

SWE 1 - stat. ⇔ dynam. Komponenten

	dynamisch	statisch
Deklaration	ohne <code>static</code>	mit <code>static</code>
existieren	in jedem Objekt	nur einmal pro Klasse
Felder werden angelegt	beim Erzeugen des Objekts	beim Laden der Klasse
Konstruktor wird aufgerufen	beim Erzeugen des Objekts	-
Felder werden freigegeben	vom GC, wenn kein Zeiger mehr auf das Objekt zeigt	am Ende des Programms
Felder werden angesprochen	<code>obj.field</code>	<code>Class.field</code>
Methoden werden aufgerufen	<code>obj.meth();</code>	<code>Class.meth();</code>

SWE 1 - Anagramme (1)

Gesucht ist ein Java-Programm, das zwei Wörter einliest und überprüft, ob das zweite Wort ein Anagramm des ersten Wortes ist.

Ein Wort ist dann ein Anagramm eines anderen, wenn es sich durch Permutation der Buchstaben erzeugen lässt.

Beispiele:

<code>item</code>	<code>time</code>	Anagramm
<code>acres</code>	<code>caress</code>	kein Anagramm
<code>scare</code>	<code>cares</code>	Anagramm
<code>react</code>	<code>crater</code>	kein Anagramm

SWE 1 - Anagramme (2)

```
public class Anagram extends Basic {
    public static void main(String[] arg) {
        String s1, s2;

        write("first word ? ");
        s1 = readName(); readLn();
        write("second word ? ");
        s2 = readName(); readLn();

        write("The words are ");
        if (!isAnagram(s1, s2))
            write("NOT ");
        write("anagrams.");
    } // end main
} // end Anagram
```

SWE 1 - Anagramme (3)

```
static boolean isAnagram(String s1, String s2) {
    boolean isAnagram = false;

    if (s1.length() == s2.length()) {
        // create editable copy of s2
        StringBuffer sb2 = new StringBuffer(s2);

        // stike out letters of s1 in sb2
        for (int i = 0; i < s1.length(); i++) {
            int pos = sb2.toString().indexOf(s1.charAt(i));
            if (pos != -1)
                sb2.setCharAt(pos, (char)0);
        } // end for
    }
}
```

SWE 1 - Anagramme (4)

```
// check if all letters are striked out
int i = 0;
while (i < sb2.length() && sb2.charAt(i) == (char)0)
    i++;

isAnagram = (i == sb2.length());
} // end if

return isAnagram;
} // end isAnagram
```

SWE 1 - Wh: Arrays ↔ Klassen

Arrays	Klassen
Bestehen aus mehreren <i>gleichartigen</i> Elementen	Können aus mehreren <i>verschiedenartigen</i> Feldern bestehen
Die Elemente eines Arrays haben keinen Namen, sondern werden über einen <i>Index</i> angesprochen	Die Felder einer Klasse haben einen <i>Namen</i> und werden über diesen angesprochen
Die Anzahl der Elemente wird bei der <i>Erzeugung</i> des Arrayobjekts festgelegt.	Die Anzahl der Felder wird bei der <i>Deklaration</i> der Klasse festgelegt.