

Wird von fast allen Funktionen unterstützt.  
Braucht man um etwas von einer Funktion zurückgeben zu können.

Bisher wurde immer keine Wert von der Funktion zurückgegeben, also es wurde der Rückgabewert 'void' verwendet.

```
void function foo() {
```

```
    // Tut was  
}
```

Wenn wir nun einen Wert von unserer Funktion zurückgeben wollen, geben wir hier statt 'void' einen Datentyp an, bspw. 'int' und geben den Wert innerhalb der Funktion mit dem Schlüsselwort 'return' zurück.

```
int function foo() {
```

```
    int ergebnis = 42 * 1337;  
    return ergebnis;  
}
```

Hierbei gilt zu beachten das der Code der nach dem Code stehen würde nicht mehr ausgeführt wird, ähnlich wie bei break u. continue nur eben für Funktionen.

Falls man hier nun bspw. IF-Verzweigungen nutzt, muss man immer ein return gegeben sein, da sonst ein Fehler entsteht.

```
int function foo() {  
    int ergebnis = 42 * 1337;  
    bool abc = true;  
    if (abc) {  
        return ergebnis;  
    }  
    else {  
        return 0;  
    }  
}
```

Ein besser gewählter Programmierstil wäre es eine 'int ret;' Variable zu nutzen, da man nur auf den Wert dieser Variablen achten muss und nicht jedes mal 'return' schreiben muss.

```
int function foo() {  
    int ergebnis = 42 * 1337;  
    bool abc = true;  
    int ret = 0;  
    if (abc) {  
        ret = ergebnis;  
    }  
    return ret;  
}
```

Nun kann man den Rückgabewert in eine Variable schreiben. Dabei gilt zu beachten, dass die Variable und die Funktion den selben Datentyp haben.

```
int meinInteger = foo();
```