Documentation for the assignment of Learning By Doing II

1. Defined classes for the project:

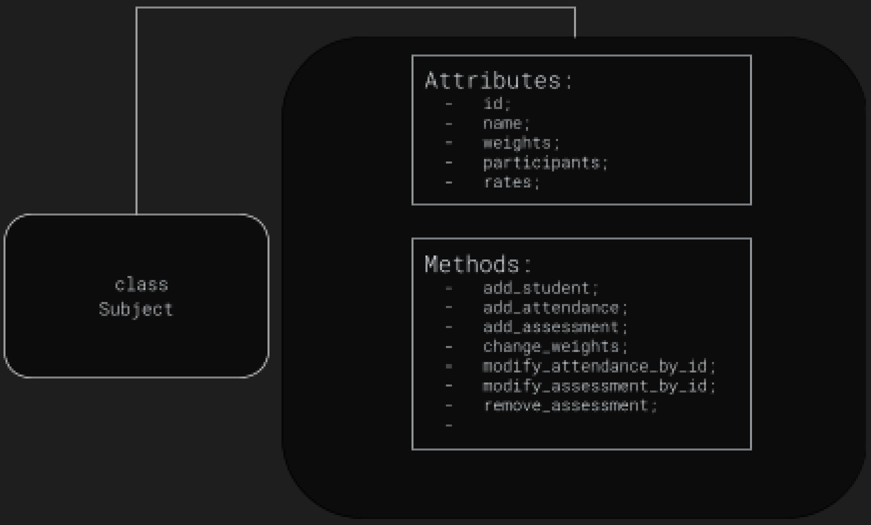
```
class Errormessages:
        "”" Class that holds error messages for every input scenario """


class Attendance:
    # Attributes of the Attendance class
    total_minutes: int
    late_minutes: int
    day: int
    id: int
    status: str

    # Methods of the Attendance class
    def get_attendance_status() -> str:
        """ Converts the late minutes to status using the dic_attend dict and
returns the later """


class Assessment:
    # Attributes of the Assessment class
    assessment_type: str
    grade: int
    id: int
    weight: float


class Student:
    # Attributes of the Student class
    name: str
    gender: str
    id: int
    group: str
    subjects: list[Subject]
    attendance: dict[subject_id: int, list[Attendance]]
    assessments: dict[subject_id: int, list[Assessment]]
    # Methods of the Student class
    def print_info() -> None:
        """ Displays the records of every student, returns None """
    def add_subject(subject: Subject) -> None:
        """ Internal to Subject Class. Adds Subject object to subjects list
"""
    def final_grade() -> float:
        "”" Function that calculates the final grade of a Student """


class Subject:
    # Attributes of the Subject class
    id: int
    name: str
    weights: dict[assessment_type: str, float]
    participants: list[Student]


    # Methods of the Subject class
    def add_student(student: Student) -> None:
        """ Adds Student object to participants list AND calls add_subject
"""

    def add_attendance(total_minutes: int, late_minutes: list[int], day:
str) -> None:
        """ Adds an Attendance object to each student in participants list
"""

    def add_assessment(assessment_type: str, grades: list[int]) -> None:
        """ Adds an Assessment object to each student in participants list
"""

    def change_weights(assessment_type: str, new_weight: int) -> None:
        """ Changes the weight of the given assessment type to new_weight """

    def modify_assessement_by_id(student: Union[Student, student_id: int],
assessment_id: int, new_grade: int) -> None:
        """ Modifies the assessment's grade of a student given the student's
id\student object and the assessment's id """

    def remove_assessment(assessment_id: int) -> None:
        """ Removes the specified assessment from the assessments dict of
every student """

    def modify_attendance_by_id(student: Union[Student, student_id: int],
attendance_id: int, new_late_minutes: int) -> None:
        """ Modifies the attendance of student using the id, changes
late_minutes to new_late_minutes """
```

2. Other useful functions:

```
def get_id() -> int:
    """ Returns an id unique to global runtime """
```

3. A general idea of the project:

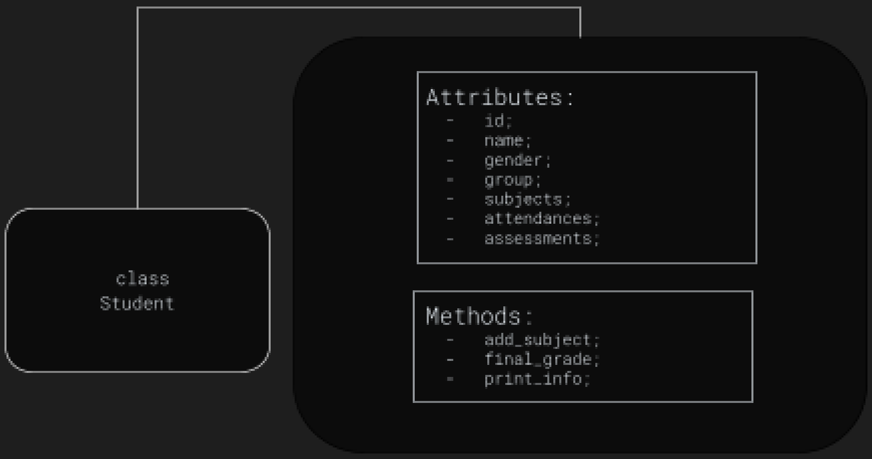
- Visualization of the code’s main components



class Subject

Attributes:
- id;
- name;
- weights;
- participants;
- rates;

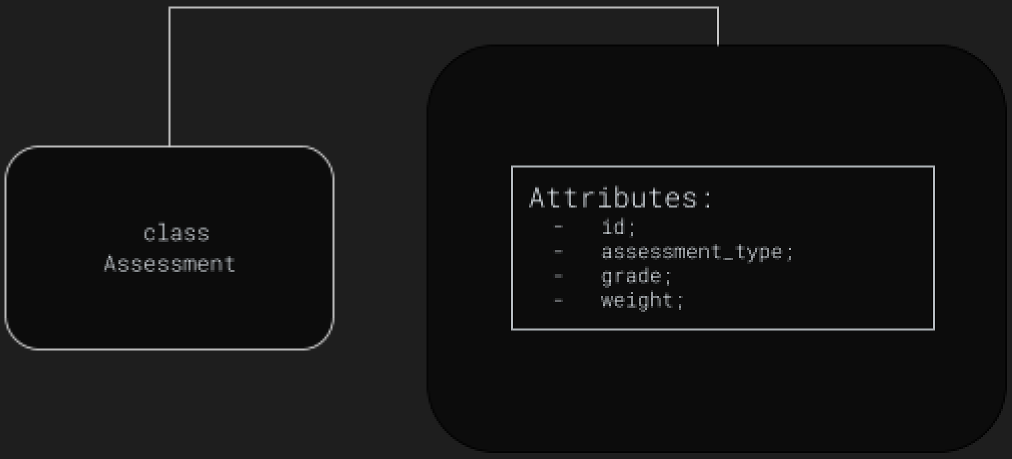
Methods:
- add_student;
- add_attendance;
- add_assessment;
- change_weights;
- modify_attendance_by_id;
- modify_assessment_by_id;
- remove_assessment;
-



class Student

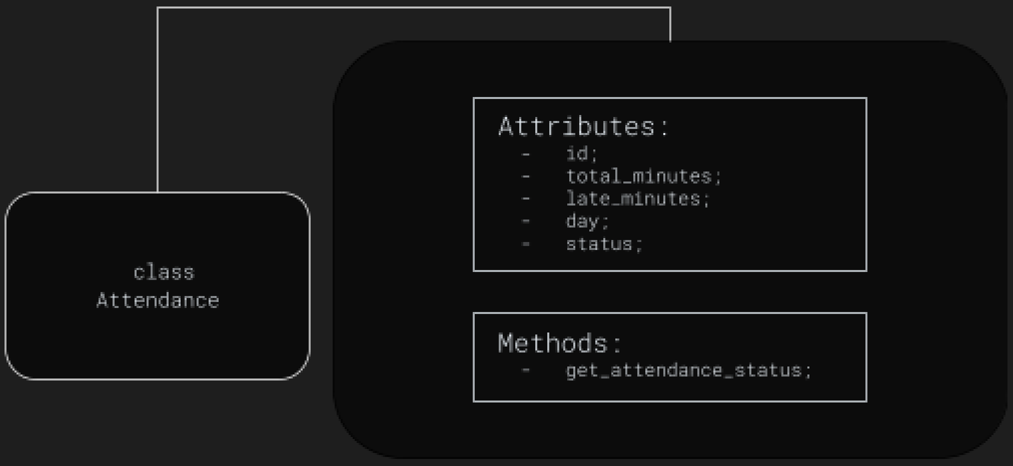
Attributes:
- id;
- name;
- gender;
- group;
- subjects;
- attendances;
- assessments;

Methods:
- add_subject;
- final_grade;
- print_info;



class Assessment

Attributes:
- id;
- assessment_type;
- grade;
- weight;



class Attendance

Attributes:
- id;
- total_minutes;
- late_minutes;
- day;
- status;

Methods:
- get_attendance_status;

Done by:
Mohammed Amine SALIMI (M0-Amine),
Othmane AZOUBI (elEsquina),
Mounia BADDOU (MTheCreator),
For the Learning By Doing II project.

Mohammed VI Polytechnic University,
School Of Computer Science, 2022 / 2023