

Ein erstes Programmierbeispiel

openHPI-Java-Team

Hasso-Plattner-Institut







```
1 class HelloPaco{
2
3
4
5 }
```

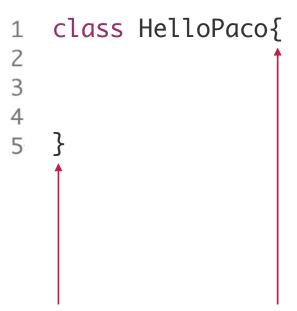
Klasse

- Grundeinheit der objektorientierten Programmierung
- Schlüsselwort class









Geschweifte Klammern

- Strukturieren unser Programm in Code-Blöcke
- Inhalt zwischen { und } wird für den aktuellen Bezeichner ausgewertet (in diesem Fall class)
- Code zwischen { und } (Code-Block) wird nach Konvention eingerückt





Grundgerüst eines Programms (2)

```
class HelloPaco{
   public static void main(String[] args){
        //hier startet unser Programm
   }
}
```

main()-Methode

Gibt es genau einmal pro Programm

Kommentare





```
class HelloPaco{
   public static void main(String[] args){
      // ein einzeiliger Kommentar
      /* ein mehrzeiliger
      Kommentar */
   }
}
```

Kommentare

- Werden bei der Programmausführung ignoriert
- Dienen der Erklärung des Programmes für Entwickler







```
class HelloPaco{
   public static void main(String[] args){
      System.out.println("Hallo Paco");
}
```

System.out.println()

- gibt eine Zeile aus
- Englisch: print line
- Ist eine Methode

Textausgabe (2/2)





```
class HelloPaco{
   public static void main(String[] args){
       System.out.println("Hallo Paco");
   }
}
Ausgabetext
```

Ausgabetext

- Text innerhalb der Klammern wird ausgegeben
- Anführungszeichen sorgen dafür, dass es als Text erkannt wird
- In Java wird Text auch String genannt
- "Hallo Paco" wird Argument der Methode genannt







```
class HelloPaco{
   public static void main(String[] args){
      System.out.println("Hallo Paco");
   }
}
```

Semikolon

Damit endet jede Anweisung







```
class HelloPaco{
   public static void main(String[] args){
       System.out.println("Hallo Paco");
}
```

Ausgabe:

Hallo Paco