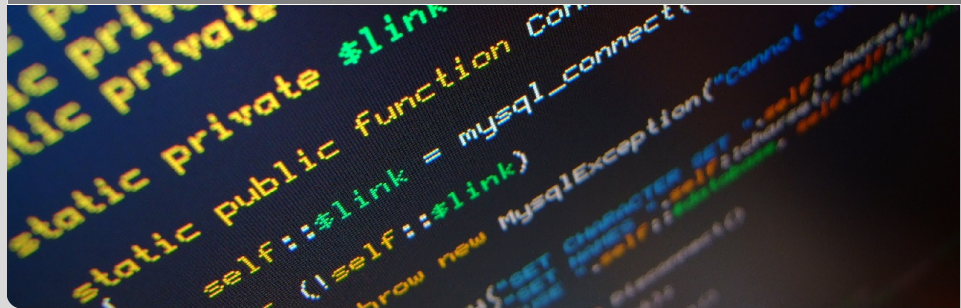


# Tutorium Programmieren

Tut Nr.11: Exceptions, java.util

Michael Friedrich | 21. / 23.11.2013

INSTITUT FÜR THEORETISCHE INFORMATIK



## 1 Exceptions

## 2 java.util

```
public static void main(String[] args) {
    String input = "";
    BufferedReader buf = new BufferedReader( new InputStreamReader(
        System.in));
    while (!input.equals("quit")) {
        try {
            input = buf.readLine();
        } catch (IOException e1) {
            System.out.println("unable to read – shutting down...");
            input = "quit";
        }
        try {
            int i = Integer.parseInt(input);
        } catch (NumberFormatException e) {
            System.out.println("Eingabe war keine Zahl!");
        }
    }
    System.out.println("shutting down...");
}
```

# Was war bei dem Beispiel wichtig?!

# Was war bei dem Beispiel wichtig?!

- Programm stürzt nie unkontrolliert ab

# Was war bei dem Beispiel wichtig?!

- Programm stürzt nie unkontrolliert ab
- User wird über seine Fehler informiert und kann diese verbessern

# Was war bei dem Beispiel wichtig?!

- Programm stürzt nie unkontrolliert ab
- User wird über seine Fehler informiert und kann diese verbessern

# Was war bei dem Beispiel wichtig?!

- Programm stürzt nie unkontrolliert ab
- User wird über seine Fehler informiert und kann diese verbessern

Wir als Programmierer müssen unfähigen User (und Kollegen...) entgegen arbeiten.

Beispiel?



# Was war bei dem Beispiel wichtig?!

- Programm stürzt nie unkontrolliert ab
- User wird über seine Fehler informiert und kann diese verbessern

Wir als Programmierer müssen unfähigen User (und Kollegen...) entgegen arbeiten.

Beispiel? NullPointerException abfangen, falsche Werte geliefert, falsche Formattierung...

Java bietet von sich aus schon sehr viel Funktionalität, z.Bsp LinkedList.

Java bietet von sich aus schon sehr viel Funktionalität, z.Bsp LinkedList.

## Example

weitere eingebaute Datenstrukturen

Java bietet von sich aus schon sehr viel Funktionalität, z.Bsp LinkedList.

## Example

weitere eingebaute Datenstrukturen

- Collections: ungeordneter Pool an Objekten
  - `Collection<Product> products;`

Java bietet von sich aus schon sehr viel Funktionalität, z.Bsp LinkedList.

## Example

weitere eingebaute Datenstrukturen

- Collections: ungeordneter Pool an Objekten
  - `Collection<Product> products;`
- SortedSet: Menge mit totaler Ordnung
  - `SortedSet<Product> products;`

Java bietet von sich aus schon sehr viel Funktionalität, z.Bsp LinkedList.

## Example

weitere eingebaute Datenstrukturen

- Collections: ungeordneter Pool an Objekten
  - `Collection<Product> products;`
- SortedSet: Menge mit totaler Ordnung
  - `SortedSet<Product> products;`
  - Product MUSS hier `Comparable<Product>` implementieren

Java bietet von sich aus schon sehr viel Funktionalität, z.Bsp LinkedList.

## Example

weitere eingebaute Datenstrukturen

- Collections: ungeordneter Pool an Objekten
  - `Collection<Product> products;`
- SortedSet: Menge mit totaler Ordnung
  - `SortedSet<Product> products;`
  - Product MUSS hier `Comparable<Product>` implementieren
- ArrayList: ähnlich LinkedList, aber mit Index
  - `ArrayList<Product> products;`

Java bietet von sich aus schon sehr viel Funktionalität, z.Bsp LinkedList.

## Example

weitere eingebaute Datenstrukturen

- Collections: ungeordneter Pool an Objekten
  - `Collection<Product> products;`
- SortedSet: Menge mit totaler Ordnung
  - `SortedSet<Product> products;`
  - Product MUSS hier `Comparable<Product>` implementieren
- ArrayList: ähnlich LinkedList, aber mit Index
  - `ArrayList<Product> products;`
- Maps: key-value Paare
  - `TreeMap<Product, Customer> orders;`
  - `HashMap<Product, Customer> orders;`

Nehmt die Hinweise auf dem Übungsblatt als Einstiegspunkt.