

Functions

```
mul :: (Num a) => a -> a -> a
mul a b = a * b
```

Flow Control

```
if      if cond then expr else expr
case   case expr of
```

Typeclasses

```
Eq      ==, /=
Ord     ==, <=, >=, <, >
Show    show 50.2 -> "50.2"
Read    read "50" :: Int -> 50
Enum    pred, succ
Num     alle numerischen Werte
Integrals  alle Int und Integers Werte
Floating  alle Float und Double Werte

function (Typeclass a) => a -> ...
```

C

By **heribert hase**
cheatography.com/heribert-hase/

Not published yet.
Last updated 16th January, 2020.
Page 1 of 1.

Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
<https://apollopad.com>