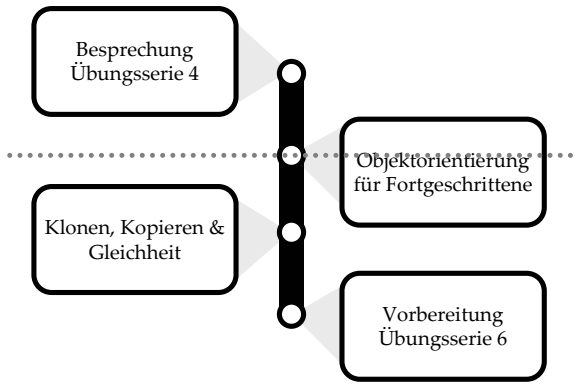
Montag, 1. Dezember & Dienstag, 3. Dezember



1 Dry Gin

```
from
    metro_stations.start
    -- "set found to false"
until
    metro_stations.after or found
loop
    if ( metro_stations.item.name = "Balard" ) then
        -- "set found to true"
    end
if ( not metro_stations.after ) then
    -- "perform some operations on the found station"
end
end
```

1 Dry Gin II

```
from
    metro_stations.start
    -- "set found to false"
until
    metro_stations.after or found
loop
    if ( metro_stations.item.name = "Balard" ) then
        -- "set found to true"
    end
    metro_stations.forth
if ( not metro_stations.after ) then
    -- "perform some operations on the found station"
end
end
```

1 Dry Gin III

```
from
    metro_stations.start
    -- "set found to false"
until
    metro_stations.after or found
loop
    if ( metro_stations.item.name = "Balard" ) then
        -- "set found to true"
    end
    metro_stations.forth
    if ( not metro_stations.after ) then
        -- "perform some operations on the ..."
    end
end
```

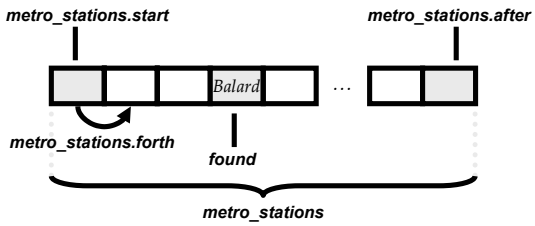
Teil, der wiederholt wird

1 Dry Gin IV

```
from
    metro_stations.start
    -- "set found to false"
until
    metro_stations.after or found
loop
    if ( metro_stations.item.name = "Balard" ) then
        -- "set found to true"
    end
    metro_stations.forth
end

if ( not metro_stations.after ) then
    -- "perform some operations on the found station"
end
```

Metrostationen, `start`, `forth`, `after`



2 Changing the display

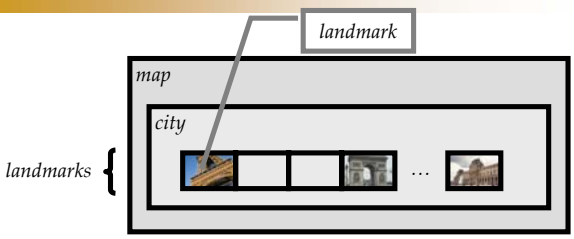
```
class
    SWITCH_REPRESENTATION

inherit
    EXERCISE
        redefine
            start
        end
    SHARED_MAP

feature

    start is
            -- switch between landmark representations
        do
            -- was soll passieren?
        end
end
```

die Stadt der Liebe



was ist ein landmark?

class *LANDMARK* **...**
feature *set_icon_on(b : BOOLEAN)* **is ...**

LANDMARK_DISPLAYER_IMP

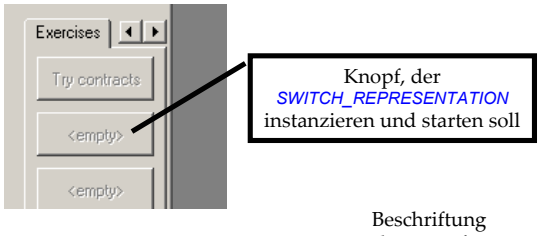
wie wird das dann angezeigt?

feature *initialize* **...**

LANDMARK_DISPLAYER_IMP.initialize

```
feature
    initialize is
    do
      if not element.has_graphical_representation then
        Precursor
      else
        if element.icon_on then
            basic_repentation.extend( icon )
            ...
        else
            basic_repentation.extend(
             circle ( default_radius, standard_color, True)
            )
            ...
        end
      end
    end
end
```

wie starten wir das?



Knopf, der
SWITCH_REPRESENTATION
instanzieren und starten soll

Beschriftung
des Knopfes

extend(**create** {*SWITCH_REPRESENTATION*}, *"Landmarks"*)

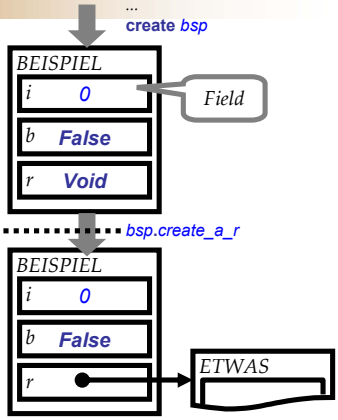
instanziere ein Objekt
dieser Klasse

Objektstruktur

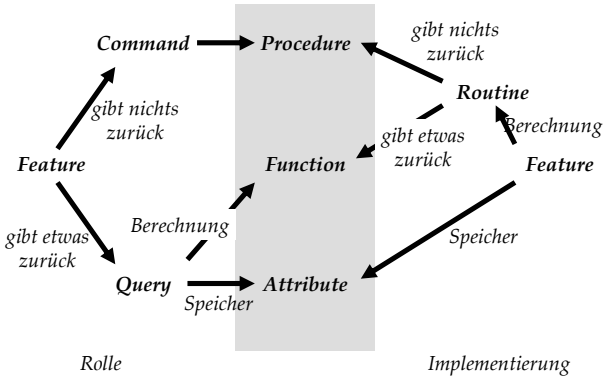
```
class BEISPIEL
feature
   i : INTEGER
   b : BOOLEAN
   r : ETWAS
   inc_i is
      do
         i := i + 1
      end
   make_b_true is
      do
         b := true
      end
   create_a_r is
      do
         create r
      end
end
```

Attribut

bsp : BEISPIEL
...
create *bsp*



Feature Kategorien



Zuweisungen

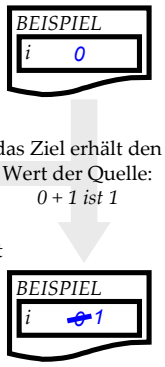
```
class BEISPIEL
feature
   i : INTEGER
   b : BOOLEAN
   r : ETWAS
   inc_i is
      do
         i := i + 1
      end
   make_b_true is
      do
         b := true
      end
   create_a_r is
      do
         create r
      end
end
```

Ziel Quelle

`i := i + 1`

das Ziel erhält den Wert der Quelle: *0 + 1 ist 1*

Zuweisungszeichen (bestehend aus Doppelpunkt und Gleichheitszeichen)



Der Vorgang der Zuweisung

`i := i + 1`

1. Quelle auswerten
    1. i lesen & einsetzen
    2. i + 1 berechnen
2. Ziel mit Resultat überschreiben