Внутренние классы

- Внутренний класс - класс в определении другого класса, может взаимодействовать с ним
- Позволяют группировать классы и управлять доступом к ним
- Создание внутренних классов - размещение определения класса внутри класса
- Внешний класс может содержать метод, возвращ внутр
- Доступ к объекту как к полю внешнего класса
- Объект внутр получает ссылку на внешний объект, который его создал -> может обращаться к членам внешнего объекта без доп. Уточнений
- Для внутренних классов доступны все элементы внешних (и закрытые в т.ч)
- При необходимости получения ссылки на объект внешнего класса - указать имя_внешнего.this для создания нового внутреннего объекта через внешний используется внешний.new
- Чтобы создать объект внутреннего должен быть объект внешнего. При создании вложенного класса ссылка не нужна

- Внутренние классы и восходящее преобразование
- Восходящее преобразование в базовому классу\интерфейсу. Внутренний класс - реализация интерфейса - может быть недоступным - сокр реализ
- Использование закрытых внутренних классов - полное сокрытие реализации

- Внутренние классы в методах и областях действия
- Причины создания классов там - Желание создавать и возвращать ссылки на интерфейс, хотите создать класс, решающий сложную задачу, но чтоб никто не видел
- Локальный внутренний класс - создание класса в области действия метода
- Анонимные внутренние классы -

Следующий пример выглядит довольно странно:

```
//: innerclasses/Parcel7.java
// Возвращение экземпляра анонимного внутреннего класса.

public class Parcel7 {
  public Contents contents() {
    return new Contents() { // Вставка определения класса
      private int i = 11;
      public int value() { return i; }
    }; // Точка с запятой здесь необходима
  }
  public static void main(String[] args) {
    Parcel7 p = new Parcel7();
    Contents c = p.contents();
  }
} ///:~
```

- Вложенные классы - статические внутренние классы (не хранят ссылку на объект внешнего)
- Для создания объекта не понадобится объект внешнего
- Не можете обращаться к членам не-статического объекта внеш из объекта влож
- Поля и методы статического внутреннего класса могут быть статич

- Классы внутри интерфейсов
- Любой класс в интерфейсе - открытый и статич
- Если класс вложен многократно, то доступ к полям внешнего все равно есть

- Внутренний класс способен независимо наследовать определенную реализацию -> внутренний класс не ограничен при наследовании, если внешний класс тоже наследует -> внутренние классы - множественное наследование без интерфейсов
- Можно создать произвольное количество экземпляров
- Один внешний может содержать много внутренних и там разная реализация
- Место создания объекта не привязано к месту создания объекта внеш класса
- Внутренний класс - отдельная сущность

- Замыкание - вызываемый объект сохраняет информацию о том, где был создан - внутренний содержит инф о месте создания (внешнем) -> позволяет вызывать объект, который произвел начальный вызов

- Система управления
- Каркас приложения - набор классов разработанных для опр круга задач. Для применения каркаса - наслед+реализ - изменяемая часть - шаблонный метод - постоянная
- Наследование от внутренних классов
- При осуществлении необходимо проинициализировать ссылку через внешний_класс.super
- При наследование от внешнего класса с внутренним ничего не происходит
- Локальный внутренний класс позволяет создать множество объектов этого класса