Лабораторная работа №4.

1. Создайте класс Vector для представления вектора в трехмерном пространстве (свойства для координат x, y, z). Добавьте в прототип Vector два метода plus() и scalar() для вычисления суммы двух векторов и скалярного произведения двух векторов. Добавьте в прототип свойство только для чтения length, подсчитывающее длину вектора. Переопределите в классе Vector методы toString() и valueOf(). Протестируйте созданный класс.

2. Задача имеет название, описание, дату начала и дату окончания. Любая задача может иметь набор дочерних подзадач. Создайте класс для представления задачи. Выполняемая задача – наследник задачи с дополнительными свойствами: - процент выполнения (число); - флаг задача завершена.

3. Реализовать базовый класс Point, который характеризуется положением на осях координат (свойства для координат x, y, z) и условным обозначением (имя объекта-точки). Определите класс-наследник ColorPoint, который наследует свойства родительского класса и имеет дополнительную характеристику (с дополнительными полем "цвет" ). Создать коллекцию разработанных объектов и создать методы класса для доступа к значениям его полей. Написать функцию, которая осуществляет поиск по набору точек и ищет ближайюшую для заданных координат.

4. Реализовать базовый класс Фигура, который характеризуется набором точек (x,y). Создать классы наследники (Квадрат, Круг, Треугольник). Для классов наследников реализовать возможность вычисления площади и периметра, методы доступа к расчётным значениям.

5. Реализуйте класс Работник, который будет иметь следующие свойства: имя, фамилия, ставка за день работы, количество отработанных дней. Также класс должен иметь метод getSalary(), который будет выводить зарплату работника. Создайте класс наследник для работников с определённой специализацией (маляр, штукатур), в котором добавьте поле “тип выполняемых работ”. Реализуйте возможность формирования бригады с работниками различной специализации.

6. Реализуйте родительский класс Validator с произвольными полями и методами доступа к ним. Создайте классы-наследники для валидации телефонных номеров, email-адресов и доменов сайтов. Сформируйте тестовый набор данных для проверки с помощью реализованных классов-валидаторов. Для проверки значений допускается использование регулярных выражений.

7. Реализуйте класс Библиотека, в котором будет храниться коллекция объектов класса Издание. Создайте классы-наследники для различных типов изданий с соответствующими полями (имя, год издания, тип, автор(ы) и т.д.). Предусмотреть возможность поиска издания по нескольким параметрам и их сортировки по автору.

8. Реализуйте класс Факультет, который хранит список студентов. Класс Студент содержит информацию о студенте. Создайте классы-наследники для студентов отдельных специальностей. В программе предусмотреть возможность вывода всех студентов заданной специальности и поиска студента по фамилии.

9. Реализуйте класс Автопарк, который содержит коллекцию автомобилей организации. Класс Автомобиль должен содержать следующие поля: марка, год выпуска, дата последнего обслуживания, регистрационный номер. Создать классы-наследники для транспортных средств различной категории (легковые, грузовый и т.д.) Предусмотреть возможность вывода автомобилей соответствующей категории и поиска автомобиля по номеру.

10. Разработать класс Пользователь. Реализовать иерархию классов-наследников согласно типам пользователей в системе организации (класс Информационная система). Предусмотреть вывод пользователей определённой группы, поиск пользователя. Реализовать возможность блокировки, добавления и удаления пользователя из системы.

11. Создать класс Тур. Организовать иерархию наследования туров согласно их типам (велотуры, автотуры, гастрономические туры и т.д.). Предусмотреть возможность подбора тура заданного типа на конкретную дату. При выводе списка туров производить группировку по типам.

При выполнении лабораторной использовать классы.