

Övning 2.

Syfte

Denna övning behandlar fält, objektsamlingar (ArrayList, TreeSet, Map), förenklad **for**-sats och iteratorer.

Uppgift 1

Nedan finns definitionen till klassen **Person**.

```
public class Person {  
    private String name;  
    private int birthYear;  
    ...  
    public String getName() {  
        return name;  
    }  
    public int getBirthYear() {  
        return birthYear;  
    }  
    ...  
}
```

a) Skriv en metod

```
public static String[] bornThisYear1(Person[] people, int year)
```

som tar ett fält **people** av **Person**-objekt och ett årtal **year**, och returnerar ett fält som innehåller namnen på de personer i fältet **people** som är födda år **year**.

b) Skriv en metod

```
public static ArrayList<String> bornThisYear2(Person[] people, int year)
```

som gör samma sak som metoden i deluppgift a) förutom att metoden nu skall returnera namnen i en **ArrayList**.

c) Skriv en metod

```
public static ArrayList<String> bornThisYear3(ArrayList<Person> people, int year)
```

som gör samma sak som metoden i deluppgift a) förutom att parametern **person** nu är en **ArrayList** istället för ett fält.

Gör två implementeringar av metoden, dels en som använder en *förenklad for-sats* vid genomlöpnigen av listan **people**, dels en som använder en *iterator* vid genomlöpnigen av listan **people**,

d) Skriv en metod

```
public static void removeNames(ArrayList<Person> people, ArrayList<String> names)
```

som tar en lista **people** och en lista **names**, och tar bort alla objekt i listan **people** vars namn finns med i listan **names**.

Uppgift 2

Anta att vi har en `ArrayList` som innehåller ord och där samma ord kan förekomma flera gånger.

a) Skriv en metod

void uniqueWords(`ArrayList<String>` words)

som skriver ut vilka unika ord som förekommer i listan `words`.

b) Skriv en metod

void nrOfOccurrence(`ArrayList<String>` words)

som skriver ut hur många gånger varje unikt ord förekommer i listan `words`.

c) Skriv en metod

void positionOfOccurrence(`ArrayList<String>` words)

som skriver på vilka positioner varje unikt ord förekommer i listan `words`.