

Hochschule Harz	FB Automatisierung und Informatik
Programmierung2	Dipl.-Inf., Dipl.-Ing. (FH) M. Wilhelm
<u>Tutorial 08:</u>	„Programmierung 2“ für MI / WI Thema: Singleton

Versuchsziele

Kenntnisse in der Anwendung von:

- Singleton
- Der Student soll erkennen, dass die Verwendung eines Singleton gegenüber eines normalen Objektes erhebliche Vorteile hat (absolute Lokalität).

Tutorial08a:

In dieser Aufgabe soll mit Hilfe einer grafischen Oberfläche eine gemeinsame Instanz entwickelt und getestet werden. Es gibt eine gemeinsame Instanz der Klasse „Farbe“ bzw. ein Singleton. Jedes neue Fenster muss Zugriff auf die Instanz erhalten.

Aufgaben

1. Teilaufgaben: Projekt erstellen und aufbauen:

- Projektname: Tutorial08a

2. Teilaufgaben: Event-Methoden:



Abbildung 1 Überblick der GUI: Tutorial08a

- Bauen Sie in Ihr Projekt den vorgegeben Quellcode ein (Homepage).

2. Teilaufgabe: Klasse Farbe

Diese Aufgabe implementiert eine Klasse „Farbe“, die ein öffentliches Attribut „farbe“ hat. Dieses wird mit Color.BLUE initialisiert.

3. Teilaufgabe: Hauptframe, Methode create_click()

In dieser Methode soll ein neues JFrame „Tutorial08a“ erstellt und angezeigt werden. In der Vorgabe wird die Position per Zufallszahlen bestimmt. Die Verwendung der Klasse „Farbe“ kommt später.

4. Teilaufgabe: Hauptframe

Jeder Instanz des JFrame „Tutorial08a“ soll eine **identische** Instanz der Klasse „Farbe“ zugewiesen werden.

Beschränkungen:

- Die Instanz darf keine statische Variable sein.
- **Es darf nur ein Konstruktor verwendet werden.**

Aufgaben:

- Überlegen Sie, wo die **einzig**e Instanz aller JFrames erzeugen werden muss.
- Vervollständigen Sie Ihren Quellcode, so dass die erste Instanz des JFrame schon die Instanz erhält.
- Vervollständigen Sie Ihren Quellcode, so dass in „create_click“ die Instanz **weitergeleitet** wird.

5. Teilaufgabe: „Farb-Schalter“

Aufgaben:

- Wenn ein Farb-Schalter angeklickt wird, so wird die Hintergrundfarbe des aktuellen JFrames gesetzt.
- Wenn ein Farb-Schalter angeklickt wird, so wird die Farbe in der Instanz „farbe“ gesetzt.
- Damit ändert sich aber nur die Farbe im aktuellen JFrame (gewollt).

6. Teilaufgabe: „get-Schalter“

Aufgaben:

- Wenn der get-Schalter angeklickt wird, so wird die Hintergrundfarbe des Panels durch den aktuellen Wert in der Instanz „farbe“ gesetzt.

7. Teilaufgabe: „Testablauf“

1. Starten

- Das Panel ist blau.

2. create_click

- das neue Fenster hat auch die Farbe blau.

3. Im ersten Fenster wird die Farbe auf rot gesetzt

4. create_click im ersten Fenster

- das neue Fenster hat die Farbe rot.

5. Im zweiten Fenster wird die Farbe grün gesetzt.

- Nur das zweite Fenster hat ein grünes Panel.

6. create_click im ersten Fenster.

- **das neue Fenster hat die Farbe grün, obwohl das erste Fenster ein rotes Panel hat.**

7. Im ersten Fenster wird der „get“-Schalter angeklickt.

- Nun erscheint hier auch die Farbe grün.

Tutorial08b

Nun soll ein Singleton-Pattern eingebaut werden.
Bitte den Anhang beachten.

1. Teilaufgaben: Neues Projekt erstellen und umbauen:

- Projektname: Tutorial08b

2. Teilaufgaben: Event-Methoden:



Abbildung 2 Überblick der GUI: Tutorial08b

- Bauen Sie in Ihr Projekt den vorgegeben Quellcode ein (Homepage: Tutorial08b.java).
- **Es darf keine globale Instanz des Klasse „Farbe“ im JFrame „Tutorial08b“ vorhanden sein.**

3. Teilaufgabe: Klasse Farbe

Diese Aufgabe implementiert eine Klasse „Farbe“, die ein öffentliches Attribut „farbe“ hat. Dieses wird mit Color.BLUE initialisiert. **Diese Klasse soll aber als Singleton implementiert werden.**

4. Teilaufgabe: Hauptframe, Methode setGUI

In dieser Methode soll das JPanel mit der aktuellen Hintergrundfarbe gesetzt werden. Dazu wird eine Instanz per Singleton „erzeugt“ und die Hintergrundfarbe gesetzt.

5. Teilaufgabe: Hauptframe, Methode create_click()

In dieser Methode soll ein neues JFrame „Tutorial08b“ erstellt und angezeigt werden. In der Vorgabe wird die Position per Zufallszahlen bestimmt. **Die Verwendung der Klasse „Farbe“ ist hier nicht notwendig und nicht erlaubt.**

6. Teilaufgabe: „Farb-Schalter“

Aufgaben:

- Wenn ein Farb-Schalter angeklickt wird, so wird die Hintergrundfarbe des aktuellen JFrames gesetzt.
 - Dazu wird **lokal** eine Instanz der Klasse „Farbe“ geholt.
 - Das Panel wird gesetzt.
 - Die aktuelle Farbe wird auch im Singleton-Objekt gesetzt.
- Damit ändert sich aber nur die Farbe im aktuellen JFrame.

7. Teilaufgabe: „get-Schalter“

Aufgaben:

- Wenn der get-Schalter angeklickt wird, so wird die Hintergrundfarbe des Panels gesetzt.
 - Dazu wird **lokal** eine Instanz der Klasse „Farbe“ geholt.
 - Das Panel wird gesetzt mit dem aktuellen Wert des Singleton-Objektes gesetzt.

8. Teilaufgabe: „Testablauf“

1. Starten

- Das Panel ist blau

2. create_click

- das neue Fenster hat auch die Farbe blau

3. Im ersten Fenster wird die Farbe auf rot gesetzt

4. create_click im ersten Fenster

- das neue Fenster hat die Farbe rot

5. Im zweiten Fenster wird die Farbe grün gesetzt

- Nur das zweite Fenster hat ein grünes Panel

6. create_click im ersten Fenster

- das neue Fenster hat die Farbe grün, obwohl das erste Fenster ein rotes Panel hat.

7. Im ersten Fenster wird der „get“-Schalter angeklickt.

- Nun erscheint hier auch die Farbe grün

Tutorial08c

Nun soll ein Singleton-Pattern mit einer generischen Klasse kombiniert werden.
Bitte den Anhang beachten.

1. Teilaufgaben: Neues Projekt erstellen und aufbauen:

- Projektname: Tutorial08c

2. Teilaufgaben: Event-Methoden:



Abbildung 3 Überblick der GUI: Tutorial08c

- Kopieren Sie den Quellcode aus „Tutorial8b.java“ in Ihr Projekt.
- Nennen Sie die Klasse „Tutorial8b“ in „Tutorial8c“ um.
- Diese Variante verwendet eine generische Singleton-Klasse. Es darf wiederum keine globale Instanz der Klasse „Daten“ im JFrame „Tutorial08c“ vorhanden sein.

3. Teilaufgabe: Klasse Daten

Diese Aufgabe implementiert eine generische Klasse „Daten“.

- `public class Daten<T>`
- Attribut:
 - T data, initialisiert mit null
- Methoden:
 - **setData**
 - Parameter: T data
 - **getData**
 - Rückgabewert: T
 - wenn `this.data` null ist, soll eine `NullPointerException` ausgeworfen werden.
 - `throw new NullPointerException("Singleton nicht initialisiert");`
 - **isDataNull**
 - Bool'scher Rückgabewert, ob `this.data` den Wert null hat

4. Teilaufgabe: Hauptframe, Methode setGUI

- Da in der Singleton-Klasse das Attribut mit null initialisiert wurde, muss es beim ersten Aufruf gesetzt werden.
- Dazu wird eine Instanz per Singleton „erzeugt“.
- Nun wird mit der Methode „isDataNull“ abgeprüft, ob die Farbe schon gesetzt wurde.
 - Wenn nicht, soll sie mit `Color.BLUE` gesetzt werden.
- Danach soll das JPanel mit der aktuellen Hintergrundfarbe gesetzt werden.

5. Teilaufgabe: Hauptframe, Methode create_click()

In dieser Methode soll ein neues JFrame „Tutorial08c“ erstellt und angezeigt werden. In der Vorgabe wird die Position per Zufallszahlen bestimmt. **Die Verwendung der Klasse „Farbe“ ist hier nicht notwendig und nicht erlaubt.**

6. Teilaufgabe: „Farb-Schalter“

Aufgaben:

- Wenn ein Farb-Schalter angeklickt wird, so wird die Hintergrundfarbe des aktuellen JFrames gesetzt.
 - Dazu wird **lokal** eine Instanz der Klasse „Farbe“ geholt.
 - Das Panel wird gesetzt.
 - Die aktuelle Farbe wird auch im Singleton-Objekt gesetzt.
- Damit ändert sich aber nur die Farbe im aktuellen JFrame.

7. Teilaufgabe: „get-Schalter“

Aufgaben:

- Wenn der get-Schalter angeklickt wird, so wird die Hintergrundfarbe des Panels gesetzt.
 - Dazu wird **lokal** eine Instanz der Klasse „Farbe“ geholt.
 - Das Panel wird gesetzt mit dem aktuellen Wert des Singleton-Objektes gesetzt.

8. Teilaufgabe: „Testablauf“

1. Starten

- Das Panel ist blau

2. create_click

- das neue Fenster hat auch die Farbe blau

3. Im ersten Fenster wird die Farbe auf rot gesetzt

4. create_click im ersten Fenster

- das neue Fenster hat die Farbe rot

5. Im zweiten Fenster wird die Farbe grün gesetzt

- Nur das zweite Fenster hat ein grünes Panel

6. create_click im ersten Fenster

- das neue Fenster hat die Farbe grün, obwohl das erste Fenster ein rotes Panel hat.

7. Im ersten Fenster wird der „get“-Schalter angeklickt.

- Nun erscheint hier auch die Farbe grün

Anhang

Singleton

```
public class Singleton {
    private static Singleton instance = new Singleton();
    protected Singleton(){
    }

    public static Singleton getInstance(){
        return instance;
    }

    public String data="ich bin ein Singleton";
}
```

Generische Klasse

Deklaration:

```
class MyClass<T> {

    private MyClass() {
    }

    private T data=null;

    public void setData(T data) {
        this.data = data;
    }

    public T getData() {
        return this.data;
    }

}
```

Benutzung:

```
MyClass<String> zeile = new MyClass<String>();
zeile.setData( "ich bin eine Zeile" );
System.out.println( zeile.getData() );
```