Class Department:

Attributes:

- id: integer

- name: string

- teachersMap: map of teachers (key: integer, value: Teacher)

- staffMap: map of staff members (key: integer, value: Staff)

- dean: Teacher (head of department)

Methods:

- Constructor(id: integer, name: string):

Initializes the id and name attributes with the provided values.

Initializes teachersMap and staffMap as empty maps.

Adds this department to the department manager.

- addTeacherMap(teacher: Teacher)

Adds a teacher to the map of teachers in the department.

Sets the department id and name for the teacher.

- addStaffMap(staff: Staff)

Adds a staff member to the map of staff members in the department.

Sets the department id and name for the staff member.

- getDean(): Teacher

Returns the head of department (dean).

- setDean(dean: Teacher)

Sets the head of department (dean) for the department.

Verifies if the teacher belongs to the department before assigning as dean.

Throws an InvalidDeanException if the teacher does not belong to the department.

Class DepartmentManager:

Attributes:

- departmentsMap: HashMap (key: integer, value: Department)

Methods:

- addDepartment(department: Department)

Adds a Department object to the departments map using the department's ID as the key.

- findDepartmentById(departmentId: integer): Department

Searches for a department by its ID in the departments map.

Returns the department if found, otherwise returns null.

- writeToTextFile(filename: string)

Writes the department and its teachers/staff members to a text file.

Formats the department and its members as strings and writes them to the file.

Throws an IOException if writing to the file fails.

Class FullTimeTeacher extends Teacher:

Constructor:

- Parameters: specialty, degree, name, birthyear, id, gender

- Calls the super constructor of the Teacher class with the following parameters:

- specialty, degree, hoursWorked = 32, coursesQuantity = 4, name, birthyear, id, gender

Methods:

- computePayroll():

- Overrides the computePayroll method from the Teacher class.

- Calculates the payroll for a full-time teacher.

- Returns the computed payroll.

Class PartTimeTeacher extends Teacher:

Constructor:

- Parameters: specialty, degree, hoursWorked, name, birthyear, id, gender

- Calls the super constructor of the Teacher class with the following

parameters:

- specialty, degree, hoursWorked, coursesQuantity = 2, name, birthyear, id, gender

Methods:

- computePayroll():

- Overrides the computePayroll method from the Teacher class.

- Calculates the payroll for a part-time teacher.

- Returns the computed payroll.

Interface PayRoll:

Method:

- computePayroll()

- Description: This method computes the payroll for a staff member or teacher.

- Returns: The computed payroll as a double value.

Abstract Class Person:

Fields:

- name: String

- birthyear: int

- id: int

- departmentId: int

- gender: String

- category: String

- department: String

Constructor:

- Person(name: String, birthyear: int, id: int, gender: String, category: String):

- Initializes the name, birthyear, id, gender, and category fields with the provided values.

Methods:

- setCategory(): abstract

- Description: Abstract method to set the category of the person.

Class Staff:

Fields:

- duty: String

- workLoad: int

Constructor:

- Staff(duty: String, workLoad: int, name: String, birthyear: int, id: int, gender: String):

- Initializes the name, birthyear, id, gender, category, duty, and workLoad fields with the provided values.

Methods:

- setCategory(): override

- Description: Sets the category of the staff member to "Staff".

- computePayroll(): override

- Description: Computes the payroll for the staff member based on their work load.

- Returns: double representing the computed payroll.

- readFromTextFile(filename: String): static

- Description: Reads staff data from a text file and adds staff members to departments.

- Parameters: filename - String representing the name of the text file to read from.

- Throws: FileNotFoundException if the file is not found or NumberFormatException if there's an error parsing the file.

Class Teacher:

Fields:

- specialty: String

- degree: String

- degreeRate: int

- hoursWorked: int

- coursesQuantityWeekly: int

Constructor:

- Teacher(specialty: String, degree: String, hoursWorked: int, coursesQuantityWeekly: int, name: String, birthyear: int, id: int, gender: String):

- Initializes the name, birthyear, id, gender, category, specialty, degree, hoursWorked, and coursesQuantityWeekly fields with the provided values. It also calculates the degreeRate based on the degree.

Methods:

- setCategory(): override

- Description: Sets the category of the teacher to "Teacher".

- computePayroll(): override

- Description: Computes the payroll for the teacher based on their hours worked, degree rate, and courses quantity weekly.

- Returns: double representing the computed payroll.

- readFromTextFile(filename: String): static

- Description: Reads teacher data from a text file and adds teachers to departments.

- Parameters: filename - String representing the name of the text file to read from.

- Throws: FileNotFoundException if the file is not found or NumberFormatException if there's an error parsing the file.