

Wiederholungen (Iterationen)

- Die FOR-Schleife
- Die WHILE-Schleife
- DIE DO-WHILE-Schleife
- DIE FOREACH-Schleife

Die FOR-Schleife

Ist eine "kopfgesteuerte Schleife".

Syntax:

```
for (Zählerinitialisierung; Bedingung; Zählerreinitialisierung)
{
    //Anweisung(en)
}
```

Beispiel:

```
for (int i = 0; i <= 10; i++)
{
    //Anweisung(en)
}
```

Die WHILE-Schleife

Ist auch eine "kopfgesteuerte Schleife".

Wird solange ausgeführt bis die Bedingung erfüllt oder nicht mehr erfüllt ist:

Syntax:

```
While (bedingung)
{
    //Anweisung(en)
}
```

```
while(bedingung == true)
{
    //Anweisung(en)
}
```

```
While(bedingung == false)
{
    //Anweisung(en)
}
```

Beispiel:

```
j = 0;
while (j < 10)
{
    Console.WriteLine(j);
    j++;
}
```

Die DO-WHILE-Schleife

Ist eine "fußgesteuerte Schleife"

Syntax:

```
do {  
  //Anweisung(en)  
} while(bedingung == true)
```

Beispiel:

```
k = 0;  
do  
{  
  Console.WriteLine(k);  
  k++;  
} while (k < 10);
```

Der für die Praxis wichtigste Vorteil der do-while-Schleifen ist wohl der, dass diese Schleife auf jeden Fall einmal durchlaufen wird, da die Schleifenbedingung erst nach dem ersten Durchlauf überprüft wird. do-while-Schleifen sind nachprüfende Schleifen.

Die FOREACH-Schleife

Ist eine besondere Form der Iteration. Dieser Schleifentyp kann nur durch Elemente eines Arrays iterieren.

Syntax:

```
Foreach(typ variablenname in arrayName)
{
    //Anweisung(en)
}
```

Beispiel:

```
string[] myPokemon = {"Glumanda", "Pikachu", "Bisasam", "Schiggy"};
```

```
Foreach(string pokemon in myPokemon)
{
    Console.WriteLine(pokemon);
}
```