/\*\*

\*

\* @author Zhenhua (Aaron) Yang

\* @version 20180829

\*/

public class Employee {

private String name;

static int id = 0;

static int countEmployee = 0;

//constructors

public Employee(){

name = null;

}

public Employee( String newName ){

name = newName;

id ++;

countEmployee ++;

}

public int getId(){

return id;

}

public String getName(){

return name;

}

public void setName(String newName){

name = newName;

}

@Override

public String toString(){

return getClass().getName() + ": " + id + name;

}

@Override

public boolean equals(Object o){

Employee objE = (Employee) o;

if (!(o instanceof Employee ))

return false;

else {

return id == objE.id && name .equals(objE.name);

}

}

}

/\*\*

\*

\* @author Zhenhua (Aaron) Yang

\* @version 20180829

\*/

public class Salaried extends Employee {

private String title;

private int salary;

private int salariedNum;

//Constructors:

public Salaried(){

super();

title = null;

salary = 0;

salariedNum ++;

}

public Salaried( String newName, String newTitle, int newSalary ){

super( newName );

title = newTitle;

salary = newSalary;

salariedNum ++;

}

//getters and setters

public String getTitle(){

return title;

}

public int getSalary(){

return salary;

}

public int countSalaried(){

return salariedNum;

}

public void setTitle(String newTitle){

title = newTitle;

}

public void setSalary(int newSalary){

salary = newSalary;

}

//the "toString" mathod to print the attributes of a instance.

@Override

public String toString(){

return getClass().getName() + ": " + ( super.getId() + 1 ) + ". " + super.getName() + ": " +

getTitle() + ", $" + getSalary() + "/year.";

}

}

/\*\*

\*

\* @author Zhenhua (Aaron) Yang

\* @version 20180829

\*/

public class Hourly extends Employee {

private String position;

private double hourlyRate;

private int hourlyNum;

//Constructors:

public Hourly(){

super();

position = null;

hourlyRate = 0;

hourlyNum ++;

}

public Hourly ( String newName, String newPosition, double newRate ){

super(newName);

position = newPosition;

hourlyRate = newRate;

hourlyNum ++;

}

//Getter and setters.

public String getPosition(){

return position;

}

public double getRate(){

return hourlyRate;

}

public void setPosition(String newTitle){

position = newTitle;

}

public void setRate(int newRate){

hourlyRate = newRate;

}

public int countHourly(){

return hourlyNum;

}

//the "toString" mathod to print the attributes of a instance.

@Override

public String toString(){

return getClass().getName() + ": " + ( super.getId() + 1 ) + ". " + super.getName() + ": " +

getPosition() + ", $" + getRate() + "/hour.";

}

}

/\*\*

\*

\* @author Zhenhua (Aaron) Yang

\* @version 20180829

\*/

import java.util.Scanner;

public class Client {

public static void main(String[] args){

Employee[] employeeList = new Employee[10]; //Create an array called "employeeList" with the length of 10.

Scanner inputInfo = new Scanner(System.in); //Create an Scanner object "inputInfo".

String name;

String typeOfEmployment;

String title;

String position;

int salary;

double hourlyRate;

for( int i = 0; i < 3; i++){ //Create a loop to ask user enter employee information from keyboard.

Salaried newPerson = new Salaried();

System.out.print("Please enter the employee's name: ");

name = inputInfo.next();

System.out.print("Salaried(please enter \"S\") or Hourly(please enter \"H\")? ");

typeOfEmployment = inputInfo.next() ;

if(typeOfEmployment.equals("S")){ //chack if the employee type is Salaried or HourlyRated.

System.out.print("Please enter the employee's title: ");

title = inputInfo.next();

System.out.print("Please enter the salary of the employee: ");

salary = inputInfo.nextInt();

employeeList[i] = new Salaried(name, title, salary ); // if letter "S" was entered, create an array element of salaried employee.

}else{

System.out.print("Please enter the employee's position: ");

position = inputInfo.next();

System.out.print("Please enter the hourlyRate of the employee: ");

hourlyRate = inputInfo.nextDouble();

employeeList[i] = new Hourly(name, position, hourlyRate ); // if letter "H" was entered, create an array element of hourly rated employee.

}

}

System.out.println("\n");

for ( Object element: employeeList ){ //print out the element of the array "employeeList".

System.out.println(element);

}

}

}

run-single:

===\* Please enter three Salaried employees \*===

Employee ID: 1

Employee name: Amy

Employee Title: Designe

Salary: 98000

Employee ID: 2

Employee name: Aliy

Employee Title: Designer

Salary: 90000

Employee ID: 3

Employee name: Emily

Employee Title: Professore

Salary: 99000

===\* Please enter three Hourly employees \*===

Employee ID: 4

Employee name: Jhon

Employee Position: Architect

Hourly Rate: 75

Employee ID: 5

Employee name: Mike

Employee Position: Maneger

Hourly Rate: 85

Employee ID: 6

Employee name: Jake

Employee Position: Planner

Hourly Rate: 60

Salaried@1:Amy Designe:$98000.0

Salaried@2:Aliy Designer:$90000.0

Salaried@3:Emily Professore:$99000.0

Hourly@4:Jhon Architect:75.0

Hourly@5:Mike Maneger:85.0

Hourly@6:Jake Planner:60.0

null

null

null

null

Salaried@1:Amy Designe:$107800.00000000001

Salaried@2:Aliy Designer:$99000.00000000001

Salaried@3:Emily Professore:$108900.00000000001

Hourly@4:Jhon Architect:82.5

Hourly@5:Mike Maneger:93.50000000000001

Hourly@6:Jake Planner:66.0

BUILD SUCCESSFUL (total time: 2 minutes 1 second)