+ side2 + " side3 = "+ side3;

计算三角形面积的公参见编程练习题 2.19。toStrin() 方法的实现如下所示：

return"Triangle: side1 = "+ side1 + " side2 = "+ side2 + " side3 = "+ side3;

Draw the UML diagrams for the classes Triangle and GeometricObject and implement the classes. Write a test program that prompts the user to enter three sides of the triangle, a color, and a Boolean value to indicate whether the triangle is filled. The program should create a Triangle object with these sides and set the colorand filledproperties using the input. The program should display the area, perimeter, color, and true or false to indicate whether it is filled or not.

画出 Triangle 类和 GeometricObject 类的 UML 图，并实现这些类。编写一个测试程序，

提示用户输入三角形的三条边、顔色以及一个 Boolean 值表明该三角形是否填充。程序应该使用输人创建一个具有这些边并设置 color 和 fllied 属性的三角形。程序应该显示面积、边长、 顔色以及表明是否填充的真或者假的值

GeometricObject

- color: String

- filled: boolean

- dateCreated: Date

+ GeometricObject()

+ GeometricObject(color: String, filled: boolean)

+ getColor(): String

+ setColor(color: String): void

+ isFilled(): void

+ setFilled(filled: boolean): boolean

+ getDateCreated(): Date

+ toString(): String

Triangle

- side1: double

- side2: double

- side3: double

+ Triangle ()

+ Triangle (side1: double, side2: double, side3: double)

+ getSide1(): double

+ getSide2(): double

+ getSide3(): double

+ setSide1(side1: double): void

+ setSide2(side2: double): void

+ setSide3(side3: double): void

+ getArea(): double

+ getPerimeter(): double

**GeometricObject.java**

package extendsHomeworks.first;

import java.util.Date;

public class GeometricObject {

private String color = "white";

private boolean filled;

private Date dateCreated;

public GeometricObject() {

dateCreated = new Date();

}

public GeometricObject(String color, boolean filled) {

dateCreated = new Date();

this.color = color;

this.filled = filled;

}

public String getColor() {

return color;

}

public void setColor(String color) {

this.color = color;

}

public boolean isFilled() {

return filled;

}

public void setFilled(boolean filled) {

this.filled = filled;

}

public Date getDateCreated() {

return dateCreated;

}

public String toString() {

return "created on " + dateCreated + "\ncolor: " + color + " and filled: " + filled;

}

}

**Triangle.java**

package extendsHomeworks.first;

public class Triangle extends GeometricObject {

private double side1 = 1.0;

private double side2 = 1.0;

private double side3 = 1.0;

public Triangle() {

}

public Triangle(double side1, double side2, double side3) {

this.side1 = side1;

this.side2 = side2;

this.side3 = side3;

}

double getSide1() {

return side1;

}

void setSide1(double side1) {

this.side1 = side1;

}

double getSide2() {

return side2;

}

void setSide2(double side2) {

this.side2 = side2;

}

double getSide3() {

return side3;

}

void setSide3(double side3) {

this.side3 = side3;

}

double getArea() {

//海伦公式

double p = (side1 + side2 + side3) / 2;

return Math.sqrt(p \* (p - side1) \* (p - side2) \* (p - side3));

}

double getPerimeter() {

return side1 + side2 + side3;

}

@Override

public String toString() {

return "Triangle: side1 = " + side1 + " side2 = " + side2 + " side3 = " + side3;

}

}

**Text1.java**

package extendsHomeworks.first;

import java.util.Scanner;

public class Text1 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

Triangle t= new Triangle();

System.out.println("请输入三边长：");

t.setSide1(sc.nextDouble());

t.setSide2(sc.nextDouble());

t.setSide3(sc.nextDouble());

System.out.println("请输入颜色");

t.setColor(sc.next());

System.out.println("是否被填充");

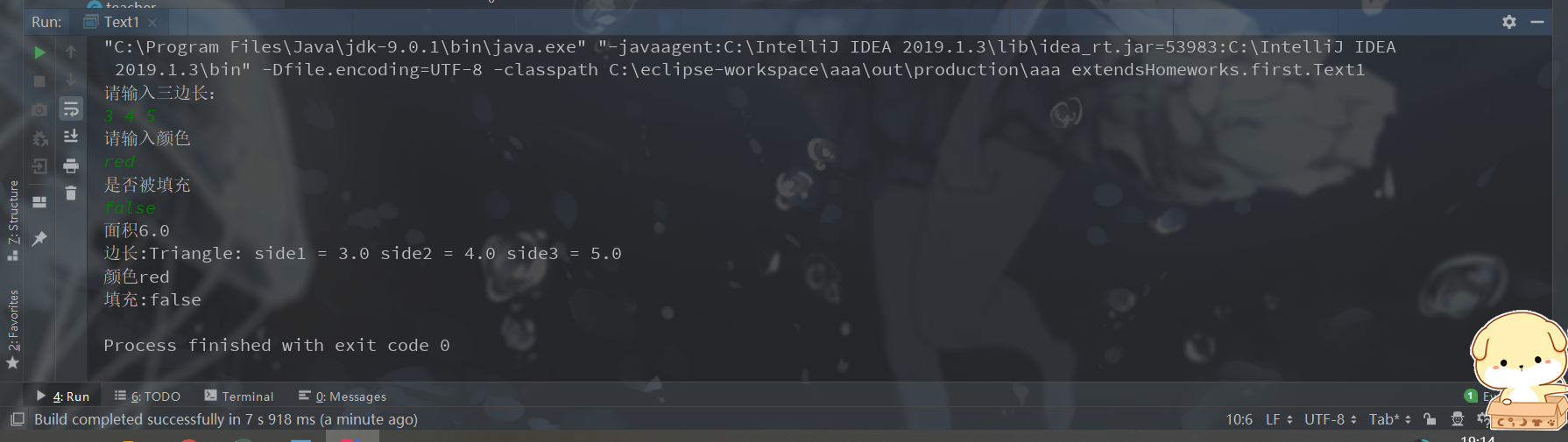
t.setFilled(sc.nextBoolean());

System.out.println("面积"+t.getArea()+"\n边长:"+t+"\n颜色"+t.getColor()+"\n填充:"+t.isFilled());

sc.close();

}

}



**11.2** (The Person, Student, Employee, Faculty, and Staff classes) Design a class named Person and its two subclasses named Student and Employee. Make Faculty and Staff subclasses of Employee. A person has a name, address, phone number, and email address. A student has a class status (freshman, sophomore, junior, or senior). Define the status as a constant. An employee has an office, salary, and date hired. Use the MyDateclass defined in Programming Exercise 10.14 to create an object for date hired. A faculty member has office hours and a rank. A staff member has a title. Override the toString method in each class to display the class name and the person’s name.

( Person、Student、Employee、Faculty 和 Staff 类）设计一个名为 Person 的类和它的两个

名为 Student 和 Employee 的子类。Employee 类又有子类：教员类 Faculty 和职员类 Staff。

每个人都有姓名、地址、电话号码和电子邮件地址。学生有班级状态（大一、大二、大三或大四）。将这些状态定义为常量。一个雇员涉及办公室、工资和受聘日期。使用编程练习题 10.14 中定义的 MyDate 类为受聘日期创建一个对象。教员有办公时间和级别。职员有职称号。覆盖 每个类中的 toString 方法，显示相应的类别名字和人名。

Draw the UML diagram for the classes and implement them. Write a test program that creates a Person, Student, Employee, Faculty, and Staff, and invokes their toString()methods.

画出这些类的UML图并实现这些类。编写一个测试程序，创建 Person、Student、 Employee、Faculty 和 Staff,并且调用它们的 toString() 方法。

Faculty

- workTime: String

- rank: String

+ Faculty ()

+ Faculty (name: String, address: String, tel: long, email: String, worktime: String, rank: String)

+ getWorkTime (): String

+ setWorkTime (workTime: String): void

+ getRank (): String

+ setRank (rank: String): void

Student

+ ONE: String

+ TWO: String

+ THREE: String

+ FOUR: String

+ grade: String

+ Student ()

+ Student (name: String, address: String, tel: long, email: String, grade: String)

+ getGrade (): String

+ setGrade (grade: String): void

Person

- name: String

- address: String

- tel: long

- email: String

+ Person ()

+ Person (name: String, address: String, tel: long, email: String)

+ getName(): String

+ setName(name: String): void

+ getAddress(): String

+ setAddress(address: String): void

+ getTel(): long

+ setTel(tel: long): void

+ getEmail(): String

+ setEmail(email: String): void

+ toString(): String

Staff

- titleNum: String

+ Staff ()

+ Staff (name: String, address: String, tel: long, email: String, titleNum)

+ getTitleNum (): String

+ setTitleNum (titleNum: String): void

Employee

- office: String

- salary: int

- employedDate: MyDate

+ Employee ()

+ Employee (name: String, address: String, tel: long, email: String, office: String, salary: int, employedDate : MyDate)

+ getOffice (): String

+ setOffice(office: String): void

+ getSalary (): int

+ setSalary (salary: int): void

+ getEmployedDate (): MyDate

+ setEmployedDate (employedDate: MyDate): void

**Person.java**

package extendsHomeworks.second;

public class Person {

//private:当前类

//protected:同包或其子孙类可访问

//空:同包

//public:任意位置

protected String name;

protected String address;

protected long tel;

protected String email;

public Person() {

}

public Person(String name, String address, long tel, String email) {

this.name = name;

this.address = address;

this.tel = tel;

this.email = email;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public long getTel() {

return tel;

}

public void setTel(long tel) {

this.tel = tel;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

@Override

public String toString() {

return "People{"

+ "name=" + name

+ ", address=" + address

+ ", tel=" + tel

+ ", email=" + email + "}";

}

}

**Student.java**

package extendsHomeworks.second;  
  
public class Student extends Person {  
 public final static String ONE = "Freshman";  
 public final static String TWO = "Sophomore";  
 public final static String THREE = "Junior";  
 public final static String FOUR = "Senior";  
 private String grade;//年级  
  
 public Student() {  
 super();  
 }  
  
 public Student(String name, String address, long tel, String email, String grade) {  
 super(name, address, tel, email);  
 this.grade = grade;  
 }

public String getGrade() {

return grade;

}

public void setGrade(String grade) {

this.grade = grade;

}

@Override  
 public String toString() {  
 return "Student{"  
 + "name=" + name  
 + ", address=" + address  
 + ", tel=" + tel  
 + ", email=" + email  
 + ", grade=" + grade + "}";  
 }  
  
}

**Employee.java**

package extendsHomeworks.second;  
  
public class Employee extends Person {  
 protected String office;  
 protected int salary;  
 protected MyDate employedDate;  
  
 public Employee() {  
 super();  
 }  
  
 public Employee(String name, String address, long tel, String email, String office, int salary,  
 MyDate employedDate) {  
 super(name, address, tel, email);  
 this.office = office;  
 this.salary = salary;  
 this.employedDate = employedDate;  
 }  
  
 public String getOffice() {  
 return office;  
 }  
  
 public void setOffice(String office) {  
 this.office = office;  
 }  
  
 public int getSalary() {  
 return salary;  
 }  
  
 public void setSalary(int salary) {  
 this.salary = salary;  
 }  
  
 public MyDate getEmployedDate() {  
 return employedDate;  
 }  
  
 public void setEmployedDate(MyDate employedDate) {  
 this.employedDate = employedDate;  
 }  
  
 @Override  
 public String toString() {  
 return "Employee{"  
 + "name=" + name  
 + ", address=" + address  
 + ", tel=" + tel  
 + ", email=" + email  
 + ", office=" + office  
 + ", salary=" + salary  
 + ", employedDate=" + employedDate + "}";  
 }  
  
}

**Faculty.java**

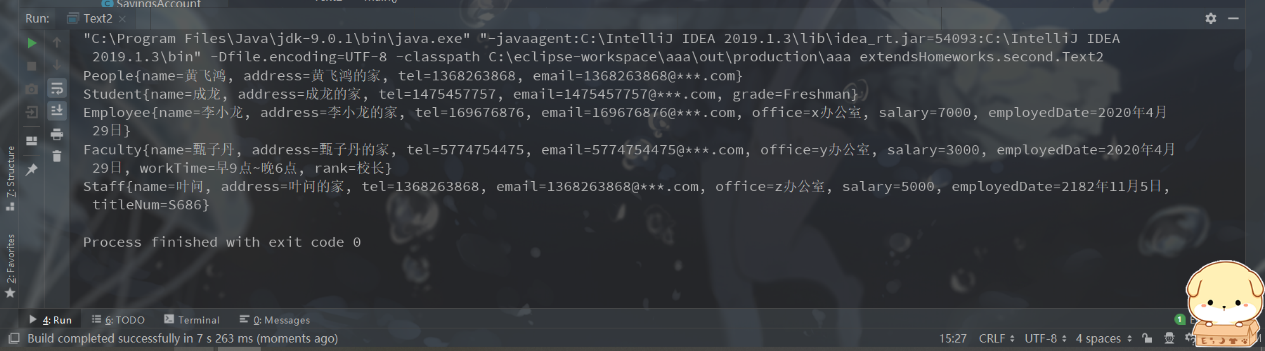
package extendsHomeworks.second;  
  
class Faculty extends Employee {  
 private String workTime;// 工作时间  
 private String rank;// 级别  
  
 public Faculty() {  
 super();  
 }  
  
 public Faculty(String name, String address, long tel, String email, String office, int salary, MyDate employedDate,  
 String workTime, String rank) {  
 super(name, address, tel, email, office, salary, employedDate);  
 this.workTime = workTime;  
 this.rank = rank;  
 }  
  
 public String getWorkTime() {  
 return workTime;  
 }  
  
 public void setWorkTime(String workTime) {  
 this.workTime = workTime;  
 }  
  
 public String getRank() {  
 return rank;  
 }  
  
 public void setRank(String rank) {  
 this.rank = rank;  
 }  
  
 @Override  
 public String toString() {  
 return "Faculty{"  
 + "name=" + name  
 + ", address=" + address  
 + ", tel=" + tel  
 + ", email=" + email  
 + ", office=" + office  
 + ", salary=" + salary  
 + ", employedDate=" + employedDate  
 + ", workTime=" + workTime  
 + ", rank=" + rank + "}";  
 }  
  
  
}

**Staff.java**

package extendsHomeworks.second;  
  
class Staff extends Employee {  
 private String titleNum;  
  
 public Staff() {  
 super();  
 }  
  
 public Staff(String name, String address, long tel, String email, String office, int salary, MyDate employedDate,  
 String titleNum) {  
 super(name, address, tel, email, office, salary, employedDate);  
 this.titleNum = titleNum;  
 }  
  
 public String getTitleNum() {  
 return titleNum;  
 }  
  
 public void setTitleNum(String titleNum) {  
 this.titleNum = titleNum;  
 }  
  
 @Override  
 public String toString() {  
 return "Staff{"  
 + "name=" + name  
 + ", address=" + address  
 + ", tel=" + tel  
 + ", email=" + email  
 + ", office=" + office  
 + ", salary=" + salary  
 + ", employedDate=" + employedDate  
 + ", titleNum=" + titleNum + "}";  
 }  
  
}

**Text2.java**

package extendsHomeworks.second;  
  
import java.util.Date;  
  
public class Text2 {  
 public static void main(String[] args) {  
 Person people = new Person("黄飞鸿", "黄飞鸿的家", 1368263868, "1368263868@\*\*\*.com");  
 Employee employee = new Employee("李小龙", "李小龙的家", 169676876, "169676876@\*\*\*.com", "x办公室", 7000, new MyDate());  
 Student student = new Student("成龙", "成龙的家", 1475457757, "1475457757@\*\*\*.com",Student.ONE);  
 Faculty faculty = new Faculty("甄子丹", "甄子丹的家", 5774754475L, "5774754475@\*\*\*.com","y办公室", 3000, new MyDate(new Date()),"早9点~晚6点","校长");  
 Staff staff = new Staff("叶问", "叶问的家", 1368263868, "1368263868@\*\*\*.com","z办公室", 5000, new MyDate(new Date(6716784334467L)),"S686");  
  
 //默认调用toString()方法  
 System.out.println(people);  
 System.out.println(student);  
 System.out.println(employee);  
 System.out.println(faculty);  
 System.out.println(staff);  
  
 }  
}



**11.3**(Subclasses of Account) In Programming Exercise 9.7, the Account class was defined to model a bank account. An account has the properties account number, balance, annual interest rate, and date created, and methods to deposit and withdraw funds. Create two subclasses for checking and saving accounts. A checking account has an overdraft limit, but a savings account cannot be overdrawn.

(账户类 Account 的子类）在编程练习題 9.7 中定义了一个 Account 类来建模一个银行账户。一个账户有账号、余额、年利率、开户日期等属性，以及存款和取款等方法。创建两个支票账户（checking account) 和储蓄账户（saving account)的子类。支票账户有一个透支限定额，但储蓄账户不能透支。

Draw the UML diagram for the classes and then implement them. Write a test program that creates objects of Account, SavingsAccount, and CheckingAccount and invokes their toString()methods.

画出这些类的UML图并实现这些类。编写一个测试程序，创建Account、 SavingsAccount 和 CheckingAccount的对象，然后调用它们的 toString() 方法。

Account

- Id: int

- balance: double

- annualInterestRate: double

- dateCreated: Date

+ Account()

+ Account(id: int, balance: double)

+ setId(id: int): void

+ getId(): int

+ getBalance(): double

+ setBalance(balance: double): void

+ setAnnualInterestRate(annualInterestRate: double): void

+ getDateCreated(): String

+ getMonthlyInterestRate(): double

+ getMonthlyInterest(): double

+ withDraw(money: double): Boolean

+ deposit(money: double): void

+ toString(): String

CheckingAccount

- overdraftLimit: double

+ CheckingAccount ()

+ CheckingAccount (id: int, balance: double, overdraftLimit: double)

+ getOverdraftLimit (): double

+ setOverdraftLimit (overdraftLimit: double): void

SavingsAccount

+ SavingsAccount ()

+ SavingsAccount (id: int, balance: double)

**Account.java**

package extendsHomeworks.third;  
  
import java.text.SimpleDateFormat;  
import java.util.Date;  
  
public class Account {  
 protected int id;  
 protected double balance;// 余额  
 protected double annualInterestRate;// 年利率  
 protected Date dateCreated;// 开户日期  
  
 public Account() {  
 id = 0;  
 balance = 0;  
 dateCreated = new Date();  
 }  
  
 public Account(int id, double balance) {  
 dateCreated = new Date();  
 this.id = id;  
 this.balance = balance;  
 }  
  
 public int getId() {  
 return id;  
 }  
  
 public void setId(int id) {  
 this.id = id;  
 }  
  
 public double getBalance() {  
 return balance;  
 }  
  
 public void setBalance(double balance) {  
 this.balance = balance;  
 }  
  
 public double getAnnualInterestRate() {  
 return annualInterestRate;  
 }  
  
 public void setAnnualInterestRate(double annualInterestRate) {  
 this.annualInterestRate = annualInterestRate;  
 }  
  
 public String getDateCreated() {  
 //格式化日期  
 SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");  
 return sdf.format(dateCreated);  
 }  
  
 //获取月利率  
 public double getMonthlyInterestRate() {  
 return annualInterestRate / 12;  
 }  
  
 //获取月利息  
 public double getMonthlyInterest() {  
 return getMonthlyInterestRate() \* balance;  
 }  
  
 public boolean withDraw(double money) {  
 if (balance > money) {  
 balance -= money;  
 return true;  
 } else  
 return false;  
 }  
  
 public void deposit(double money) {  
 balance += money;  
 }  
  
 @Override  
 public String toString() {  
 return "Account{" +  
 "id=" + id +  
 ", balance=" + balance +  
 ", annualInterestRate=" + annualInterestRate + "%" +  
 ", dateCreated=" + getDateCreated() +  
 '}';  
 }  
}

**CheckingAccount.java**

package extendsHomeworks.third;  
  
public class CheckingAccount extends Account {  
 double overdraftLimit;//透支额  
  
 public CheckingAccount() {  
 super();  
 }  
  
 public CheckingAccount(int id, double balance, double overdraftLimit) {  
 super(id, balance);  
 this.overdraftLimit = overdraftLimit;  
 }  
  
 public double getOverdraft() {  
 return overdraftLimit;  
 }  
  
 public void setOverdraft(double overdraftLimit) {  
 this.overdraftLimit = overdraftLimit;  
 }  
  
 @Override  
 public String toString() {  
 return "CheckingAccount{" +  
 "id=" + id +  
 ", balance=" + balance +  
 ", annualInterestRate=" + annualInterestRate + "%" +  
 ", dateCreated=" + getDateCreated() +  
 ", overdraftLimit=" + overdraftLimit +  
 '}';  
 }  
}

**SavingsAccount.java**

package extendsHomeworks.third;  
  
public class SavingsAccount extends Account {  
 public SavingsAccount() {  
 }  
  
 public SavingsAccount(int id, double balance) {  
 super(id, balance);  
 }  
  
 @Override  
 public String toString() {  
 return "SavingsAccount{" +  
 "id=" + id +  
 ", balance=" + balance +  
 ", annualInterestRate=" + annualInterestRate + "%" +  
 ", dateCreated=" + getDateCreated() +  
 '}';  
 }  
}

**Text3.java**

package extendsHomeworks.third;  
  
public class Text3 {  
 public static void main(String[] args) {  
 Account a = new Account(7328, 10000);  
 CheckingAccount ca = new CheckingAccount(7327, 10000,1500);  
 SavingsAccount sa = new SavingsAccount(73232, 10000);  
  
 //设置年利息  
 a.setAnnualInterestRate(3.5);  
 ca.setAnnualInterestRate(4.0);  
 sa.setAnnualInterestRate(3.0);  
  
 //默认调用toString  
 System.out.println(a);  
 System.out.println(ca);  
 System.out.println(sa);  
 }  
}

