

Лабораторная работа 11. Классы, наследование.

Задание 1. Иерархия классов «Наемные работники»

1. Использовать класс Person из лабораторной работы №10.

Он соответствует личным данным человека и содержит поля:

- фамилия (String),
- имя (String),
- дата рождения (три целых или LocalDate),
- пол (один символ char)

Методы, необходимые для правильной работы с объектами такого класса:

- геттеры, сеттеры, причем для поля **пол** в сеттере проверять корректность устанавливаемых значений
- конструкторы
- методы ввода-вывода,
- метод, возвращающий возраст человека – количество полных лет.

На основе класса Person создать производный класс «Наемный работник с фиксированным окладом».

Этот класс должен содержать дополнительно поля:

- оклад (double),
- %премии (double)

Добавьте или переопределите необходимые методы (геттеры, сеттеры, конструкторы, методы ввода-вывода).

Добавьте методы для расчета

- общей суммы заработной платы с учетом премии
- суммы подоходного налога 13% от оклада с премией
- суммы, выдаваемой работнику, за вычетом налога

Составьте программу для тестирования этого класса.

а) В программе создать двух работников. Для первого объекта указать в аргументах конструктора свои данные. Данные второго работника ввести с клавиатуры.

Вывести для каждого работника суммы его зарплаты, уплаченные им налоги и суммы, которые они получили на руки.

б) В программе ввести данные для пяти работников в массив. Вывести общую сумму уплаченного налога и среднюю зарплату в этой команде.

Вывести зарплату самого младшего работника

2. На основе класса «наемный работник с фиксированным окладом» создать производный класс «работник с почасовой оплатой».

Добавьте и/или переопределите все необходимые элементы такого класса самым минимальным образом:

- добавить в этот класс только одно поле – количество отработанных часов
- поле базового класса, понимаемое ранее как оклад, в новом классе должно хранить ставку почасовой оплаты
- переопределите только один метод из расчетных для зарплаты с премией
также можно переопределить методы ввода-вывода
- новые методы добавлять не надо

Составьте программу для тестирования созданного Вами класса

Задание 2. Иерархия классов Геометрические фигуры

1. Создать класс «точка на плоскости» (можно использовать реализацию из лабораторной работы 8)

Класс должен включать

- поля для хранения координат точки на плоскости
 - методы доступа к полям (геттеры, сеттеры)
 - конструкторы
 - методы консольного ввода и вывода
 - метод toString()
 - метод расчета расстояния между двумя такими точками.
-

2. Определить класс «цилиндр».

Считать, что окружность в основании цилиндра лежит на координатной плоскости XOY, известны координаты ее центра и какой-то (любой) точки на окружности.

Кроме того известна высота цилиндра

Класс должен включать поля (**protected**):

- точка на плоскости - центр окружности в основании цилиндра
- любая точка на окружности в основании цилиндра
- высота цилиндра (длина боковой поверхности)

Определите все необходимые методы (геттеры, сеттеры, конструкторы, методы ввода и вывода).

Добавить методы, рассчитывающие

- площадь основания
- длину окружности в основании
- объем цилиндра
- площадь боковой поверхности

Написать программу, проверяющую корректность всех определенных методов.

3. На основе класса «цилиндр» определить класс «прямая треугольная призма»,

Класс должен включать поля

- три точки основания (три точки, с координатами (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , две из них унаследованы от цилиндра, поэтому добавить надо только описание третьей точки)
- длина боковой стороны (унаследовано от цилиндра, добавлять не надо)

Определить (переопределить) все необходимые методы.

В секцию **protected** добавить вспомогательную функцию, рассчитывающую площадь треугольника по его сторонам.

Методы, рассчитывающие площадь основания и периметр основания должны быть переопределены

Методы расчета объема и площади боковой поверхности должны корректно работать без переопределения.

Написать программу, проверяющую корректность всех определенных методов.

4. На основе класса «прямая треугольная призма» определить класс «прямая призма с четырехугольным основанием», добавив координаты еще одной точки основания

Добавить/переопределить все необходимые поля и методы.

Добавить методы, определяющие

- является ли такая призма параллелепипедом
- является ли призма прямоугольными параллелепипедом
- является ли призма кубом

Написать программу, проверяющую корректность всех определенных методов.