

## Array

Arrayer tillåter en datatyp att ta emot mer än ett värde. De kännetecknas av hak-parenteser som sätts ut direkt efter variabelns namn utan något mellanrum. E.g.:

```
string[] namn = new string[3]
```

Detta skapar en array med tre element som man sedan kan kalla på genom att skriva det önskade elementet inom hakparentesen. T.ex.:

```
Console.WriteLine("Person 2  
heter " + namn[1]);
```

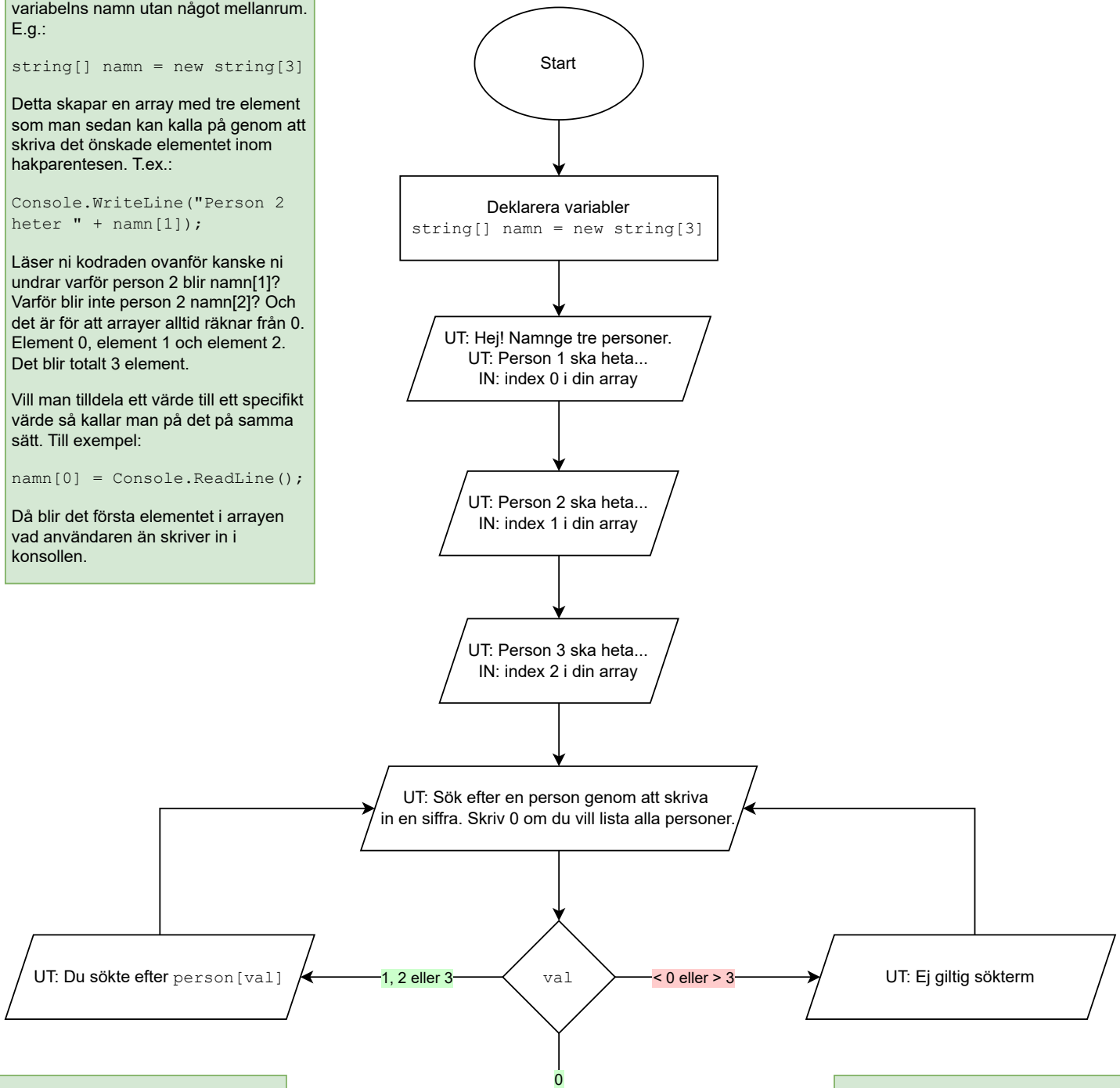
Läser ni kodraden ovanför kanske ni undrar varför person 2 blir `namn[1]`? Varför blir inte person 2 `namn[2]`? Och det är för att arrayer alltid räknar från 0. Element 0, element 1 och element 2. Det blir totalt 3 element.

Vill man tilldela ett värde till ett specifikt värde så kallar man på det på samma sätt. Till exempel:

```
namn[0] = Console.ReadLine();
```

Då blir det första elementet i arrayen vad användaren än skriver in i konsollen.

## For-loopar och arrayer



## For-loop

Till skillnad från while-loop som upprepar ett kodblock så länge ett eller flera villkor är sanna så fungerar for-loopen som så att den upprepar ett kodblock ett bestämt antal gånger.

Man använder loopen såhär:

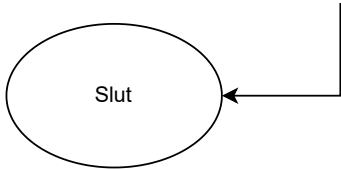
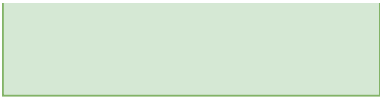
```
for(int i = 0; i <= hur  
många gånger du vill att  
loopen ska loopa; i++){  
    //kod  
}
```

## Egenskaper

Objekt och klasser har vad vi kallar för "egenskaper" som vi kan kalla på för att antingen hämta eller modifiera information. T.ex. en array är en sorts klass, och till denna klass kan vi kalla på en egenskap som heter "Length". Det gör vi genom att skriva namnet på vår array och sedan sätta en punkt och skriva ut "Length". Till exempel:

```
namn.Length
```

Skulle man skriva in den koden i `Console.WriteLine()` så kommer




datorn att skriva ut ett heltal för hur många element som finns i vår array. Det vill säga hur lång den är.

Denna information kan vi sedan använda i vår for-loop genom att skriva såhär:

```
for(int i = 0; i <=
namn.Length; i++) {
    //kod
}
```

`int number`



I variabeln får ett heltal plats

`int numbers = new int[3]`

2	index 0
7	index 1
4	index 2

Variabeln består nu av tre lådor som vi kan lägga unika värden i och vi kallar på dem via deras index