

# static Variables - Speicherdarstellung

```
class Program
{
    static void Main(string[] args)
    {
        Pupil p1 = new Pupil("Franz");
        Pupil p2 = new Pupil("Hans");
        Pupil p3 = new Pupil("Reinhard");
    }
}

class Pupil
{
    static int Counter { get; private set; }

    public string FirstName { get; set; }
    public int CatalogNr { get; set; }

    public Pupil (string FirstName)
    {
        this.FirstName = FirstName;
    }
}
```

Name	Value
...	
p1	
p2	
p3	

Object of Pupil  
FirstName = „Franz“

Object of Pupil  
FirstName = „Hans“

Object of Pupil  
FirstName = „Reinhard“

class variable  
Counter = 0;

*Deklaration der  
Klassenvariable mit  
Schlüsselwort „static“*

# Anforderung – Objekte zählen

```
class Program
{
    static void Main(string[] args)
    {
        Pupil p1 = new Pupil("Franz");
        Pupil p2 = new Pupil("Hans");
        Pupil p3 = new Pupil("Reinhard");

        Console.WriteLine("Es wurden {0} Objekte der Klasse Pupil instanziiert", Pupil.Counter);
    }
}

class Pupil
{
    public static int Counter { get; private set; }

    public string FirstName { get; set; }
    public int CatalogNr { get; set; }

    public Pupil (string FirstName)
    {
        this.FirstName = FirstName;
        Counter++;
    }
}
```

**Klassenvariablen können nicht über die  
Objekte sondern nur über die Klasse  
angesprochen werden!!!  
Der Zugriff muss erlaubt sein:  
Modifizierer: public**

# Klassen-Methoden



Neben **klassenglobalen** Eigenschaften können auch Methoden klassenglobal definiert werden.

Die Definition erfolgt mit dem Schlüsselwort „static“:

```
public static void UpdateTimeStamp ()  
{  
    FirstObjectCreated = DateTime.Now;  
}
```

In diesen Methoden besteht die Zugriffsmöglichkeit auf Klassenvariablen – nicht jedoch auf Objektvariablen!

# Klassen-Methoden



```
56  
57 public static void UpdateTimeStamp ()  
58 {  
59     // In dieser Methode gibt es keinen Zugriff auf Objekt-Variablen (auf Objekte im engeren Sinn)  
60     // weil diese Methode der Klasse zugeordnet ist und nicht den Objekten  
61     FirstName = "Seppi"; // Compiler-Error  
62     FirstObjectCreated = DateTime.Now;  
63 }
```

## Error List

1 Error 0 Warnings 0 Messages

Description
1 An object reference is required for the non-static field, method, or property 'StaticConcept.Pupil.FirstName.get'

Fehler bei Zugriff auf Objektvariablen mit statischen Methoden:

„An object reference is required for the non-static field, method or property.“

# Übungsaufgabe



Erstellen Sie eine statische Klasse, die das Umrechnen von Kilowatt in Pferdestärke ermöglicht. Folgende Methoden – Signaturen sind anzubieten:

- `double KilowattToPS (double Kilowatt);`
- `double PSToKilowatt (double PS);`

Der Umrechnungswert von PS zu KW beträgt 1,36 .

# Übungsaufgabe: Brutto-Netto-Rechner

Erstellen Sie einen vereinfachten Brutto-Netto-Rechner, der basierend auf einem Jahresgehalt die Sozialversicherung, die Lohnsteuer und das Netto-Jahresgehalt errechnet.

Realisieren Sie diesen Rechner mit einer statischen Klasse.

Berechnung der Sozialversicherungsabgabe:

# Übungsaufgabe: Brutto-Netto-Rech

Vereinfachte Berechnung des Netto-Gehalts:

Vom Bruttogehalt ist zuerst der Sozialversicherungsbeitrag zu errechnen und in Folge abzuziehen. Dabei ist zwischen Arbeitern und Angestellten zu unterscheiden. Die Sozialversicherungsbemessungsgrundlage ist mit 4.650,- monatlich gedeckelt.

1) bei laufenden Bezügen =	Angestellte: 18,07 %
	Arbeiter: 18,20 %
Höchstbeitragsgrundlage =	€ 4.650,-- monatlich

# Übungsaufgabe: Brutto-Netto-Rechner

Brutto-Gehalt abzüglich Sozialversicherungsbeitrag ergibt die Bemessungsgrundlage für die Lohnsteuer. Die Lohnsteuer wird progressiv laut folgender Tabelle errechnet:

- » 1.) Einkünfte von 0 bis 11.000 Euro pro Jahr: 0,00 % Einkommenssteuer
- » 2.) Einkünfte von 11.000 bis 25.000 Euro pro Jahr: 36,50 % Einkommenssteuer
- » 3.) Einkünfte von 25.000 bis 60.000 Euro pro Jahr: 43,2143 % Einkommenssteuer
- » 4.) Einkünfte über 60.000 Euro pro Jahr: 50,00 % Einkommenssteuer.

**Aufgabe:** Erstellen Sie einen vereinfachten Brutto-Netto-Rechner mit Hilfe einer statischen Klasse!