Объект – экземпляр класса

Наследование - Суперкласс, подкласс. Extends – создание дочернего класса, производного от родительского класса. Если члены суперкласса объявлены как private, то подкласс не может иметь к ним доступа.

Полиморфизм - способность функции обрабатывать данные разных типов. Несколько конструктором, вы можем писать методы для разных типов данных и т.п. Перегрузка конструкторов, методов.

Инкапсуляция – как изоляция методов, переменных. Мы можем, например, изолировать методы одного класса от другого или его переменные.

Класс - это элемент, образующий основу Java. А поскольку класс определяет форму и сущность объекта, то он является той логической конструкцией, на основе которой построен весь язык Java. Существуют экземпляры класса, они же объекты. Класс – это как макет, а его объекты, например, скульптуры. конкретная форма и сущность.

Члены класса включают все члены, объявленные в этом классе, а также все члены (кроме конструкторов и методов завершения), объявленные во всех классах в иерархии наследования данного класса. Атрибут – переменная, параметр – ее значение. Методы, они же функции, выполняют какие-то действия. Они могут быть возвращаемые или нет(void). Метод- тело метода.

Модификаторы доступа:

public: публичный, общедоступный класс или член класса. Поля и методы, объявленные с модификатором public, видны другим классам из текущего пакета и из внешних пакетов.

private: закрытый класс или член класса, противоположность модификатору public. Закрытый класс или член класса доступен только из кода в том же классе.

protected: такой класс или член класса доступен из любого места в текущем классе или пакете.

Модификатор по умолчанию. Отсутствие модификатора у поля или метода класса предполагает применение к нему модификатора по умолчанию. Такие поля или методы видны всем классам в текущем пакете.

private int Cat(); // так выглядит метод по имени Cat

Cat(); // так выглядит конструктор класса Cat

Box defaultBox = new Box();

subOb.i 7;

subOb.j = 8;

Если переменная объявлена во вложенном блоке, то ее не будет видно за пределами него, однако если она объявлена во внешнем блоке, то во вложенном ее будет видно.

Благодаря static мы можем создать переменную или метод, что будет принадлежать к классу в целом. Это означает, что можно обратиться к статическому методу или статическому полю напрямую через класс без создания объекта. Иногда требуется определить член класса, который будет использоваться независимо от любого объекта этого класса. объявлять статическую переменную в статическом методе - так тоже нельзя.

Модификатор final - предназначен для создания констант. Если прописать final перед типом данных для поля, то такое поле нельзя будет изменить в ходе выполнения программы: Если final прописать перед названием класса, то класс не сможет иметь классов-наследников. В отличие от переменных константам можно присвоить значение только один раз.\

Пакеты нужны для того, например, когда закочились нормальные имена для переменных. package а.Ь.с; - создание пакета.

Объявления import (а не операторы ) являются по существу короткими средствами поддержки на уровне исходного кода: они позволяют ссылаться на тип или член static с помощью одного идентификатора (например , List , min ) в отличие от полного имени (например, java.util.List, Math.min ).

import раздел Объявления является элементом времени компиляции исходных кодов и не имеет никакого присутствия во время выполнения.

Объявление импорта позволяет ссылаться на элемент static или именованный тип простым именем, состоящим из одного идентификатора. Без использования соответствующего объявления import единственный способ ссылаться на тип, объявленный в другом пакете, или на член static другого типа-это использовать полное имя.