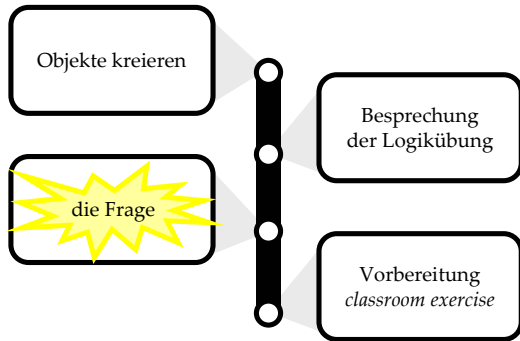
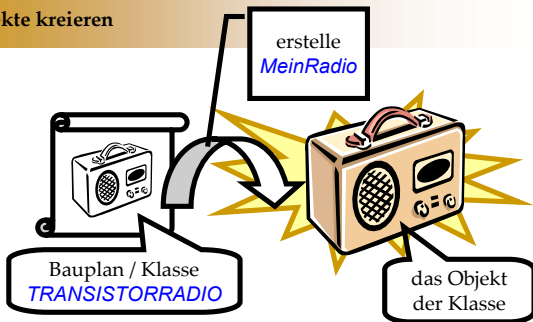


Mittwoch, 12. November



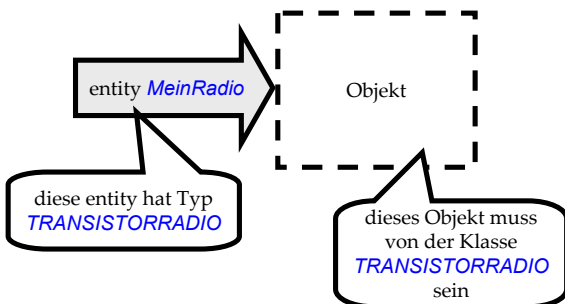
Objekte kreieren



- das Wort *MeinRadio* ist ein *Identifizier*
- *MeinRadio* steht für ein *Runtime object*
- *MeinRadio* ist deshalb eine *Entity*
- *MeinRadio* ist an ein Objekt der Klasse *TRANSISTORRADIO* *attached / angeschlossen*

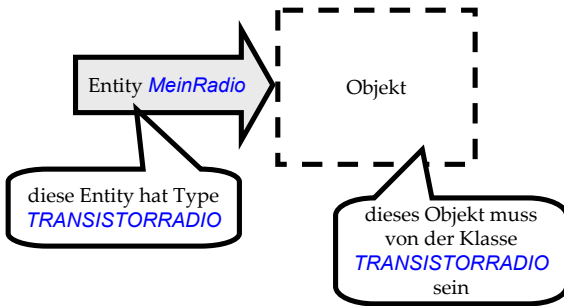
Entity & Object

eine Entity ist eventuell an einem Objekt angeschlossen



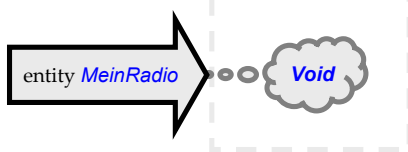
Type

eine Entity ist **eventuell** an einem Objekt angeschlossen



Void

an was die Entity angeschlossen, wenn da kein Objekt ist?



Void bedeutet *Lehre, Hohlraum*

ein **Void** kann überhaupt nichts (hat keine Features)

Feature call

MeinRadio hat Type **TRANSISTORRADIO**
deshalb hat das *attached object* auch ein Feature **abspielen**

MeinRadio.abspielen

die Entity *MeinRadio* wurde aber noch an nichts angeschlossen
was passiert jetzt?



Exceptions

eine Exception ist ein Auftritt eines abnormalen Ereignisses während das Programm ausgeführt wird. Jede Exception hat einen Namen, welcher als failure message angezeigt wird.

Exceptions treten immer dann auf, wenn nicht undefiniert ist, was der Computer tun soll.

Undefiniert ist gefährlich & verursacht Fehler

Fehler kann man kontrollieren, indem man bewusst etwas versucht, und dann entsprechend agiert, wenn ein Fehler auftritt.

Create

ein Objekt einer Klasse erzeugen und es an eine Entity anschliessen:

```
feature
  meinRadio : TRANSISTORRADIO
  mach_etwas is
    -- erstelle ein Radio & schliesse es an die Entity
    -- meinRadio
  do
    create meinRadio
  end
```

Creation procedure

wenn eine Klasse eine nicht-triviale Invariante hat, dann muss sie eine oder mehrere *creation procedures* haben, deren Funktion es ist, sicherzustellen, dass jede Instanz die Invariante erfüllt.

```
class TRANSISTORRADIO
  create sicherstellendes_Feature
  feature
    eingelegte_Batterie : BATTERIE
    sicherstellendes_Feature is
      -- erstelle ein Radio & setze Batterien ein
      do
        create eingelegte_Batterie
      end
  invariant Batterie_eingelegt: eingelegte_Batterie /= Void
end
```
