

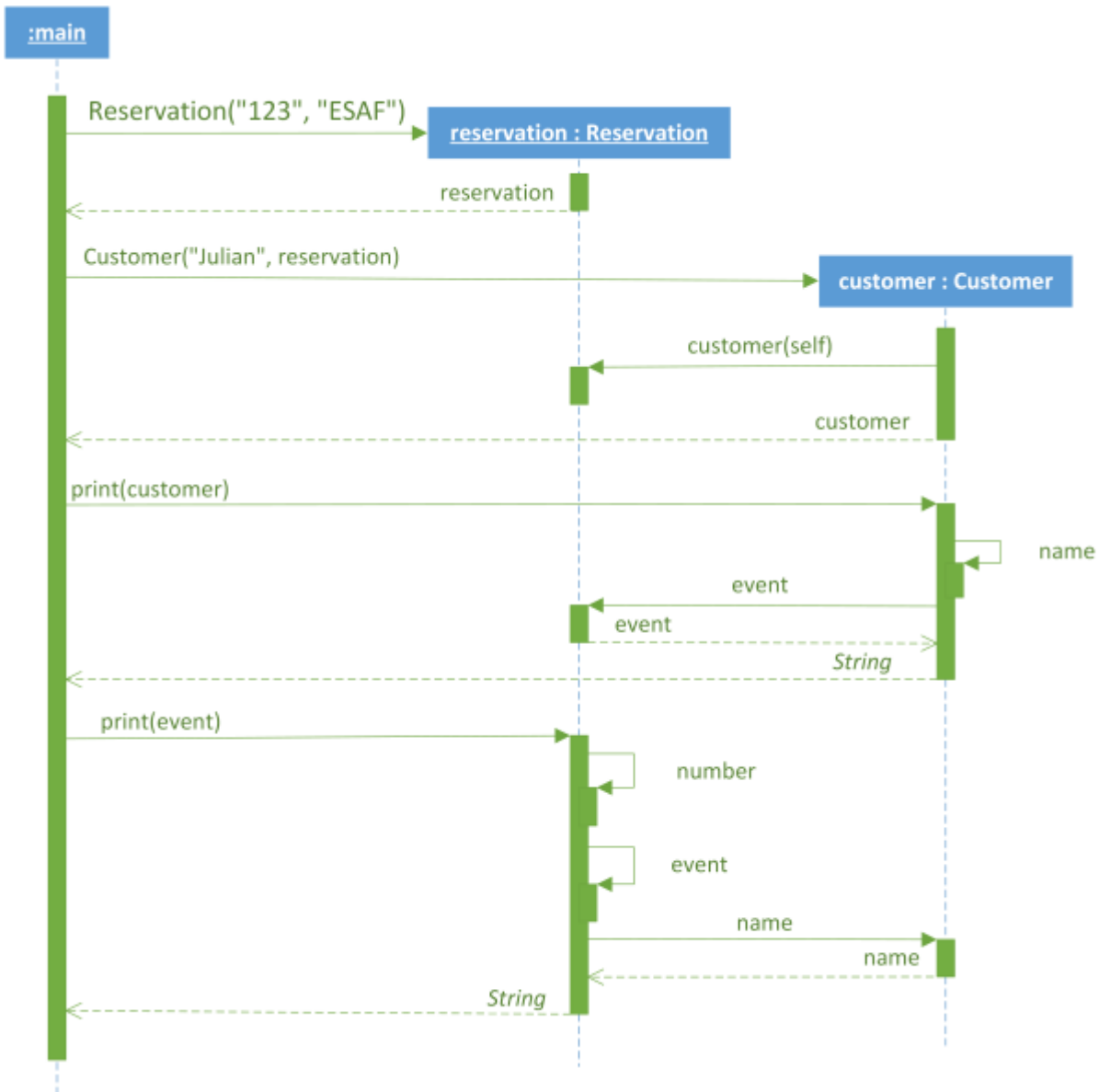
LU08.A02 - Zweiseitige Beziehung



Sie können eine zweiseitige Beziehung (Assoziation) zweier Klassen implementieren.

```
Reservation
- number : String
- event : String
- customer : Customer
+ Reservation(number : String, event : String)
+ <<prop>>number() : String
+ <<prop>>event() : String
+ <<prop>>customer() : Customer
+ <<setter>>customer(the_customer : Customer) : void
+ __str__() : String
```

```
Customer
- name : String
- reservation : Reservation
+ Customer(name : String, reservation : Reservation)
+ <<prop>>name() : String
+ <<prop>>reservation() : Reservation
+ __str__() : String
```



Auftrag

- Implementieren Sie die beiden Klassen `Reservation` und `Customer` gemäss dem Klassendiagramm (bitte in dieser Reihenfolge!).
 - Legen Sie zuerst nur den Programm-Rahmen fest. Das heisst dass Sie die benötigten Klassen und deren Methoden-Rümpfe sowie die zugehörigen Kommentare anschreiben.
 - Das 'Innenleben' der Methoden ergibt sich dann später auf Grund des Programmablaufs, der im Sequenzdiagramm gezeigt wird.
- Implementieren Sie die `main`-Methode sowie den Code der Konstruktoren und Methoden entsprechend dem gezeigten Sequenzdiagramm.
- Das Programm soll eine vergleichbare Ausgabe erzeugen.

```
Julian hat eine Reservation für den Anlass ESAF  
Reservation 123 ESAF für Kunde Julian
```

```
Process finished with exit code 0
```

Hinweis

Es handelt sich bei diesem Beispiel um eine 1:1-Beziehung.

Dauer

20 Minuten

Abgabe

Erfolgt als Push ins GitHub Repository.

⇒ *GitHub Repo für externe Besucher*

GitHub Repository <https://github.com/templates-python/m320-lu08-a02-reservation>

Lernende am BZZ müssen den Link zum GitHub Classroom Assignment verwenden

[M320-LU08](#)



René Probst, bearbeitet durch Marcel Suter

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

https://wiki.bzz.ch/modul/m320_2024/learningunits/lu08/aufgaben/zweiseitigebeziehung

Last update: **2024/09/18 09:42**

