

Teil

Objektorientierte Programmierung


Unterrichtseinheit 37

Aufzählungstypen (Enums)

Dr. Dietrich Boles


- Motivation
- Enums
- Zusammenfassung

```
class Hamster {  
    final static int NORD = 0;  
    final static int OST  = 1;  
    final static int SUED = 2;  
    final static int WEST = 3;  
  
    Hamster(int r, int s, int blickrichtung, int k) { ... }  
    ...  
}
```



Wert muss zwischen 0 und 3 liegen;
Konstante benutzen: Hamster.OST, ...

```
void main() {  
    Hamster willi = new Hamster(1, 2, Hamster.OST, 4);  
    Hamster paul  = new Hamster(1, 5, 8, 9);  
    ...  
}
```



Laufzeitfehler

- Enums = Aufzählungstypen (ab Java 5.0)
- Beispiel:

```
enum Richtung { NORD, OST, SUED, WEST }
```

```
class Hamster {
```

```
    Hamster(int r, int s, Richtung blickrichtung, int k) { ... }
```

```
    ...
```

```
}
```

```
void main() {
```

```
    Hamster willi = new Hamster(1, 2, Richtung.OST, 4);
```

```
    Hamster paul = new Hamster(1, 5, 8, 9);
```

```
    ...
```

```
}
```

Syntaxfehler

- Schlüsselwort **enum**
- Einführung eines neuen Typs

```
enum Jahreszeit { Fruehling, Sommer, Herbst, Winter }
```

```
public class EnumTest {  
    public static void main(String[] args) {  
        Jahreszeit zeit = Jahreszeit.Sommer;  
        System.out.println("Es ist " + zeit); // Es ist Sommer  
        switch (zeit) {  
            case Fruehling: System.out.println("spring"); break;  
            case Sommer:   System.out.println("summer"); break;  
            case Herbst:    System.out.println("autumn"); break;  
            case Winter:    System.out.println("winter"); break;  
        }  
    }  
}
```

- Enums = Aufzählungstypen
- Sinn und Zweck: Vermeidung bestimmter Laufzeitfehler