Lab 2: Employee Management – UML

Package: employee

<<Java Class>>
©ᴬEmployee
employee

- ˢₒᶠ GENDER_MAN: String
- ˢₒᶠ GENDER_WOMAN: String
- ˢₒᶠ GENDER_UNDEFINED: String
- ▫ name: String
- ▫ birthday: String
- ▫ salary: int
- ▫ gender: String
- ▫ᶠ employeeId: int
- ▵ˢ idGenerator: int

- ◇ᶜ Employee(Employee)
- ◇ᶜ Employee(String,String,int,String)
- ●ᴬ *calculateBonus():int*
- ● toString():String
- ● getName():String
- ● setName(String):void
- ● getBirthday():String
- ● setBirthday(String):void
- ● getSalary():int
- ● setSalary(int):void
- ● getGender():String
- ● setGender(String):void
- ● getEmployeeId():int

<<Java Enumeration>>
●Profession
employee

- ˢₒᶠ INVALID: Profession
- ˢₒᶠ SECRETARY: Profession
- ˢₒᶠ TECHNICIAN: Profession
- ˢₒᶠ PROGRAMMER: Profession

- ●ᶜ Profession()

<<Java Class>>
©Programmer
employee

- ˢₒᶠ BONUS_MULTIPLIER: double

- ●ᶜ Programmer(Employee)
- ●ᶜ Programmer(String,String,int,String)
- ● calculateBonus():int

<<Java Class>>
©Secretary
employee

- ˢₒᶠ BONUS_MULTIPLIER: double

- ●ᶜ Secretary(Employee)
- ●ᶜ Secretary(String,String,int,String)
- ● calculateBonus():int

<<Java Class>>
©Technician
employee

- ˢₒᶠ BONUS_MULTIPLIER: double

- ●ᶜ Technician(Employee)
- ●ᶜ Technician(String,String,int,String)
- ● calculateBonus():int

Package: employee.management

EmployeeManager

<<Java Class>>
EmployeeManager
employee.management

- employees: ArrayList<Employee>

- EmployeeManager()
- verifyBirthdayInput(String):boolean
- add(String,String,Profession,int,String):boolean
- getIndexOfEmployee(int):int
- delete(int):boolean
- update(int,String,String,Profession,Integer,String):boolean
- displayEmployee(Employee):void
- search(Integer,String,Profession):void
- displayAll():void
- toString():String
- getEmployees():ArrayList<Employee>

EmployeeStatistics

<<Java Class>>
EmployeeStatistics
employee.management

- employees: ArrayList<Employee>

- EmployeeStatistics(EmployeeManager)
- displayAverageSalary():void
- displayMaxSalary():void
- displayMinSalary():void
- displayTotalBonus():void
- displayWomenStaffPercentage():void
- displayMenPercentageOfWorkRole():void

Package: userio & userio.menu

MainMenu

<<Java Class>>
MainMenu
userio.menu

- F employeeManager: EmployeeManager
- S F INVALID_MENU_CHOICE: String
- S F OPT_EMPLOYEE_MANAGEMENT: int
- S F OPT_EMPLOYEE_STATISTICS: int
- S F OPT_EXIT: int

- C MainMenu()
- showMainMenu():boolean
- getEmployeeManager():EmployeeManager

-statisticsMenu 0..1

-managementMenu 0..1

StatisticsMenu

<<Java Class>>
StatisticsMenu
userio.menu

- F employeeStatistics: EmployeeStatistics
- S F OPT_AVERAGE_SALARY: int
- S F OPT_MAX_SALARY: int
- S F OPT_MIN_SALARY: int
- S F OPT_TOTAL_BONUS: int
- S F OPT_WOMEN_PERCENTAGE: int
- S F OPT_MEN_PERCENTAGE_OF_WORK_ROLE: int
- S F OPT_EXIT: int

- C StatisticsMenu(EmployeeManager,SystemInput)
- showStatisticsMenu():boolean

ManagementMenu

<<Java Class>>
ManagementMenu
userio.menu

- F employeeManager: EmployeeManager
- S F OPT_REGISTER: int
- S F OPT_DELETE: int
- S F OPT_UPDATE_NAME: int
- S F OPT_UPDATE_BIRTHDAY: int
- S F OPT_UPDATE_PROFESSION: int
- S F OPT_UPDATE_SALARY: int
- S F OPT_SEARCH_BY_NAME: int
- S F OPT_SEARCH_BY_ID_NUMBER: int
- S F OPT_SEARCH_BY_PROFESSION: int
- S F OPT_DISPLAY_ALL: int
- S F OPT_EXIT: int

- C ManagementMenu(EmployeeManager,SystemInput)
- showManagementMenu():boolean
- deleteEmployee():void
- getProfession():Profession
- searchEmployee(int):void
- updateEmployee(int):void
- registerEmployee():void

-systemInput 0..1

-systemInput 0..1

-systemInput 0..1

SystemInput

<<Java Class>>
SystemInput
userio

- F sc: Scanner

- C SystemInput()
- handleBadInput():void
- getDouble():double
- getInt():int
- getString():String