

FOP Recap #1



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Organisatorisches, Einführung, FOPBot, Schleifen





Guten Tag / Willkommen!

Das steht heute auf dem Plan



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Organisation

Einführung in Java

Wichtige Operatoren

FOPBot

Das steht heute auf dem Plan



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Organisation

Was ist das Recap?

Wann und wo findet das Recap statt?

Wichtige Links

Einführung in Java

Wichtige Operatoren

FOPBot



- Das Recap ist ein zusätzliches und freiwilliges Angebot
- Keine Anwesenheitspflicht
- Keine Beantwortung von Fragen zu Hausübungen
- Stattdessen:
 - ▣ Wiederholung des Stoffes
 - ▣ Erklärung anhand von Beispielen
 - ▣ Rückfragen und Diskussion **erwünscht**, auch über YouTube und Discord!

Organisation

Wann und wo findet das Recap statt?



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- Jeden Mittwoch um 15:30 Uhr bis ca. 17:00 Uhr
- Verfügbar:
 - ▣ In Präsenz (regulär in S1|03 226)
 - ▣ per Live-Stream auf YouTube
 - ▣ Aufzeichnung nachträglich als YouTube-Video verfügbar

Organisation

Wann und wo findet das Recap statt?



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Abbildung: Recap-Sektion im Moodle

https:

`//moodle.informatik.tu-darmstadt.de/course/view.php?id=1469§ionid=18761`



Check+Prepare Moodle-Kurs:



■ Vorbereitende Übungen

- Sinnvolle Vorbereitung für die Hausübungen
- Gerade wenn keine Vorkenntnisse vorhanden sind

Abbildung: Check+Prepare Moodle-Kurs

<https://moodle.informatik.tu-darmstadt.de/course/view.php?id=1028>



Studierendenguide „Haudy“:

- Installationsanleitungen für Linux, Windows und Mac
- Anleitungen zum Bearbeiten und Abgeben von Hausübungen
- Anleitungen zum Beheben häufig auftretender Probleme

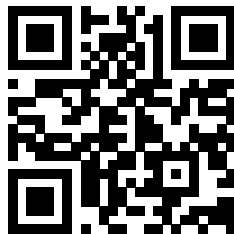


Abbildung: Studierendenguide Haudy
<https://wiki.tudalgo.org/>

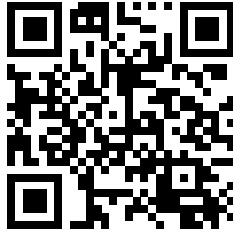


Abbildung: Code-Repository des Recaps

<https://github.com/FOP-2324/FOP-2324-Recap>

Das steht heute auf dem Plan



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Organisation

Einführung in Java

- Setup

- Java vs. FOPBot

- Hello World & Syntax

- Klassen, Methoden und Variablen

- Methodenaufruf

- Schleifen

Wichtige Operatoren

FOPBot



Setup



■ Abgrenzung

- ▣ Was ist Java?
- ▣ Was ist FopBot?

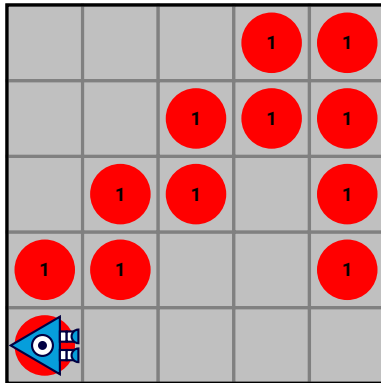


Abbildung: FOPBot

Einführung in Java

Hello World & Syntax



TECHNISCHE
UNIVERSITÄT
DARMSTADT



HelloWorld.java



```
1 public class HelloWorld {  
2     public static void main(String[] args) {  
3         // This is a comment  
4         System.out.println("Hello World!");  
5     }  
6 }
```

Einführung in Java

Hello World & Syntax



TECHNISCHE
UNIVERSITÄT
DARMSTADT



HelloWorld.java



```
1 public class HelloWorld {  
2     public static void main(String[] args) {  
3         // This is a comment  
4         System.out.println("Hello World!");  
5     }  
6 }
```

```
$ Hello World!
```




SomeClass.java



```
1 public class SomeClass {  
2     // Attribute: Gespeicherte Werte  
3     // Methoden: Wiederverwendbarer Code  
4     // ... Anderes ...  
5 }
```



```
1 // ... class ...  
2 public void myMove() {  
3     // Code  
4 }  
5 protected int getMyNumber(int amount, boolean test) {  
6     // Code  
7 }  
8 //
```

Syntax Methodenkopf:

Zugriffsmodifikatoren Rückgabetyp Name (Parameter1, Parameter2, ...)

Einführung in Java

Klassen, Methoden und Variablen – Datentypen



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Name	Typ	Beispiel
boolean	Wahr/Falsch	true, false
int	Ganze Zahl	-5, 69
double	Gleitkommazahl	0.215, 251.998
String	Zeichenkette	"Hallo"



```
1 // ... Methodenkopf ...  
2 // .....  
3 int localVariableA = 5;  
4 localVariableA = 2;  
5 // .....  
6 //
```

- ([Methoden-]Parameter sind quasi spezielle lokale Variablen)



```
1 class Abc {  
2     // .....  
3     public void testCall(int a, boolean b) {  
4         // ! Code !  
5     }  
6     // .....  
7 }
```

```
1 Abc anyName = .....  
2 anyName.testCall(5, true);
```



```
1 while(condition) {  
2     //Run this until condition no longer evaluates to true...  
3 }
```

While-loop



TECHNISCHE
UNIVERSITÄT
DARMSTADT

```
1  int counter = 50;
2
3  while(counter > 5) {
4      counter -= 1;
5  }
6
7  if(counter == 5) {
8      System.out.println("Yes!");
9  }
```

While-loop



TECHNISCHE
UNIVERSITÄT
DARMSTADT

```
1  int counter = 50;
2
3  while(counter > 5) {
4      counter -= 1;
5  }
6
7  if(counter == 5) {
8      System.out.println("Yes!");
9  }
```

\$ Yes!



```
1  for(initialization; condition; increment) {  
2      //Run this until condition no longer evaluates to true...  
3  }
```



```
1  for(int i = 5; i < 5; i = i + 1) {  
2      // .....  
3  }
```

```
1  int i = 5;  
2  for( ; i < 5; ) {  
3      // .....  
4      i = i + 1;  
5  }
```

For-loop



```
1  int i = 5;  
2  for( ; i < 5; ) {  
3      // .....  
4      i = i + 1;  
5  }
```

```
1  int i = 5;  
2  while(i < 5) {  
3      // .....  
4      i = i + 1;  
5  }
```

Das steht heute auf dem Plan



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Organisation

Einführung in Java

Naming Convention

Wichtige Operatoren

FOPBot



Warum benennt man Klassen und Methoden so und nicht anders?



- Schreibweise meistens: **camelCase**
- Unterschiedliche Programmiersprachen können andere Konventionen haben
- Diese Konventionen **unbedingt** einhalten!



```
1 double calculateAverage(double a, double b) {  
2     double sum = a + b;  
3     double average = sum / 2;  
4     return average;  
5 }
```

Best Practices

Unleserlich durch fehlende Leerzeichen



TECHNISCHE
UNIVERSITÄT
DARMSTADT

```
1 double calculateAverage(double a,double b)
2 {
3     double sum=a+b;
4     double average=sum/2;
5     return average;
6 }
```


Best Practices

Unleserlich durch fehlende Leerzeichen



TECHNISCHE
UNIVERSITÄT
DARMSTADT

```
1 double calculateAverage(double a,double b)
2 {
3     double sum=a+b; double average=sum/2; return average;
4 }
```

Best Practices

Fehlende Formatierung, Bruch der Naming Convention, Tippfehler



TECHNISCHE
UNIVERSITÄT
DARMSTADT

```
1 double Durschnitt_Brechnen(double ZalEins, double ZahlZwei) {  
2     double Summe = ZalEins + ZahlZwei;  
3     double Durschnitt = Summe / 2;;  
4     return Durschnitt; }
```

Best Practices

Naming Convention



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Art	Regeln für die Benennung	Beispiele
Klassen	Nomen – PascalCase – Groß	StringBuilder InputStreamReader
Methoden	Verb – camelCase – Klein	getVersion calculateAverage produceDiamond
Variablen	camelCase – Klein	input average message

Das steht heute auf dem Plan



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Organisation

Einführung in Java

Wichtige Operatoren

- Binäre arithmetische Operatoren

- Unäre arithmetische Operatoren

- Vergleichsoperatoren

- Logische Operatoren (Boolean)

FOPBot

Wichtige Operatoren

Binäre arithmetische Operatoren



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Operator	Beispiel	Mit Zuweisung
+	$a + b$	$a += b;$
-	$a - b$	$a -= b;$
/	a / b	$a /= b;$
*	$a * b$	$a *= b;$
%	$a \% b$	$a \% = b;$

Wichtige Operatoren

Unäre arithmetische Operatoren



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Operator	Beispiel	Mit Zuweisung
+	<code>+a</code>	<code>a = +a;</code>
-	<code>-a</code>	<code>a = -a;</code>
++	<code>a++,++a</code>	<code>a++; ++a;</code>
--	<code>a--,--a</code>	<code>a--; --a;</code>

Wichtige Operatoren

Vergleichsoperatoren



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Operator	Beispiel	Erklärung
==	a == b	Gleich?
!=	a != b	Ungleich?
<	a < b	Kleiner als?
<=	a <= b	Kleiner gleich?
>	a > b	Größer als?
>=	a >= b	Größer gleich?

Wichtige Operatoren

Logische Operatoren (Boolean)



TECHNISCHE
UNIVERSITÄT
DARMSTADT

&&	true	false
true	true	false
false	false	false

Und

 	true	false
true	true	true
false	true	false

Oder

!	true	false
	false	true

Negation

Das steht heute auf dem Plan



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Organisation

Einführung in Java

Wichtige Operatoren

FOPBot

Koordinaten der FOPBot-Welt:

- Wie in Mathematischem Koordinatensystem:
 - ▣ 0,0 ist der Ursprung und ist „unten links“
 - ▣ Koordinaten nehmen in Richtung „oben rechts“ zu

(0,2)	(1,2)	(2,2)
(0,1)	(1,1)	(2,1)
(0,0)	(1,0)	(2,0)

Abbildung: Koordinaten einer „FopBot“-Welt



Wichtige Befehle von Robot:

- `move()`
- `turnLeft()`
- `pickCoin()`
- `putCoin()`



Live-Coding!