

Домашка

Создать систему для управления студентами в учебном заведении, используя принципы инкапсуляции, перегрузки методов и статические члены.

Обязательные требования для всех классов:

- Инкапсуляция: все поля должны быть `private`
- Геттеры и сеттеры с валидацией данных
- Статические переменные (счетчики объектов)
- `Final` константы
- Статические методы
- Перегруженные методы (минимум 2 перегрузки в каждом классе)
- Три конструктора (по умолчанию, с основными параметрами, со всеми параметрами)

Класс 1: "Student"

Поля:

- `studentId (String)` - номер студенческого билета
- `firstName (String)` - имя
- `lastName (String)` - фамилия
- `age (int)` - возраст
- `averageGrade (double)` - средний балл
- `faculty (String)` - факультет

Задачи:

1. Инкапсуляция: сделать все поля приватными
2. Валидация в сеттерах:
 - `age` от 16 до 70
 - `averageGrade` от 0.0 до 5.0
 - `studentId` не может быть `null` или пустым
3. Статические члены:
 - `totalStudents` - счетчик созданных студентов

- UNIVERSITY_NAME - final константа с названием вуза
 - getTotalStudents() - статический метод
4. Перегруженные методы:
- displayInfo() и displayInfo(boolean detailed)
 - updateGrade(double newGrade) и updateGrade(double newGrade, String subject)
-

Класс 2: "Course" (Учебный курс)

Поля:

- courseCode (String) - код курса
- courseName (String) - название курса
- maxStudents (int) - максимальное количество студентов
- currentStudents (int) - текущее количество записанных
- isActive (boolean) - активен ли курс

Задачи:

1. Инкапсуляция всех полей
 2. Валидация:
 - currentStudents не может превышать maxStudents
 3. Статические члены:
 - totalCourses - счетчик курсов
 - MAX_CREDITS_PER_SEMESTER - final константа
 - calculateWorkload(int credits) - статический метод расчета нагрузки
 4. Перегруженные методы:
 - enrollStudent() и enrollStudent(String studentId)
 - displayInfo() и displayInfo(boolean showAvailability)
-

Класс 3: "Department" (Кафедра)

Поля:

- departmentName (String) - название кафедры

- headOfDepartment (String) - заведующий кафедрой
- officeRoom (String) - номер кабинета
- email (String) - email кафедры
- phoneNumber (String) - телефон
- budget (double) - бюджет кафедры

Задачи:

1. Инкапсуляция всех полей
2. Валидация:
 - email
 - budget не может быть отрицательным
 - phoneNumber должен содержать только цифры
3. Статические члены:
 - totalDepartments - счетчик кафедр
 - UNIVERSITY_BUDGET - final константа
 - validateEmail(String email) - статический метод проверки email
4. Перегруженные методы:
 - contact() и contact(String message)
 - allocateBudget(double amount) и allocateBudget(double amount, String purpose)

Класс 4: "StudentManager" (Менеджер студентов)

Создать класс, который будет управлять массивами студентов и курсов, с методами:

- addStudentToCourse(Student student, Course course)
- findStudentById(String studentId)
- getStudentsByFaculty(String faculty)
- calculateFacultyAverage(String faculty)