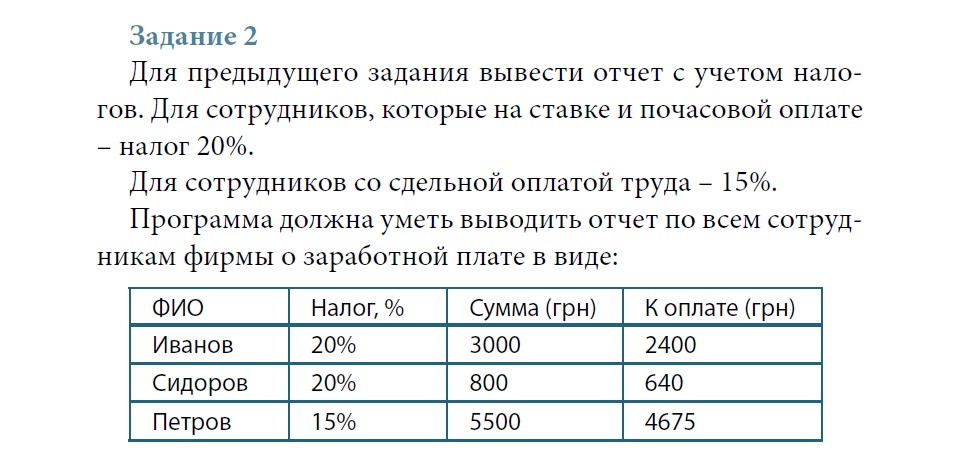
Домашнее задание № 7.2

ТЕМА: НАСЛЕДОВАНИЕ

******

***class Workers2 extends Workers***

package com.company.task2;  
  
import com.company.task1.Workers;  
  
//Класс "Работники" - Workers2 наследует от Workers  
public class Workers2 extends Workers *{* //поля  
 protected int taxPiece = 15; // налог 15%  
 protected int tax = 20; // налог 20%  
  
 // конструкторы  
 // конструктор с 5 параметрами  
 public Workers2*(*String *surname*, String *initials*, String *typeOfPayment*,  
 double *time*, double *payment) {* super*(surname*, *initials*, *typeOfPayment*, *time*, *payment)*;  
 *}* // getters and setters  
 public int getTaxPiece*() {* return taxPiece;  
 *}* public void setTaxPiece*(*int *taxPiece) {* this.taxPiece = *taxPiece*;  
 *}* public int getTax*() {* return tax;  
 *}* public void setTax*(*int *tax) {* this.tax = *tax*;  
 *}  
  
}*

***class WorkersRateOfPayment2 extends Workers2***

package com.company.task2;  
  
// класс WorkersRateOfPayment2 (ставка) - наследник Workers2  
  
public class WorkersRateOfPayment2 extends Workers2 *{* // конструкторы  
 // конструктор с 5 параметрами  
 public WorkersRateOfPayment2*(*String *surname*, String *initials*, String *typeOfPayment*,  
 double *rateDays*, double *daysWorked) {* super*(surname*, *initials*, *typeOfPayment*, *rateDays*, *daysWorked)*;  
 *}* // методы  
 // 1. метод представления класса в виде строки - перегрузка метода toString  
 @Override  
 public String toString*() {* return "|" + surname + " " + initials + "|" + tax + "%|" + salaryCalculation*()* + "руб.|" +  
 taxRate*()* + " руб.|";  
 *}* // 2. метод вычисления заработной платы работника с учётом налога 20%  
 public double taxRate*() {* return *(*salaryCalculation*()* - *(*salaryCalculation*()* \* tax / 100*))*;  
 *}  
}*

***class WorkersHourOfPayment2 extends Workers2***

package com.company.task2;  
  
// класс WorkersHourOfPayment2 (почасовая ставка) - наследник Workers2  
public class WorkersHourOfPayment2 extends Workers2 *{* // конструкторы  
 // конструктор с 5 параметрами  
 public WorkersHourOfPayment2*(*String *surname*, String *initials*, String *typeOfPayment*,  
 double *rateHour*, double *hourWorked) {* super*(surname*, *initials*, *typeOfPayment*, *rateHour*, *hourWorked)*;  
 *}* // getters and setters  
 public int getTax*() {* return tax;  
 *}* public void setTax*(*int *tax) {* this.tax = *tax*;  
 *}* // методы  
 // 1. метод вычисления заработной платы работника с учётом налога 20%  
 public double taxOnHourly*() {* return *(*salaryCalculation*()* - *(*salaryCalculation*()* \* tax / 100*))*;  
 *}* // 2. метод представления класса в виде строки - перегрузка метода toString  
 @Override  
 public String toString*() {* return "|" + surname + " " + initials + "|" + tax + "%|" + salaryCalculation*()* + "руб.|" +  
 taxOnHourly*()* + " руб.|";  
 *}  
}*

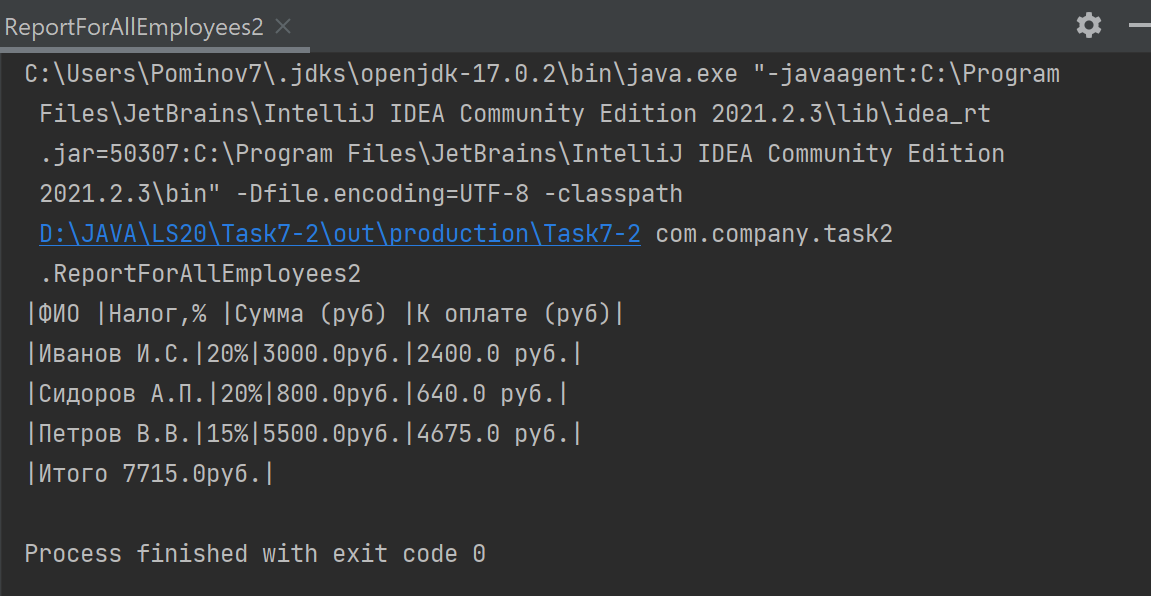
***class WorkersPieceOfPayment2 extends Workers2***

package com.company.task2;  
  
// класс WorkersPieceOfPayment2 (сдельная оплата) - наследник Workers2  
  
public class WorkersPieceOfPayment2 extends Workers2 *{* // конструкторы  
 // конструктор с 5 параметрами  
 public WorkersPieceOfPayment2*(*String *surname*, String *initials*, String *typeOfPayment*,  
 double *piecework*, double *sumWorked) {* super*(surname*, *initials*, *typeOfPayment*, *piecework*, *sumWorked)*;  
 *}* // методы  
 // 1. метод представления класса в виде строки - перегрузка метода toString  
 @Override  
 public String toString*() {* return "|" + surname + " " + initials + "|" + taxPiece + "%|" + salaryCalculation*()* + "руб.|" +  
 taxFix*()* + " руб.|";  
 *}* // 2. метод вычисления заработной платы работника с учётом налога 15%  
 public double taxFix*() {* return *(*salaryCalculation*()* - *(*salaryCalculation*()* \* taxPiece / 100*))*;  
 *}  
}*

***class ReportForAllEmployees2***

package com.company.task2;  
  
public class ReportForAllEmployees2 *{* public static void main*(*String*[] args) {* //Создаём объект работника с видом оплаты труда "Ставка"  
 WorkersRateOfPayment2 *employee1* = new WorkersRateOfPayment2*(*"Иванов", "И.С.", "Ставка",  
 100, 30*)*;  
  
 //Создаём объект работника с видом оплаты труда "Почасовая"  
 WorkersHourOfPayment2 *employee2* = new WorkersHourOfPayment2*(*"Сидоров", "А.П.", "Почасовая", 100, 8*)*;  
  
 //Создаём объект работника с видом оплаты труда "Сдельная"  
 WorkersPieceOfPayment2 *employee3* = new WorkersPieceOfPayment2*(*"Петров", "В.В.", "Сдельная", 550, 10*)*;  
  
 //вывод в консоль отчета по всем сотрудникам фирмы о заработной плате\  
 System.*out*.println*(*"|ФИО " + "|Налог,% " + "|Сумма (руб) |" + "К оплате (руб)|"*)*;  
 System.*out*.println*(employee1)*;  
 System.*out*.println*(employee2)*;  
 System.*out*.println*(employee3)*;  
 System.*out*.println*(*"|Итого " + *(employee1*.taxRate*()* + *employee2*.taxOnHourly*()* +  
 *employee3*.taxFix*())* + "руб.|"*)*;  
 *}  
}*

***Тест:***

******