

## 1 Hello World!

Schreibe nur mit Hilfe eines Texteditors ein Programm, welches **Hello World!** ausgibt. Kompiliere mit `javac` und starte das Programm mit `java`.

## 2 Barplot

Das Programm `Barplot` generiert anhand der Eingabe horizontale Balkendiagramme, die aus einer Beschriftung und einem Balken bestehen.

- Die Formatierung sieht für die Beschriftung 8 Zeichen vor, gefolgt von einem `'|'` und dem maximal 30 Zeichen langen Balken bestehend aus dem Zeichen `'#'`.
- Falls der Balken nicht die vollen 30 Zeichen ausschöpft, wird mit Leerzeichen aufgefüllt.
- An letzter Stelle wird wieder ein `'|'` angehängt, das Diagramm ist also immer 40 Zeichen breit.
- Beim Einlesen der Werte muss zwischen Fließkommazahlen und Ganzzahlen unterschieden werden. Erzeugen Sie zu diesem Zweck einen `Scanner`, der aus dem Eingabestring liest. Nutzen Sie dazu die Methoden `hasNext()`, `hasNextDouble()` und `hasNextInt()` des `Scanners`.
- Ganzzahlen liegen im Intervall `[0, 30]` und entsprechen 1:1 der Anzahl der angezeigten Balken.
- Fließkommazahlen liegen im Intervall `[0, 1]` und beschreiben die prozentuale Länge des Balkens.

### 2.1 Aufgabe

1. Teste dein Programm gegen den Test.
2. Schreibe ein Programm, welches vom User einen String der Form `"label x"` einliest, wobei `label` die Beschriftung darstellt und `x` die Länge des Balkendiagramms. Ansatz:

```
String label = null;
Scanner sc = new Scanner(System.in);
System.out.print("Input: ");
if (sc.hasNext()) {
    label = sc.next();
}
if (sc.hasNext...
```

### 2.2 Methoden

- `public static String repeat(char c, int n)`  
Liefert einen String der Länge `n` zurück, der nur aus dem Zeichen `c` besteht (beispielsweise liefert `repeat('+',4)` den String `"++++"`).

- `public static String drawLabel(String label, int length)`  
Liefert einen `String` zurück der `label` beinhaltet aber genau `n` Zeichen lang ist. Wenn `label` zu lange ist wird es abgeschnitten, wenn `label` zu kurz ist, wird der Rückgabewert mit Leerzeichen aufgefüllt.  
Beispielsweise liefert `drawLabel("abc", 5)` den `String` `"abc "` zurück.
- `static String drawBar(String label, int value)`  
Generiert eine Zeile des Balkendiagramms. `value` bezeichnet dabei die absolute Länge des Balkens. Liegt `value` nicht im Bereich von 0 bis 30 (inklusive), so wird `null` geliefert.
- `static String drawBar(String label, double value)`  
Generiert eine Zeile des Balkendiagramms. `value` bezeichnet die prozentuelle Länge des Balkens. Wird ein Prozentwert als Balken dargestellt muss eine ganzzahlige Balkenlänge berechnet werden. Diese soll durch **Runden** zur nächstgelegenen Ganzzahl entstehen. Liegt `value` nicht im Bereich von 0.0 bis 1.0 (inklusive), so wird `null` geliefert.

### 3 ASCIIArt

Erstelle eine Klasse `ASCIIArt`, welches eine Methode `main()` besitzt. Das Programm soll ein ASCII-Bild zeilenweise aus einer Datei einlesen und überprüfen, ob alle Zeilen gleich lang sind. Nach dem Einlesen wird die Breite und die Höhe des Bildes ausgegeben. Sollte die Eingabe fehlerhaft sein (also nicht alle Zeilen gleich lang), so wird stattdessen eine Fehlermeldung ausgegeben. Lesen aus einer Datei:

```
Scanner fromFile = null;
try {
    fromFile = new Scanner(new File(filename));
} catch (FileNotFoundException e) {
    System.err.println("File not found: " + filename);
    return;
}
while(fromFile.hasNextLine()) {
    String line = fromFile.nextLine();
}
fromFile.close();
```

Erwartete Ausgabe:

```
bild1.ascii ok 41 20
bild2.ascii ok 50 44
bild3.ascii Exception
```