
software
techie_kit

team_4

{

fady_ibrahem ,

ahmed_hossam ,

mahammed_abdelfattah ,

khaled_abubakr ,

}

mahammed_abdelfattah(Department)

khaled_abubakr(Student)

ahmed_hossam(Professor)

fady_ibrahem(Course)

Orange Digital Center

TECHIE KIT SOFTWARE

CREATIVA
مراكز ابداع مصر الرقمية

mahammed_abdelfattah(Department)

{

This class is a blueprint for a department. It has three private attributes: capacity, gpa_required, and name. It has three public constructors: a default constructor, a parameterized constructor, and a destructor. It also has three public setters and three public getters.

}

khaled_abubakr(Student)

{

This class is a blueprint for a student. It inherits from the `Person` class, so it has all of the attributes and methods of the `Person` class. It also has two private attributes: gpa and level. It has two public constructors: a default constructor and a parameterized constructor. It also has two public setters and two public getters for the gpa and level attributes. It has two public methods: study() and student_take_exam(). It also overrides the attend() method from the `Person` class

}

ahmed_hossam(Professor)

{

The Professor class is a subclass of the Person class. It contains the following private attributes: salary, department, and degree. The class also contains a number of public methods, including constructors, setters, getters, and an override of the attend() method.

}

fady_ibrahem(Course)

{

The Course class is a data structure that represents a course offered at a university. It contains the following private attributes: code, hours, name, professor, students, and prerequisites. The class also contains a number of public methods, including constructors, setters, getters, and a destructor.

}

all_of_team (Person)

{

The Person class is a super class to the Student class and Professor class. It contains the following protected attributes: id, age, name, and gender. The class also contains a number of public methods, including constructors, setters, getters, and a virtual method attend().

}

all_of_team(University)

{

The class "University" represents a mini university and provides functionality to manage and manipulate data related to departments, students, professors, and courses.

the University class represents a mini university and utilizes the Singleton design pattern to ensure that there is only one instance of the class throughout the program.

}