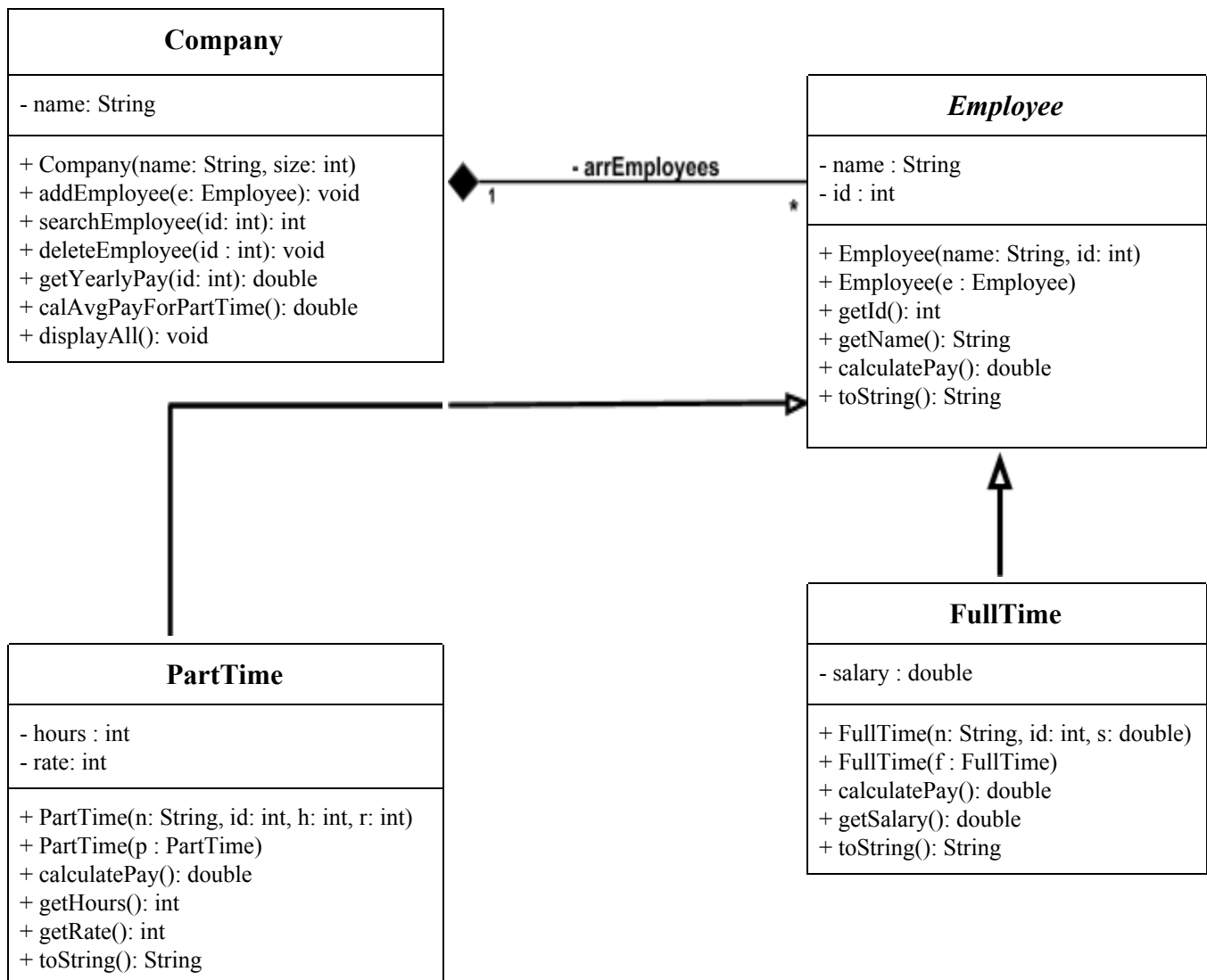


King Saud University
College of Computer & Information Science
CSC113 – Computer Programming II
Lab08: Exception Handling

Submission rules:

- The project name must be: Lab08_ID_FirstName_LastName.zip. For example: Lab08_123456789_Marwan_Almaymoni.zip
- Use the default package.
- The due date is Wednesday 18/11/2020 11:59 PM via lms.ksu.edu.sa
- Email submissions will not be accepted.



Abstract Class *Employee*:

- Attributes:
 - **name**: name of the employee.
 - **id**: employee Id number.
- Methods:
 - + Employee(name: String, id: int): *throws an exception (IllegalArgumentException)* if the passed id is less than 1.
 - + Employee(e : Employee)
 - + calculatePay()
 - + getName()
 - + getId()
 - + toString(): returns the employee information.

<ID>: <Name>

Class *PartTime*:

- Attributes:
 - **hours**: number of weekly hours the employee works.
 - **rate**: pay rate for each hour.
- Methods:
 - + PartTime(name: String, id: int, hours: int, rate: int): *throws an exception (IllegalArgumentException)* if the passed hours or rate is less than 0.
 - + PartTime(p : PartTime)
 - + calculatePay(): calculate and return the monthly pay for the part time employee as follows:
 - Monthly pay = number of hours each week * 4 * rate
 - + getRate()
 - + getHours()
 - + toString(): returns the part-time employee information

<ID>: <Name>, Weekly Hours: <Hours>, Pay Rate: <Rate>, Pay: <calculatePay()>

Class *FullTime*:

- Attributes:
 - **salary**: Employee monthly salary
- Methods:

- + FullTime(name: String, id: int, salary: double): ***throws an exception (IllegalArgumentException)*** if the passed salary is less than 0.
- + FullTime(f : FullTime)
- + calculatePay() calculate and return the monthly pay for the part time employee as follows:
 - Monthly pay = the salary after 9% deduction for social insurance.
- + getSalary()
- + toString(): returns the full-time employee information

<ID>: <Name>, Salary: <Salary>, Pay: <calculatePay()>

Class *Company*:

- Attributes:
 - **name**: company name
- Methods:
 - + Company(name: String, size: int): ***throws an exception*** when the array is created with a negative size. (**NegativeArraySizeException**)
 - + addEmployee(e: Employee): add new employee to the company. The method ***throws an exception (IllegalStateException)*** when trying to add while the array is full. It uses the method (searchEmployee(id: int)) to check if the id exists in the company, if it exists, it ***throws an exception (IllegalArgumentException)***.
 - + searchEmployee(id: int): search for the employee in the company and return the index, if the id doesn't exist return -1
 - + deleteEmployee(id: int): uses the method (searchEmployee(id: int)), then deletes the employee in the returned index. The method ***throws an exception (IndexOutOfBoundsException)*** when trying to delete non-existent employee.
 - + getYearlyPay(id: int): gets yearly pay for the employee. The method ***throws an exception (IndexOutOfBoundsException)*** if the employee doesn't exist.
 - + calAvgPayForPartTime(): calculates the average pay for all the part-time employees. The method ***throws an exception (ArithmeticException)*** when there are no part-time employees in the company (divide by zero).
 - + displayAll(): display all the employees information using toString().

Implement the previous classes, in addition to a main method to test the classes.