UML

Konrad Raue, Oliver Scholz

7. Januar 2020

Gliederung

1. Nachtrag
2. Grundlagen UML
3. UML zu Java
4. Java zu UML
5. Nächste Woche

Nachtrag

Wrapper Class

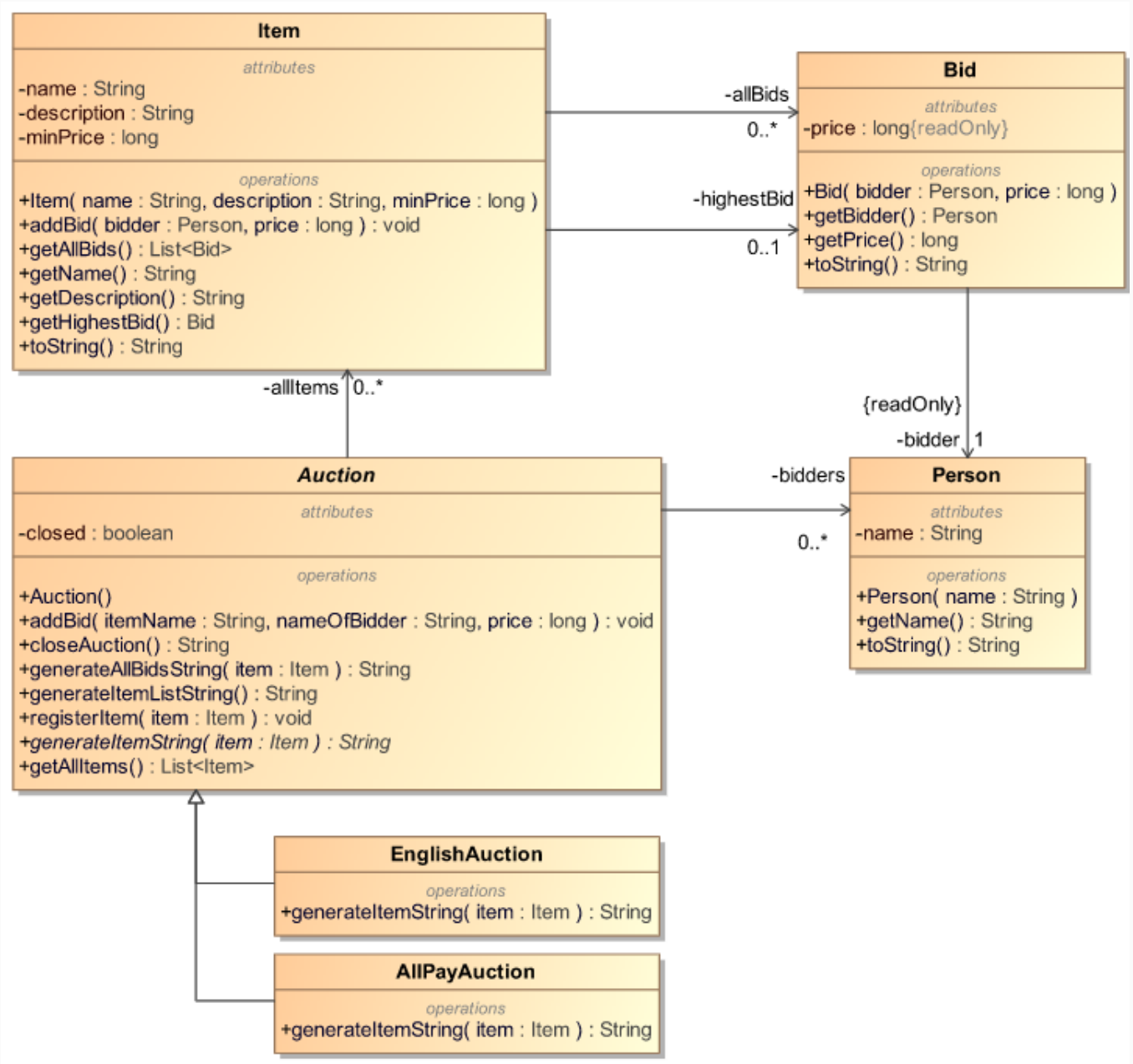
- primitive Datentypen können nicht in Collections (z.B. List) genutzt werden, daher Verwendung von Wrapper Classes
- bieten des Weiteren auch Funktionen an, z.B. Parser
- boolean =>Boolean
- byte =>Byte
- char =>Character
- int =>Integer
- float =>Float
- double =>Double
- long =>Long
- short =>Short

Grundlagen UML

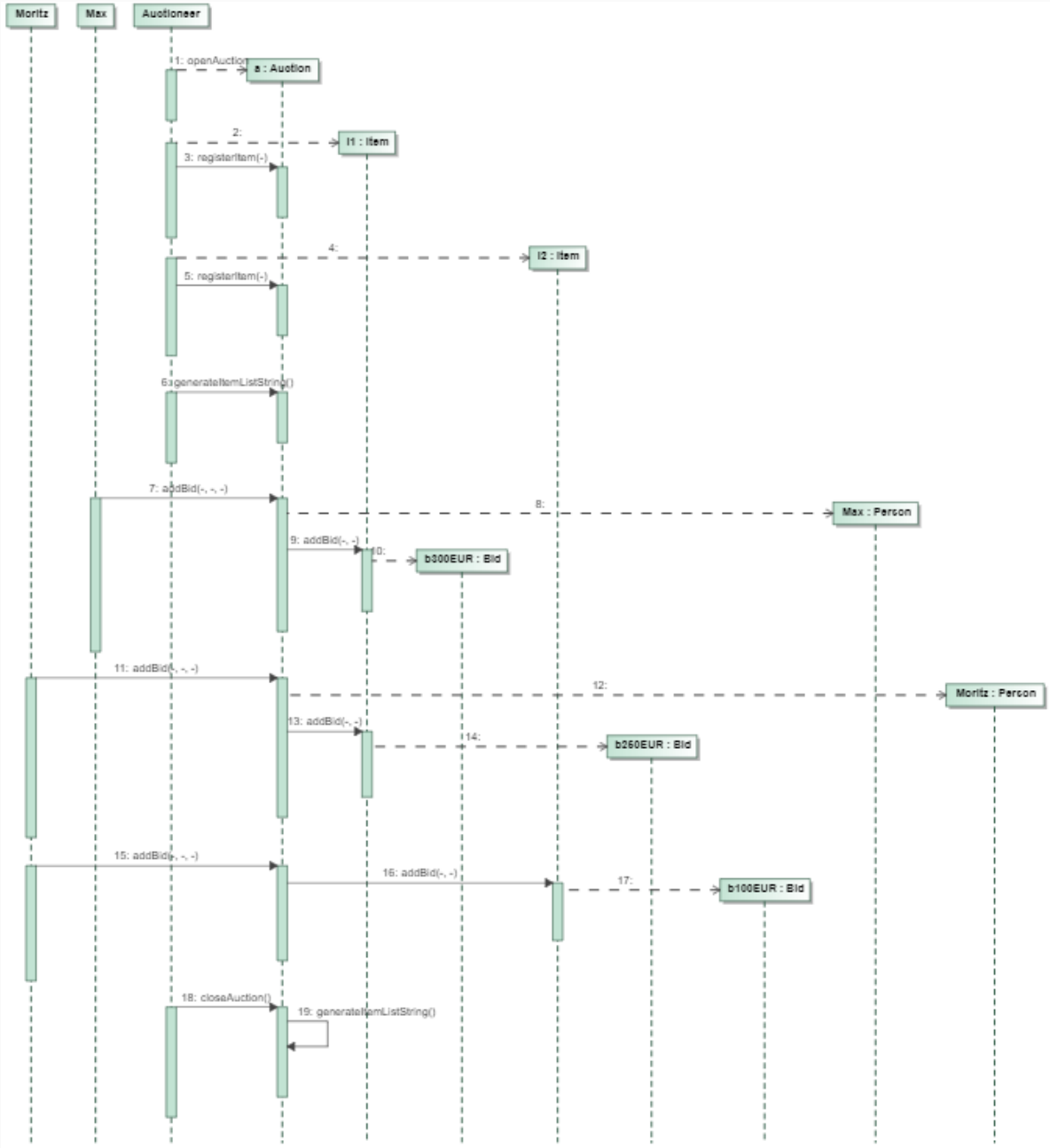
UML

- Unified Modeling Language
- sehr allgemein, nicht unbedingt Java
- grafische Modellierungssprache zum Konstruieren von Software(-Teilen)
- definiert Notation und Semantik (keinen Inhalt der Software)
- wir befassen uns mit Klassendiagrammen, in Softwaretechnologie lernt ihr weitere Diagramm-Typen kennen

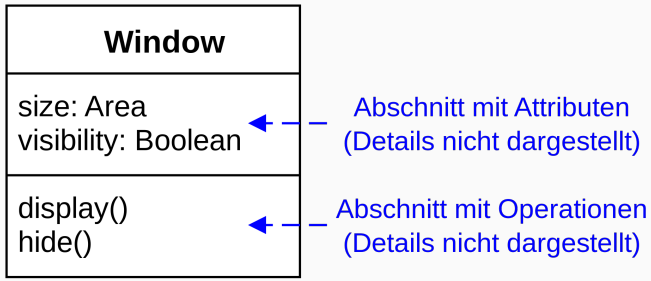
Beispiel aus SWT



Beispiel aus SWT

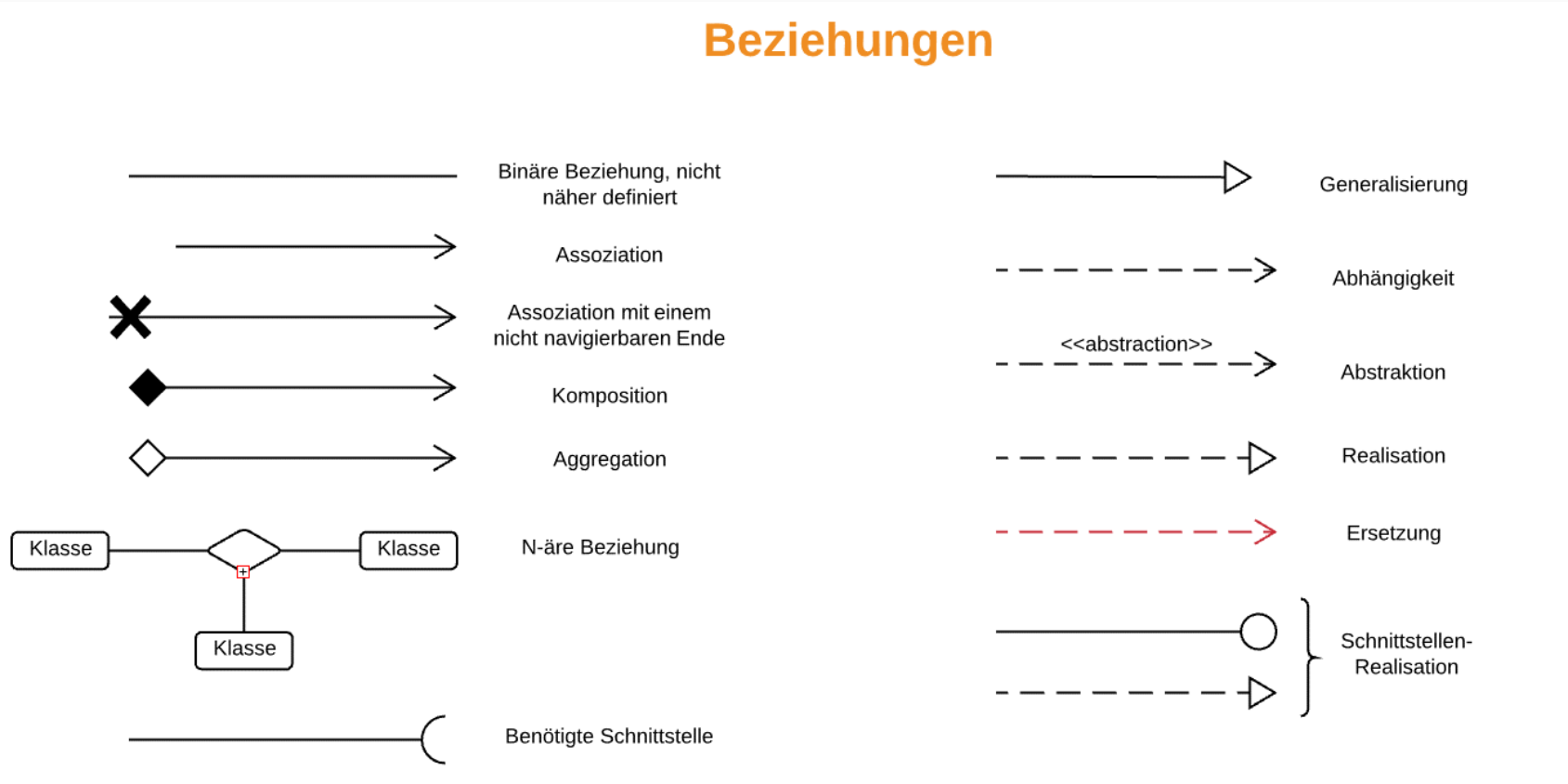


UML zu Java



- Sichtbarkeiten
  - public "+ "
  - private "- "
  - protected "# "
  - package "~ "
- final VARIABLENNAMEN GROß GESCHRIEBEN
- static unterstrichen
- Sonderformen
  - abstract *kursiv* oder <<abstract>>
  - interface grün oder <<interface>>

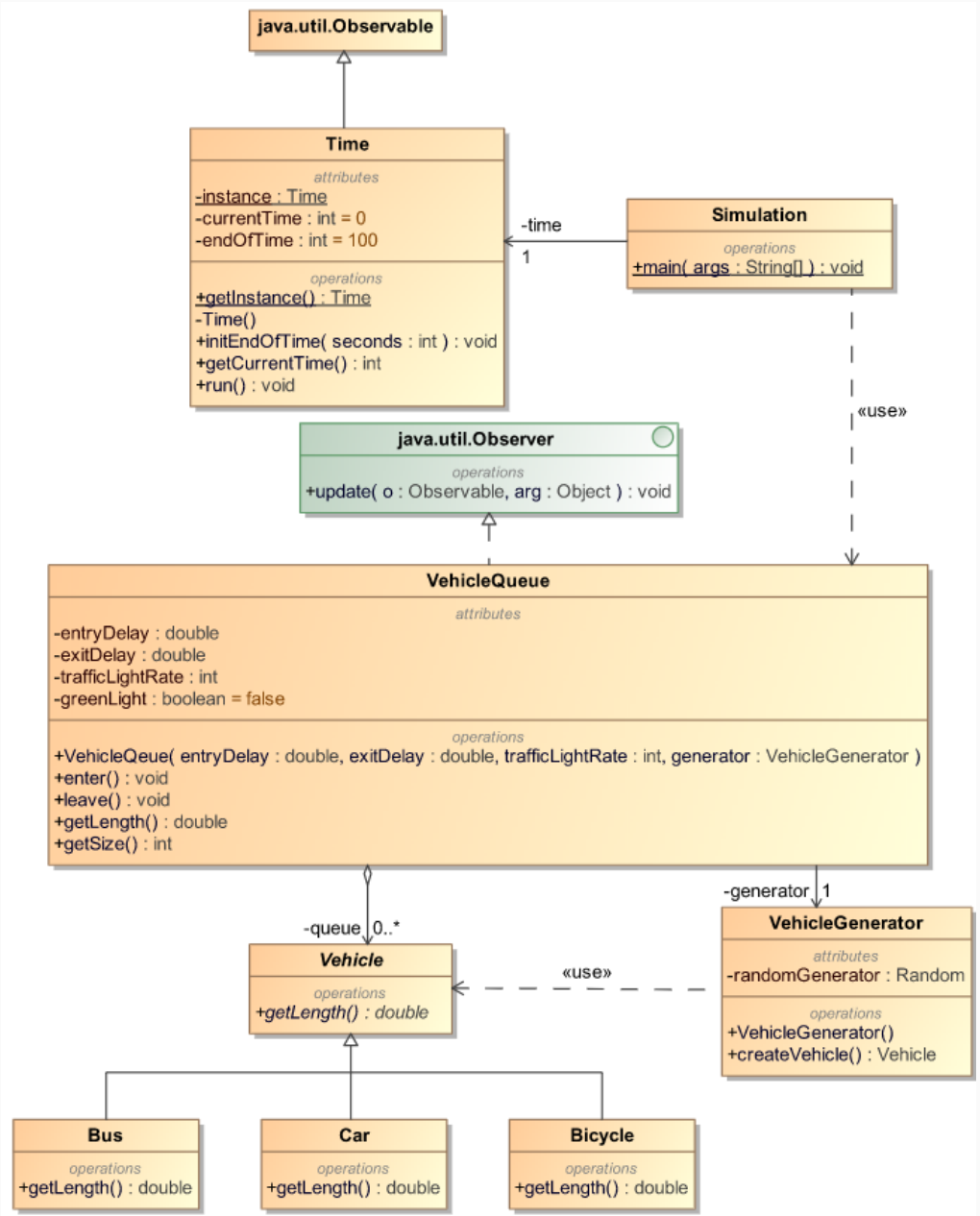
UML zu Java



Leserichtung hier von links (A) nach rechts (B):

- A has B
- A extends B
- A implements B

UML zu Java


java.util.Observable
Time
attributes
-instance : Time
-currentTime : int = 0
-endOfTime : int = 100
operations
+getInstance() : Time
-Time()
+initEndOfTime(seconds : int) : void
+getCurrentTime() : int
+run() : void
-time
1
Simulation
operations
+main(args : String[]) : void
«use»
java.util.Observer
operations
+update(o : Observable, arg : Object) : void
VehicleQueue
attributes
-entryDelay : double
-exitDelay : double
-trafficLightRate : int
-greenLight : boolean = false
operations
+VehicleQeue(entryDelay : double, exitDelay : double, trafficLightRate : int, generator : VehicleGenerator)
+enter() : void
+leave() : void
+getLength() : double
+getSize() : int
-generator 1
-queue 0..*
Vehicle
operations
+getLength() : double
«use»
VehicleGenerator
attributes
-randomGenerator : Random
operations
+VehicleGenerator()
+createVehicle() : Vehicle
Bus
operations
+getLength() : double
Car
operations
+getLength() : double
Bicycle
operations
+getLength() : double

UML zu Java

```
1  public class VehicleQueue implements java.util.Observer{
2      private double entryDelay;
3      private double exitDelay;
4      private int trafficLightRate;
5      private boolean greenLight = false;
6      private VehicleGenerator generator;
7      private java.util.List<Vehicle> queue;
8
9      public VehicleQueue(double entryDelay, double exitDelay, int
       trafficLightRate, VehicleGenerator generator){
10         ...
11     }
12
13     public void enter(){
14         ...
15     }
16     ...
17 }
```

An die vom Observer implementierte Methode update(Observable o, Object arg) denken.

UML zu Java

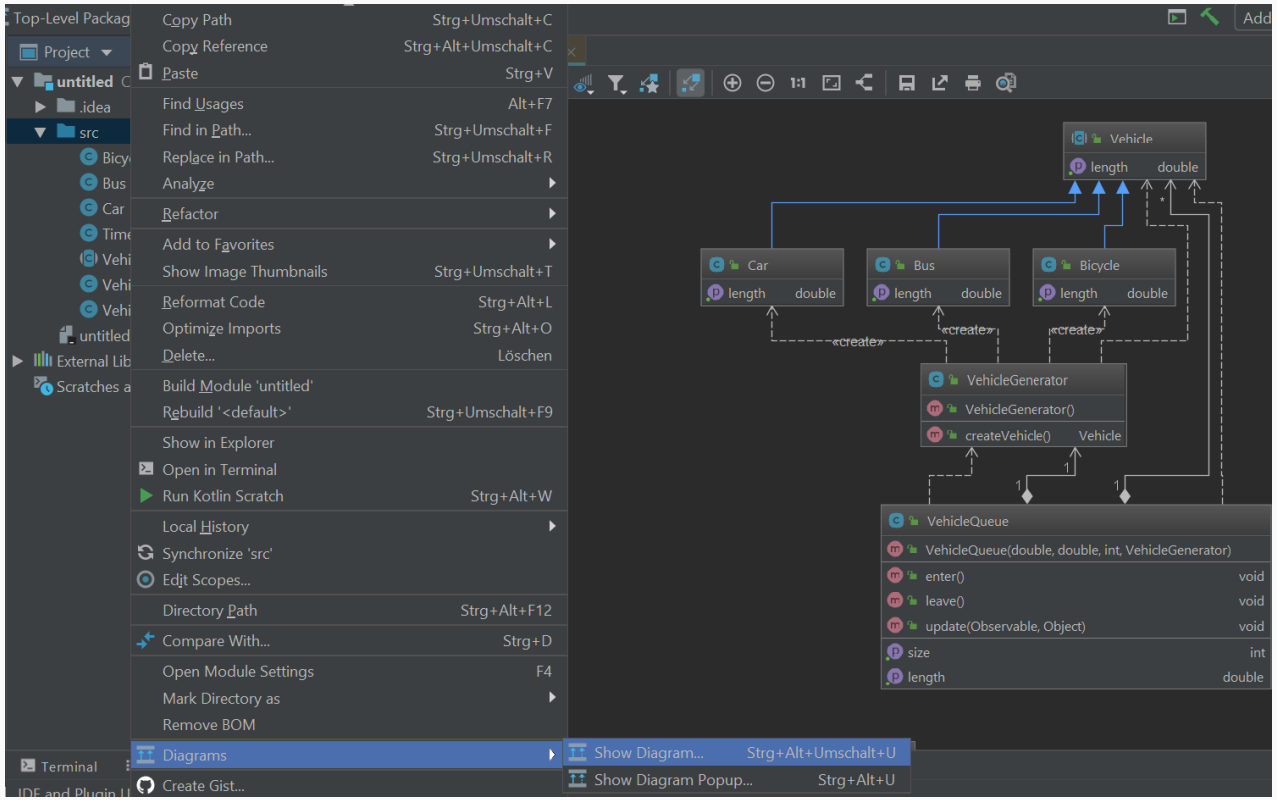
```java
public abstract class Vehicle{
    public abstract double getLength();
}
```

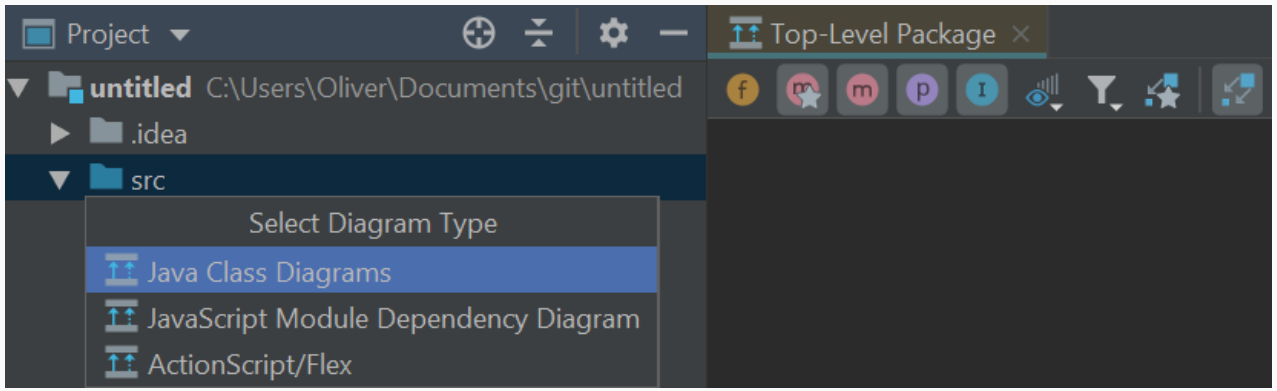
```java
public class Car extends Vehicle{
    public double getLength(){
        return 0.5;
    }
}
```

```java
public class Time extends java.util.Observable{
    private static Time instance;
    private int currentTime = 0;
    private int endOfTime = 100;

    public static Time getInstance(){
        ...
    }

    private Time(){
        ...
    }
    ...
}
```

Java zu UML

Java zu UML


Copy Path Strg+Umschalt+C
Copy Reference Strg+Alt+Umschalt+C
Paste Strg+V
Find Usages Alt+F7
Find in Path... Strg+Umschalt+F
Replace in Path... Strg+Umschalt+R
Analyze
Refactor
Add to Favorites
Show Image Thumbnails Strg+Umschalt+T
Reformat Code Strg+Alt+L
Optimize Imports Strg+Alt+O
Delete... Löschen
Build Module 'untitled'
Rebuild '<default>' Strg+Umschalt+F9
Show in Explorer
Open in Terminal
Run Kotlin Scratch Strg+Alt+W
Local History
Synchronize 'src'
Edit Scopes...
Directory Path Strg+Alt+F12
Compare With... Strg+D
Open Module Settings F4
Mark Directory as
Remove BOM
Diagrams
Create Gist...
Show Diagram... Strg+Alt+Umschalt+U
Show Diagram Popup... Strg+Alt+U
Vehicle
length double
Car
Bus
Bicycle
«create»
VehicleGenerator
VehicleGenerator()
createVehicle() Vehicle
VehicleQueue
VehicleQueue(double, double, int, VehicleGenerator)
enter() void
leave() void
update(Observable, Object) void
size int
length double


Project
Top-Level Package
untitled C:\Users\Oliver\Documents\git\untitled
.idea
src
Select Diagram Type
Java Class Diagrams
JavaScript Module Dependency Diagram
ActionScript/Flex

Nächste Woche

Nächste Woche

- GUI
- I/O