

Das Entwurfsmuster MVC

Model – View – Controller

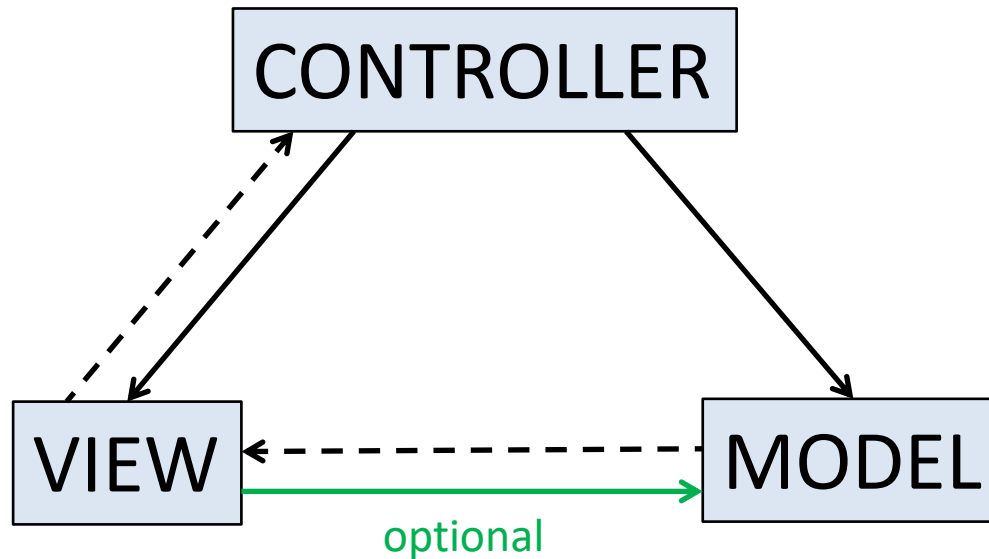
MVC ist ein zusammengesetztes Muster (compound pattern).

Es vereinigt:

- Strategiemuster (strategy pattern)
- Beobachtermuster (observer pattern)

MVC

- Model: fachliche Logik eines Programms
- Views: Sichten auf die Daten
- Controller: Steuerelemente für Benutzereingaben



Gestrichelt: über Beobachter

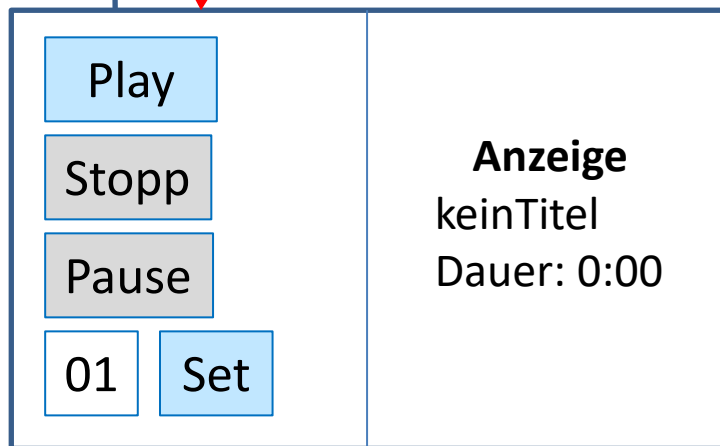
Meine Play-Taste
wurde gedrückt!



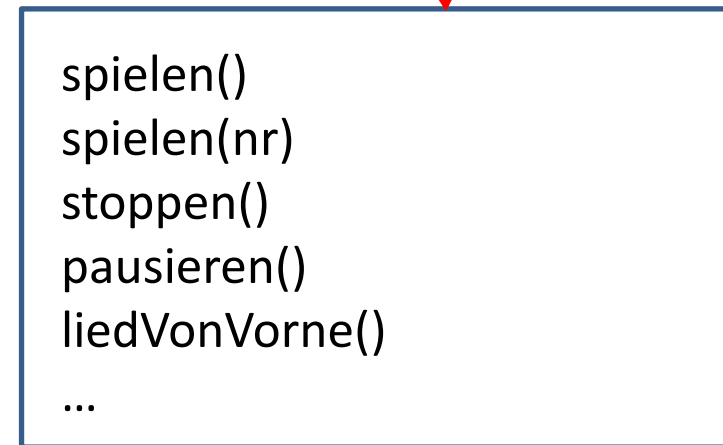
Controller

OK! Schalte die
Play-Taste auf
deaktiviert.
Aktiviere Stopp-
und Pause-Taste

`model.spielen();`



View



Model
mp3-Player

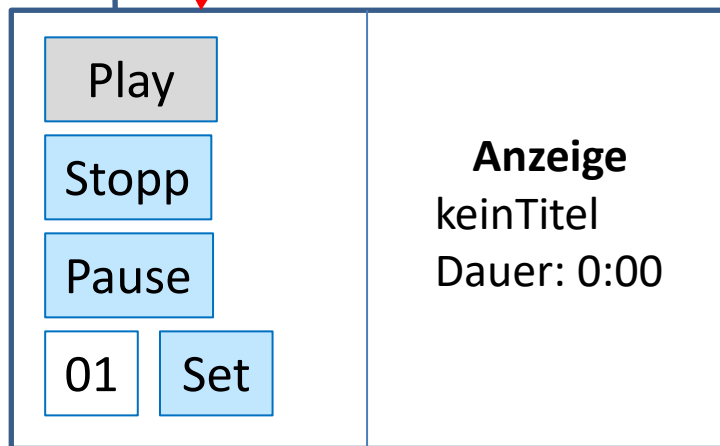
Meine Play-Taste
wurde gedrückt!



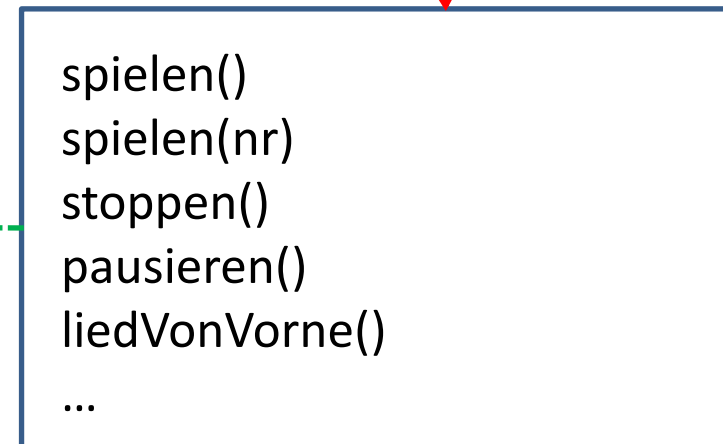
Controller

OK! Schalte die
Play-Taste auf
deaktiviert.
Aktiviere Stopp-
und Pause-Taste

`model.spielen();`



View



Model
mp3-Player

Ich spiele jetzt "TNT",
dauert 03:28

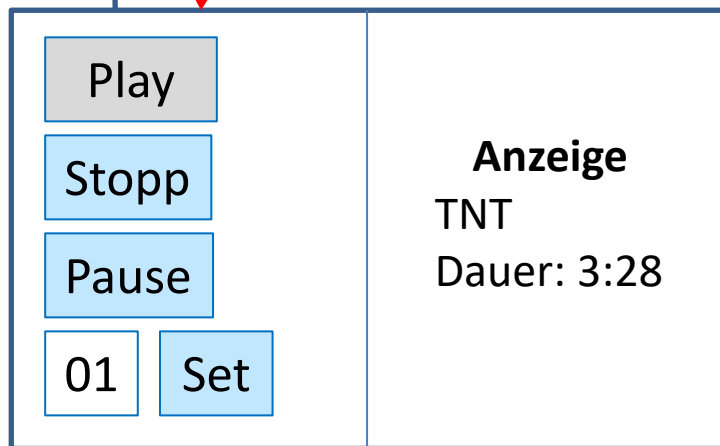
Meine Play-Taste
wurde gedrückt!



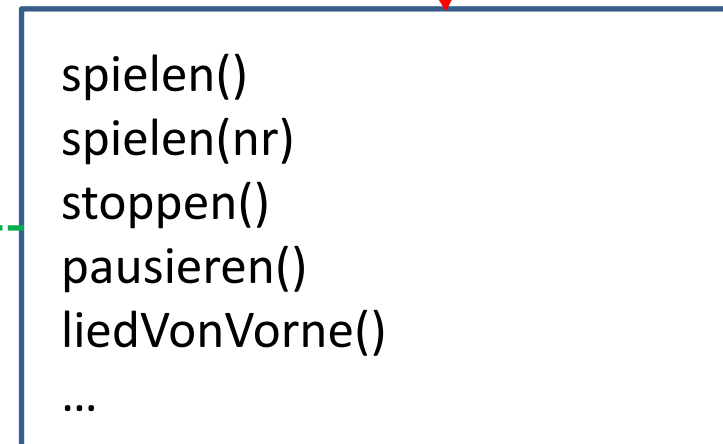
Controller

OK! Schalte die
Play-Taste auf
deaktiviert.
Aktiviere Stopp-
und Pause-Taste

`model.spielen();`



View



Model
mp3-Player

Ich spiele jetzt "TNT",
dauert 03:28

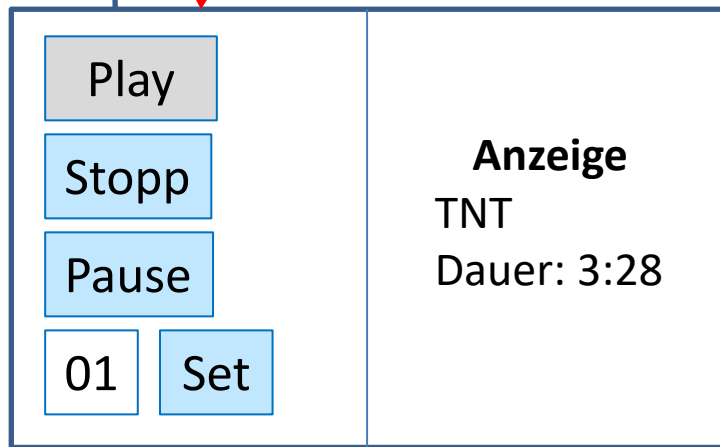
Meine Pause-Taste
wurde gedrückt!



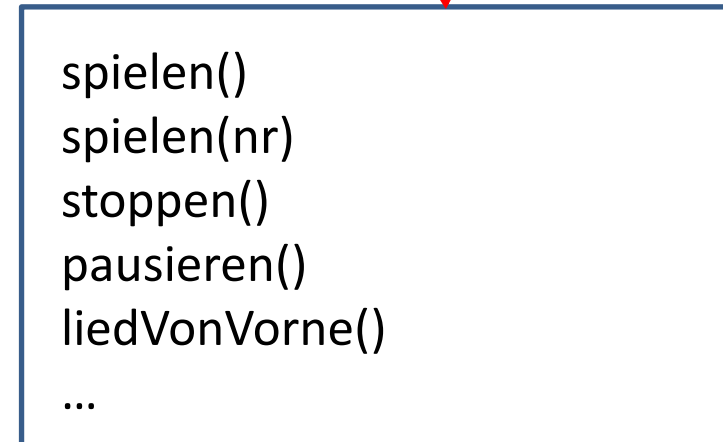
Controller

OK!

`model.pausieren();`



View



Model
mp3-Player

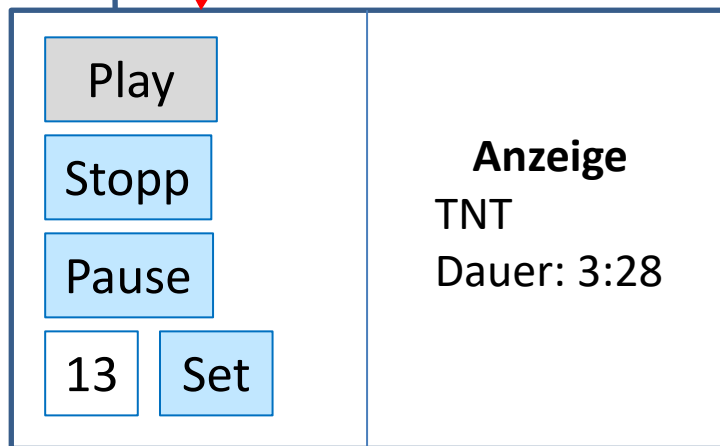
Jemand hat Set
gedrückt. Im Fenster
steht 13!



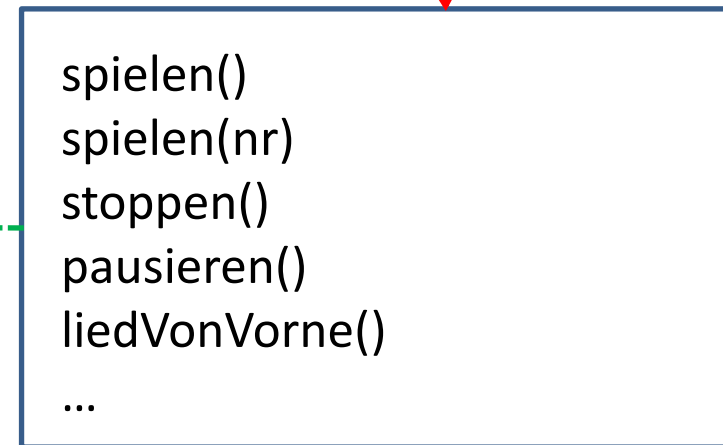
Controller

OK!

`model.spielen(13);`



View



Model
mp3-Player

Ich spiele jetzt "Bad",
dauert 04:13

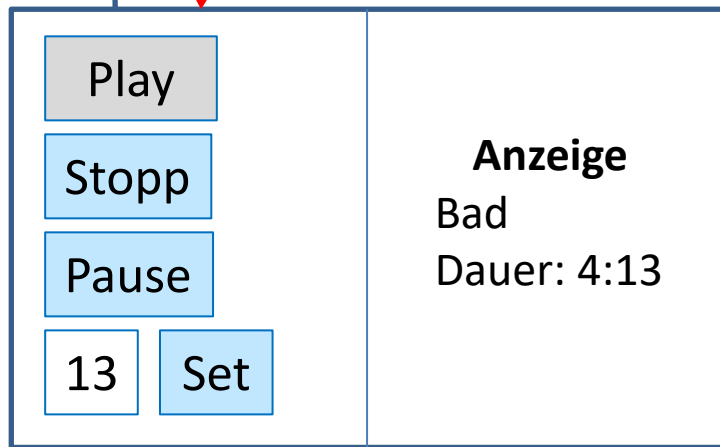
Jemand hat Set
gedrückt. Im Fenster
steht 13!



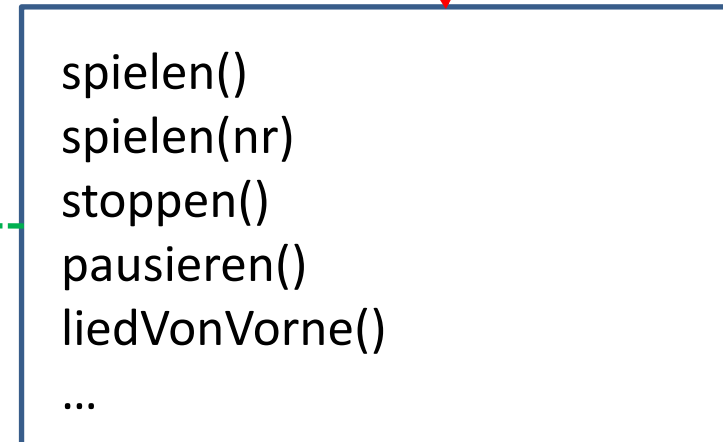
Controller

OK!

`model.spielen(13);`



View



Model
mp3-Player

Ich spiele jetzt "Bad",
dauert 04:13

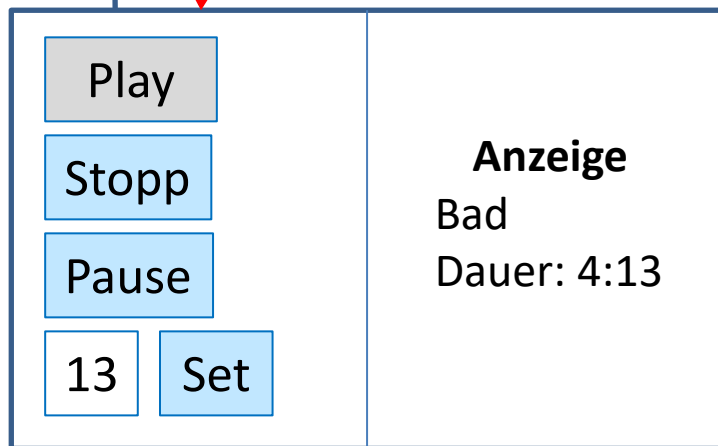
Jemand hat Set
gedrückt. Im Fenster
steht 13!



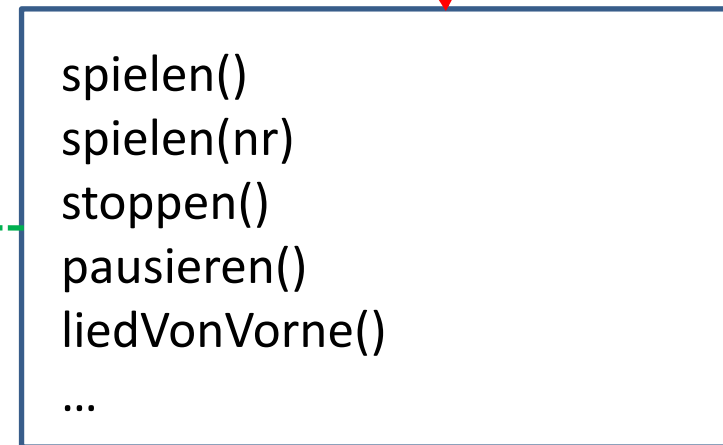
ÜblerController

OK!

`model.spielen(5);`



View



Model
mp3-Player

← `Ich spiele jetzt
"Atemlos",
dauert 03:15`

Jemand hat Set
gedrückt. Im Fenster
steht 13!

ÜblerController

OK!

`model.spielen(5);`

Ich spiele jetzt
"Atemlos",
dauert 03:15

View

Model
mp3-Player

Play

Stopp

Pause

13

Set

Anzeige
Atemlos
Dauer: 3:15

`spielen()`
`spielen(nr)`
`stoppen()`
`pausieren()`
`liedVonVorne()`
...

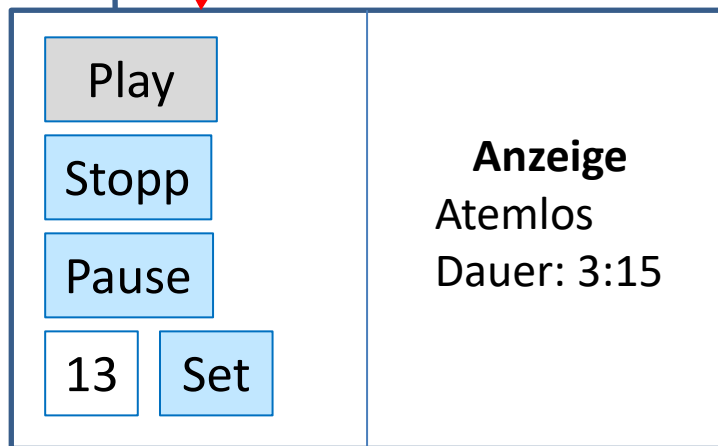
Jemand hat Stopp
gedrückt.



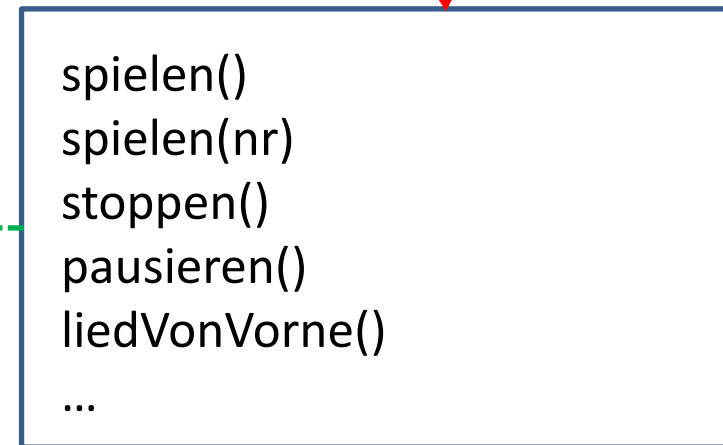
ÜblerController

OK!

model.liedVonVorne();



View



Model
mp3-Player

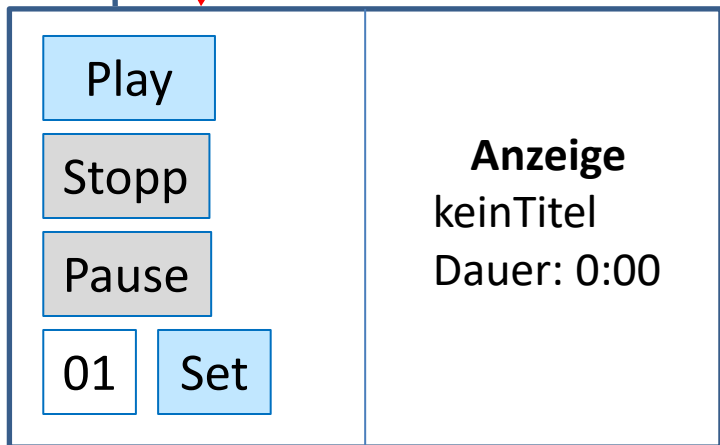
Meine Play-Taste
wurde gedrückt!



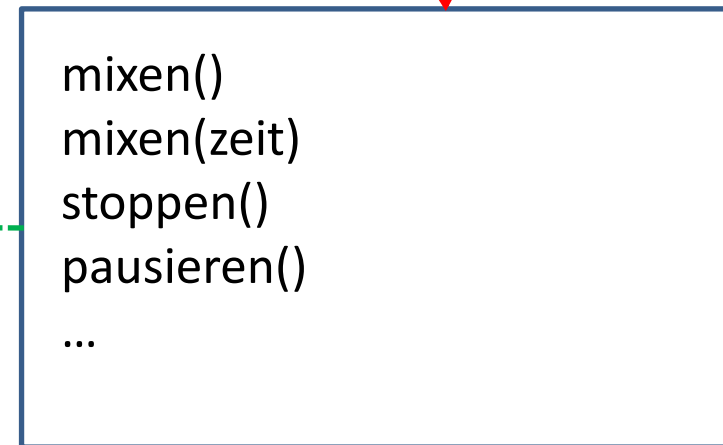
Controller

OK! Schalte die
Play-Taste auf
deaktiviert.
Aktiviere Stopp-
und Pause-Taste

`model.mixen();`

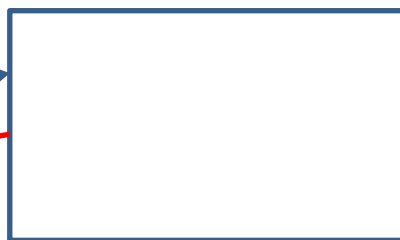


View



Model
Küchenmaschine

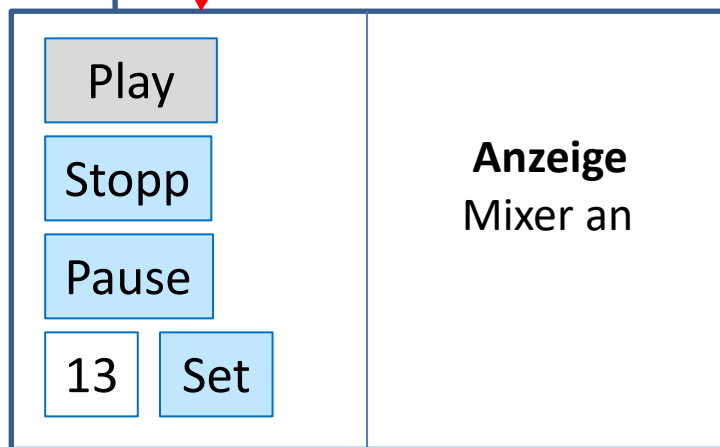
Meine Play-Taste
wurde gedrückt!



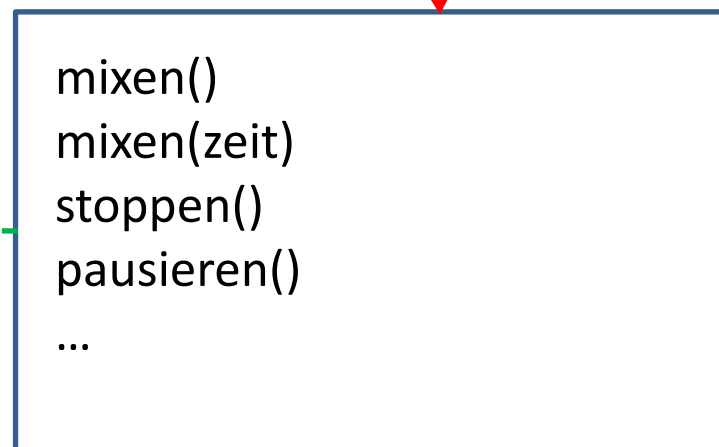
Controller

OK! Schalte die
Play-Taste auf
deaktiviert.
Aktiviere Stopp-
und Pause-Taste

`model.mixen();`



View



Model
Küchenmaschine

Entwurfsmuster Singleton (Einzelnr)

Gelegentlich muss sicher gestellt werden, dass von einer Klasse nur ein einziges Objekt erstellt werden kann:

```
public class NurEiner{  
    private static NurEiner highlander = new NurEiner();  
    private NurEiner() {...}  
    public static NurEiner InstanzGeben(){  
        return highlander;  
    }  
}
```