

Loopar



Loopar

Används när man vill köra samma kod flera gånger i rad

Exempel:

- Gå igenom en personallista och skriv ut alla namn
- Lägg ihop de 53 första heltalen
- Skriv ut alla primtal mellan 1 och 4000
- mm

loopar

I Java har vi 4 typer av loopar

- while-loopar
- do-loopar
- for-loopar
- foreach-loopar

While-loopen

```
while(villkor){
```

```
    //kod
```

```
}
```

Kommer att köra blocket mellan klamrarna så länge villkoret är sant.

OBS! Om villkoret aldrig blir falskt så fastnar programmet i en oändlig loop!

While-loopen

While-loopen kontrollerar villkoret FÖRE loopen körs.

Är villkoret inte uppfyllt så kommer koden i loopen inte köras alls.

exempel

```
int x = 6;
```

```
while(x>3){
```

```
    System.out.println(x);
```

```
    x -= 1;
```

```
}
```

Vad skrivs ut?

Do-loopen

Do-loopen fungerar likadant men kör alltid kodblocket MINST en gång!

```
do {  
    //kod  
}  
while(vilkor);
```

Villkorskontrollen inträffar inte förrän EFTER att blocket körts!

While

```
int x = 3;
```

```
while (x>3){  
    System.out.println(x);  
    x--;  
}
```

Vad skrivs ut?

Do

```
int x = 3;
```

```
do {  
    System.out.println(x);  
    X--;  
} while (x>3)
```

Vad skrivs ut?

For-loopen

Om vi vet hur många gånger något ska göras är for-loopen vår vän!

```
for(int i=0; i<45; i++){  
    System.out.println(i);  
}
```

Koden kommer att köras 45 gånger (0-44)!

For each loopen

For each loopen är en variant på For loopen som används när man loopar igenom samlingar (dvs Arrayer, Listor etc)

```
for (type name : Collection){  
    //kod här körs för varje name som finns i Collection  
}
```

Vi kikar mer på arrayer och listor senare

For

```
for (int i=0; i<array.length; i++)  
{  
    System.out.println(array[i]);  
}
```

For each

```
for (int i : array)  
{  
    System.out.println(i);  
}
```