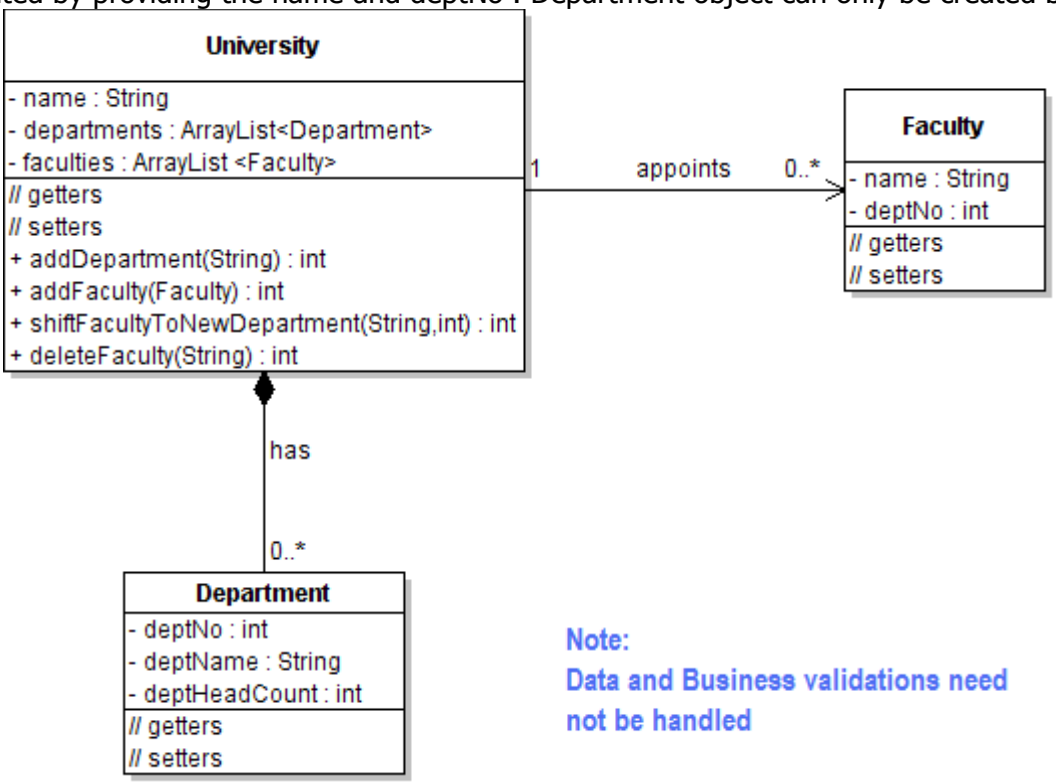
A university wants to automate the functioning of departments and faculties.

Define the classes as shown in the below class diagram, specifying all the getters and setters for each class.
University object can only be created by providing the university name. Faculty object can only be created by providing the name and deptNo . Department object can only be created by giving name.



University class (public class) has following behaviors

1) **addDepartment**: This behavior accepts a Department name and create a department object and appends that to the list of departments in the university and returns the total count of departments.

2) **addFaculty** : This behavior accepts a Faculty object as input. If department No of that Faculty is invalid (i.e. no such department exists), then it returns -1. If department No is valid, then this behavior adds a faculty to the list of Faculty in university, updates the headCount of the corresponding department and returns the headCount.

3) **shiftFacultyToNewDeparment** : This behavior accepts faculty name and new department no to which we want to associate this faculty. This behavior returns -1 if no faculty exists with given name. It returns -2 if no dept exists for given department no. If both faculty name and department number are valid it should update the deptNo for that faculty. It should also increase the headCount for new Department to which the faculty is associated and decrease the headCount for the department to which faculty belonged earlier. It returns the updated headCount of the new department to which faculty is associated.

4) **deleteFaculty** : This behavior accepts a faculty name. If the given faculty name is invalid (means no faculty with that name exists) then is returns -1. If the given name is valid then it deletes the faculty from the list of faculty and also decreases the headCount for the department with which the faculty was associated. It returns the new headCount of the department.