

Практическое задание 3

Объектно-ориентированное программирование в Java

Цель работы: сформировать навыки работы с основными конструкциями ООП языка Java. Освоить принципы применения базовых объектных технологий java.

Задание. Напишите программу на языке Java, в которой продемонстрируйте работу основных принципов ООП.

Рекомендации по выполнению задания

1. Разработайте класс для решения квадратных уравнений. Вычисление дискриминанта должен осуществлять вложенный класс. После компиляции объясните структуру class файлов. Проанализируйте использование вложенного класса.
2. Реализуйте игру в кости. Играют N игроков (компьютер в списке последний). Подкидываются одновременно K кубиков. Выигрывает тот, у кого большая сумма очков. Кто выиграл, тот и кидает первым в следующем кону. Игра идет до 7 выигрышей. Начинаете игру вы.
3. Напишите программу «Адрес человека». Есть сущность Человек, которая связана с сущностью Адрес. Считается, что у каждого человека есть только один адрес. Организовать массив объектов Человек (не менее 4) и по массиву:
 - осуществить поиск Человека по фамилии;
 - осуществить поиск человека по атрибуту адреса;
 - вывести людей, родившихся между определенными датами;
 - найти самого старого (молодого);
 - найти людей, проживающих на одной улице.

4. Класс Phone.

- а) Создайте класс Phone, который содержит переменные number, model и weight.

- б) Создайте три экземпляра этого класса.
- в) Выведите на консоль значения их переменных.
- г) Добавить в класс Phone методы: `receiveCall`, имеет один параметр – имя звонящего. Выводит на консоль сообщение “Звонит {name}”. `getNumber` – возвращает номер телефона. Вызвать эти методы для каждого из объектов.
- д) Добавить конструктор в класс Phone, который принимает на вход три параметра для инициализации переменных класса - `number`, `model` и `weight`.
- е) Добавить конструктор, который принимает на вход два параметра для инициализации переменных класса - `number`, `model`.
- ж) Добавить конструктор без параметров.
- з) Вызвать из конструктора с тремя параметрами конструктор с двумя.
- и) Добавьте перегруженный метод `receiveCall`, который принимает два параметра - имя звонящего и номер телефона звонящего. Вызвать этот метод.
- к) Создать метод `sendMessage` аргументами переменной длины. Данный метод принимает на вход номера телефонов, которым будет отправлено сообщение. Метод выводит на консоль номера этих телефонов.
- л) Изменить класс Phone в соответствии с концепцией JavaBean.

5. Создать класс Person, который содержит:

- а) поля `fullName`, `age`.
- б) методы `move()` и `talk()`, в которых просто вывести на консоль сообщение - "Такой-то Person говорит".
- в) Добавьте два конструктора - `Person()` и `Person(fullName, age)`.
Создайте два объекта этого класса. Один объект инициализируется конструктором `Person()`, другой - `Person(fullName, age)`.

6. **Создать класс "Матрица". Класс должен иметь следующие поля:**

- 1) двумерный массив вещественных чисел;
- 2) количество строк и столбцов в матрице.

Класс должен иметь следующие методы:

- 1) сложение с другой матрицей;
- 2) умножение на число;
- 3) вывод на печать;
- 4) умножение матриц - по желанию.

7. **Определить класс Reader**, хранящий такую информацию о

пользователе библиотеки: ФИО, номер читательского билета, факультет, дата рождения, телефон. Методы takeBook(), returnBook().

Разработать программу, в которой создается массив объектов данного класса. Перегрузить методы takeBook(), returnBook():

- takeBook, который будет принимать количество взятых книг. Выводит на консоль сообщение "Петров В. В. взял 3 книги".

- takeBook, который будет принимать переменное количество названий книг. Выводит на консоль сообщение "Петров В. В. взял книги: Приключения, Словарь, Энциклопедия".

- takeBook, который будет принимать переменное количество объектов класса Book (создать новый класс, содержащий имя и автора книги).

Выводит на консоль сообщение "Петров В. В. взял книги: Приключения, Словарь, Энциклопедия".

Аналогичным образом перегрузить метод returnBook(). Выводит на консоль сообщение "Петров В. В. вернул книги: Приключения, Словарь, Энциклопедия". Или "Петров В. В. вернул 3 книги".

Представьте отчет, который содержит: текст программного кода выполненных заданий и скрин результатов работы. Приложите к отчету скомпилированные программы по выполнению указанных заданий.

8. Все программы следует оформлять в соответствии с Code Convention, при оформлении кода использование комментариев обязательно. Используйте только пройденный материал И ТОЛЬКО ООП (классы обертки, коллекции и т.д. не использовать).

Список рекомендованной литературы

1. The Java Language Specification, Java SE 7 Edition [электронный документ] : <http://docs.oracle.com/javase/specs>
2. Bloch, Joshua. Effective Java™. Second Edition. – Addison-Wesley, 2008.

Вопросы для зачета:

1. Классы: определение, объявление, модификаторы доступа.
2. Члены класса: определение, модификаторы доступа, объявление.
3. Конструкторы классов.
4. Создание экземпляров классов. Безымянные классы.
5. Абстрактные методы и классы.
6. Окончательные методы и классы.
7. Статические члены класса.
8. Наследование. Отношения “Is-A” и “Has-A”.
9. Перегрузка и переопределение методов.
10. Вложенные классы

