Inner Class

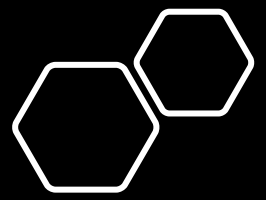
In Java

```
class Outer
{
    class Inner
    {
        void innerMethod()
        {
            System.out.println("Called the Inner class method");
        }
    }
}

class InnerDemo
{
    public static void main(String[]args)
    {
        Outer.Inner InnerObj = new Outer().new Inner();
        InnerObj.innerMethod();
    }
```

Gliederung

- Was war noch mal eine Klasse?
- Nested class in Java
- Static nested class
- Inner classes
- Vorführung im Code
- Vorteile und Nutzen
- Anwendungsbeispiel

```
{
    public boolean acceptance()
    {
        System.out.println("What type of vehicle would you l
        System.out.println("1 = Hatchback / 2 = Van");
        Scanner in1 = new Scanner(System.in);
        int type = Integer.parseInt(in1.nextLine());

        boolean accept;
        accept = true;

        if (type == 1)
        {
            if (hatchback())
            {
                accept = true;
            }
            else {accept = false;}
        }
        else if (type == 2)
        {
            if (van())
            {
                accept = true;
            }
            else {accept = false;}
        }
        return accept;
    }
```

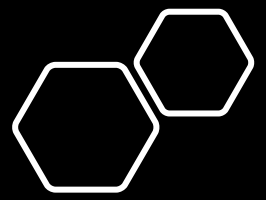


Was war noch mal eine Klasse?

- Kann Methoden halten
- Kann Variablen enthalten

```
public class MainClass
{

    public static void main(String[] args)
    {
        int price = calcPrice();
        System.out.println(price);
    }

    public static int calcPrice()
    {
        int price = 5+5;
        return price;
    }

}
```

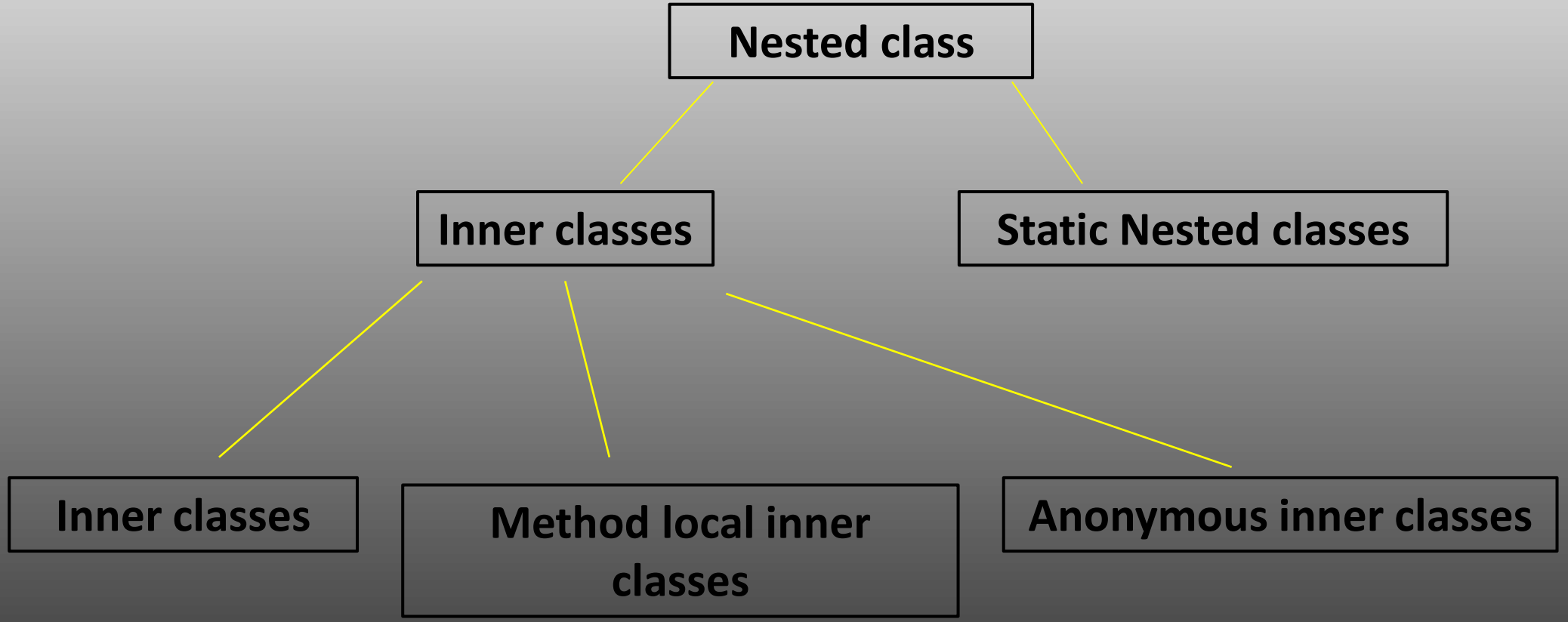
```
class Car
{
    double length = 5.5;
    double price  = 17000;
    int age       = 9;
    String color  = "blue";
}
```



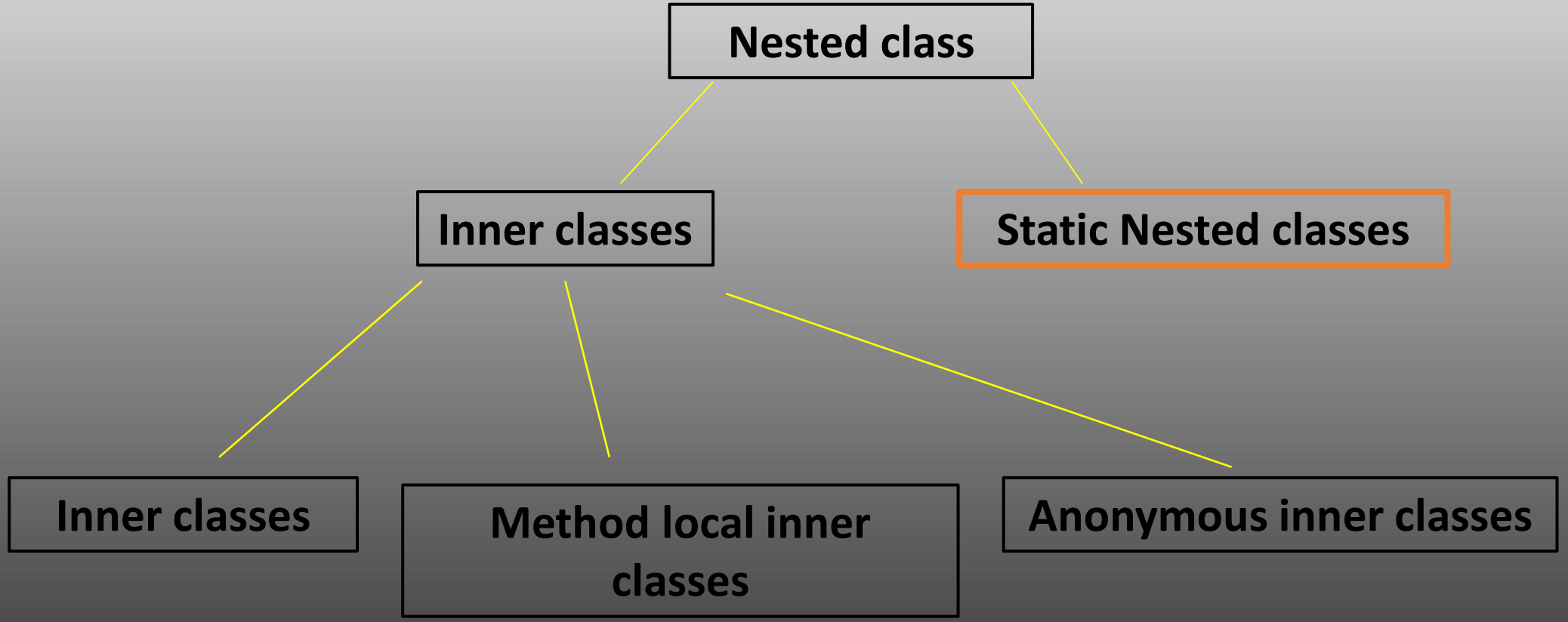
Nested class in Java

```
class Vehicles{
    static int wheels = 4;


    class Car1
    {
        double length = 5.5;
        double price  = 17000;
        int age       = 9;
        String color  = "blue";
    }

    class Car2
    {
        double length = 6;
        double price  = 23000;
        int age       = 6;
        String color  = "black";
    }
```

Nested class in Java



Nested class in Java

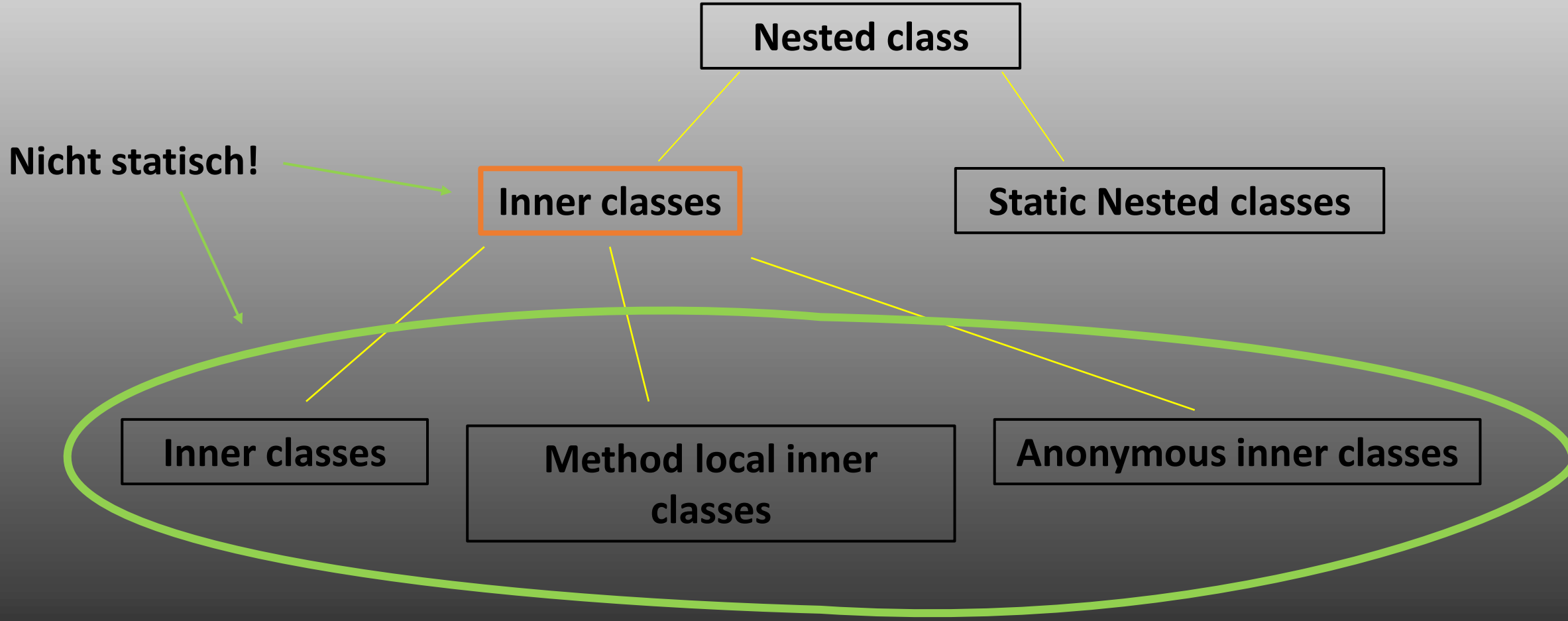


Static nested class

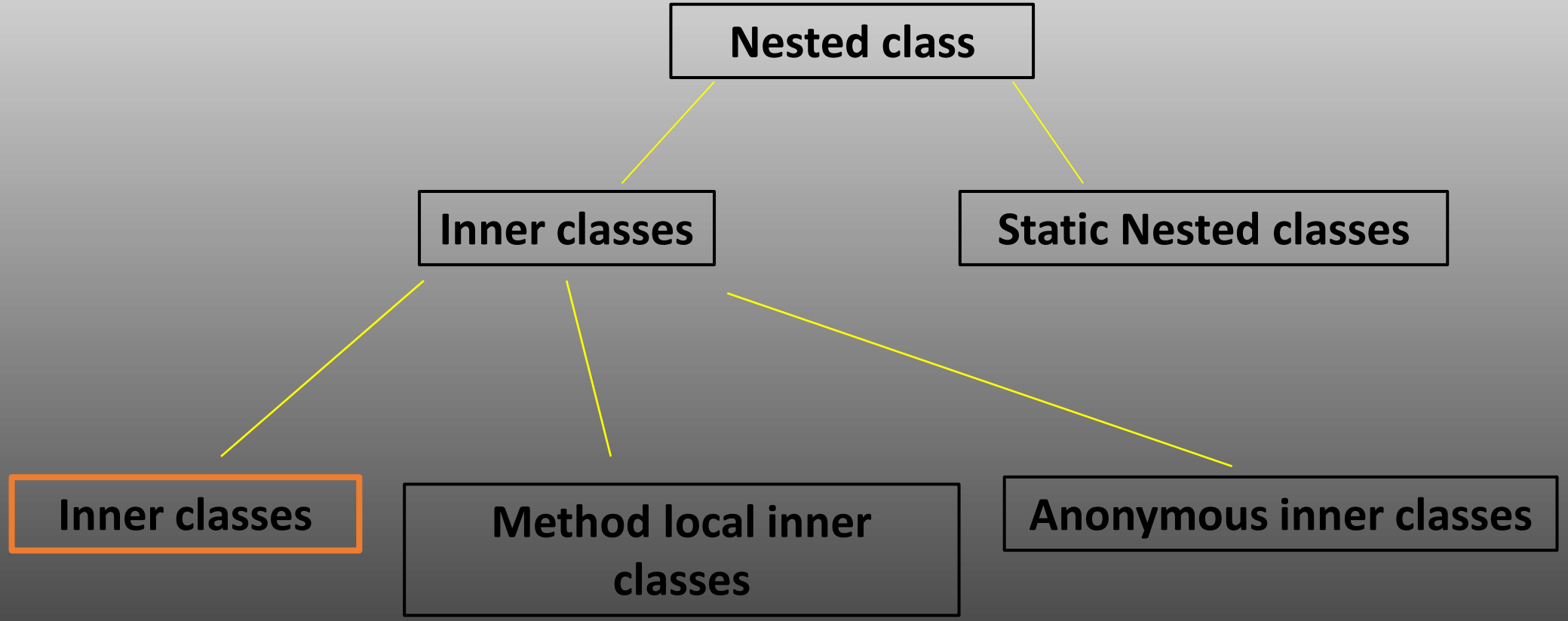
- Sind Statisch deklariert
- Verschachtelte Objekte können unabhängig und ohne äußere Klassenobjekte existieren
- Zugriff auf innere Klassen nur über Objektverweis
- Zugriff durch: `OuterClass.StaticNestedClass`
- Kann nur direkt auf statische äußere Sachen zugreifen
- Bsp.:

```
Outerclass.StaticNestedClass nestedObject = new OuterClass.StaticNestedClass();
```

Nested class in Java



Nested class in Java

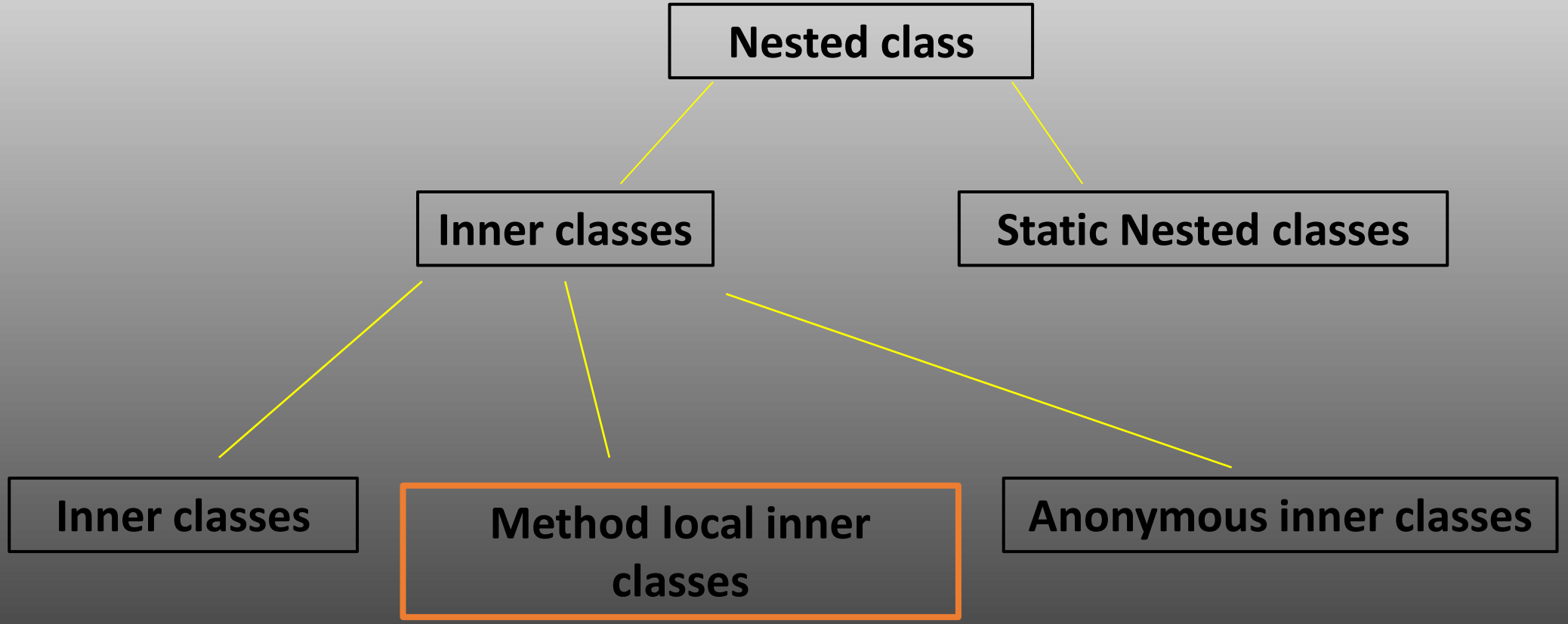


Inner class (member class)

- Beliebig tief schachtelbar
- Haben Zugriff auf alle (auch static / private) deklarierten Elemente der umgebenden Klasse
- Abhängig von äußerer Klasse
- Gleichnamige Variablen werden durch innere Klasse überschrieben
- Zugriff aus innerer Klasse auf äußere Klasse: OuterClass.this.s1
- Objektbildung mit ungewöhnlicher Schreibweise:

```
OuterClass oc = new OuterClass();
OuterClass.MemberClass mc = oc.new MemberClass();
```

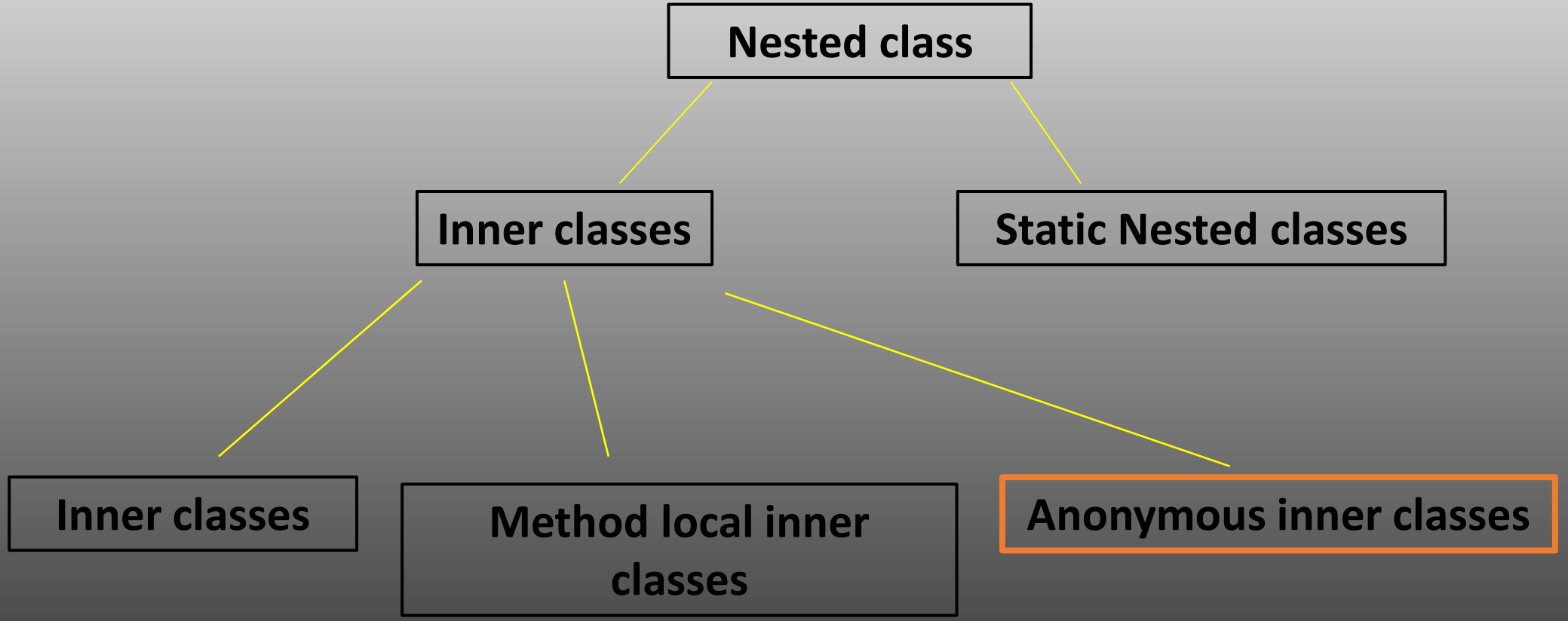
Nested class in Java



Local class

- In Anweisungsblöcken, Methoden, Konstruktoren etc. einer äußeren Klasse definiert
- Interfaces nicht möglich
- Zugriffsmodifikatoren (private/public) bei Deklaration nicht möglich
- Keine Methoden oder statische Variablen in Lokaler Klasse
- “static final” erlaubt
- Direkter Zugriff auf Variablen der äußeren Klasse
- Umgekehrt Zugriff nur über ein Objekt der inneren Klasse

Nested class in Java



Anonymous class

- Hat keinen sichtbaren Namen
- Wird für ein einzelnes Objekt erstellt
- Erstellungsmöglichkeiten → abstract class; interface

Vorteile und Nutzen

Vorteile

- Kann auf alle Variablen und Methoden der äußeren Klasse zugreifen
- Um besser lesbaren und beständigen code zu entwickeln
- Code-Optimierung: benötigt weniger Text

Nutzen

- So Programmieren, dass nur bestimmte Klassen Zugriff auf eine andere Klasse haben (versteckte Klassen)

Quellen

- https://www.javatpoint.com/java-inner-class
- https://www.geeksforgeeks.org/nested-classes-java/
- https://javabeginners.de/Klassen_und_Interfaces/Innere_Klassen.php
- https://docs.oracle.com/javase/tutorial/java/javaOO/nested.html
- https://www.w3schools.com/java/java_inner_classes.asp