

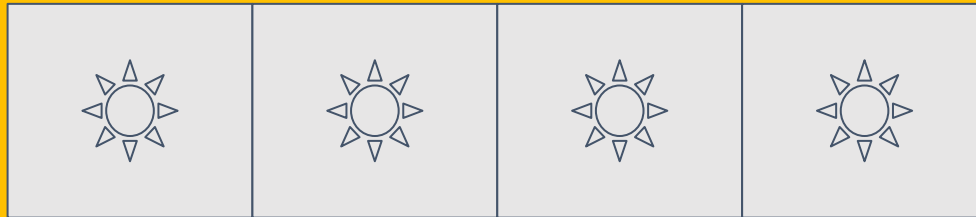
*Skolan för Datavetenskap och kommunikation*

DD1319

Programmeringsteknik

Föreläsning 8

# lista av objekt moduler



# idag

- Läsa från csv-fil
- Lista av objekt
- Moduler

[exempel i Dropbox](#)

## en fil med elevdata: provdata.txt

```
841128-5597;Sumak;Elin;8;9;10;2;9;6;6;4;0;3
991024-7016;André;Erik;0;2;4;9;10;4;4;6;9;2
841002-4155;Edengren;Katarina;10;3;1;1;2;6;1;5;3;6
960327-9196;Viirtanen;Eli;5;7;10;6;0;7;5;8;1;10
850728-7620;Denir;Sarta;9;5;9;10;4;0;8;4;7;7
```

- Filformatet kallas *csv* (comma-separated values).
- Separatoren kan vara antingen " , " eller " ; "
- Excel-filer kan exporteras som *csv*

# plan

Vi vill skriva ett program som läser in alla elever, och skriver ut en sorterad lista, med betyg.

1. Prova att läsa in en rad från filen och dela upp den
2. Skriv en klass för en Elev och testa den
3. Skriv en funktion som läser in rader från filen och skapar objekt som sparas i en lista
4. Gå igenom listan och skriv ut betyg för varje elev

# läsa från filen del 1

```
with open("provdata.csv", "r", encoding="utf8") as infil:  
    for rad in infil:  
        print(rad)
```

## läsa från filen del 2

```
with open("provdata.csv", "r", encoding="utf8") as infil:  
    for rad in infil:  
        rad = rad.strip()  
        data = rad.split(";")  
        print(data)
```

## läsa från filen del 3

```
with open("provdata.csv", "r", encoding="utf8") as infil:
    for rad in infil:
        rad = rad.strip()
        data = rad.split(";")
        pnr = data[0]
        efternamn = data[1]
        förnamn = data[2]
        poäng = data[3:13]
        print(pnr, efternamn, förnamn, poäng)
```



## hjälpfunktion för att dela upp raden

```
def elevdata(rad):  
    """841128-5597;Sumak;Elin;8;9;10;2;9;6;6;4;0;3;"""  
    rad = rad.strip()  
    data = rad.split(";")  
    pnr = data[0]  
    efternamn = data[1]  
    förnamn = data[2]  
    poäng = data[3:13]  
    return pnr, efternamn, förnamn, poäng
```

Anrop:

```
pnr, efternamn, förnamn, poäng = elevdata(rad)
```

## summera en lista

```
def summera(lista):  
    s = 0  
    for tal in lista:  
        s += tal  
    return s
```

```
lista = [1,2,3,4,5,6,7,8,9,10,11,12,13,14]  
summa = summera(lista)  
print(summa)
```

## Elev-klassen: `__init__`

```
class Elev:
```

```
    def __init__(self, pnr, efternamn, förnamn, poäng):  
        self.pnr = pnr  
        self.etternamn = efternamn  
        self.förnamn = förnamn  
        self.poäng = poäng  
        self.summa = summera(self.poäng)  
        self.betyg = self.beräkna_betyg()
```

## Elev-klassen: `__str__`

```
def __str__(self):  
    s = self.efternamn + ", "  
    s += self.förnamn + "  
    s += self.betyg  
    return s
```

## `__lt__` som jämför betyg

```
def __lt__(self, other):  
    if self.betyg < other.betyg:  
        return True  
    else:  
        return False
```

*...eller `__lt__` som jämför namn*

```
def __lt__(self, other):  
    if self.efternamn < other.efternamn:  
        return True  
    elif self.efternamn == other.efternamn:  
        return self.förnamn < other.förnamn  
    else:  
        return False
```

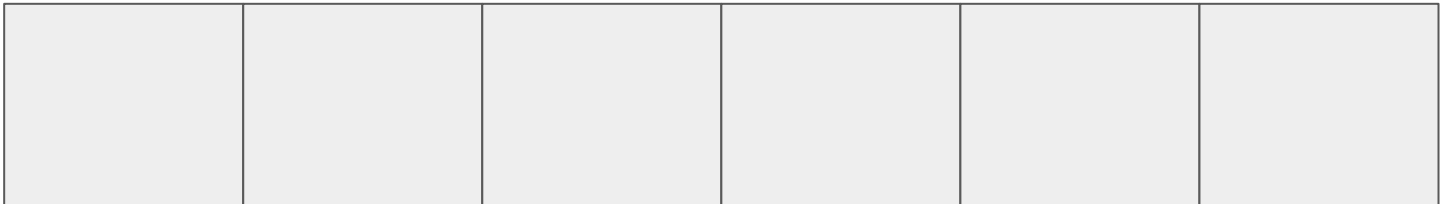
## metod beräkna\_betyg

Betyget ges av summan av poängen. Gränsen för E är 50, D är 60, C är 70, B är 80 och A är 90.

```
def beräkna_betyg(self):
    if self.summa < 50:
        return "F"
    elif self.summa < 60:
        return "E"
    elif self.summa < 70:
        return "D"
    elif self.summa < 80:
        return "C"
    elif self.summa < 90:
        return "B"
    else:
        return "A"
```

## Elev-objekt i lista

```
def elevlista(filnamn):  
    lista = []  
    #infil = open(filnamn, "r", encoding="utf8")  
    with open(filnamn, "r", encoding="utf8") as infil:  
        for rad in infil:  
            rad = rad.strip()  
            nyElev = Elev(rad)  
            lista.append(nyElev)  
    #infil.close()  
    return lista
```





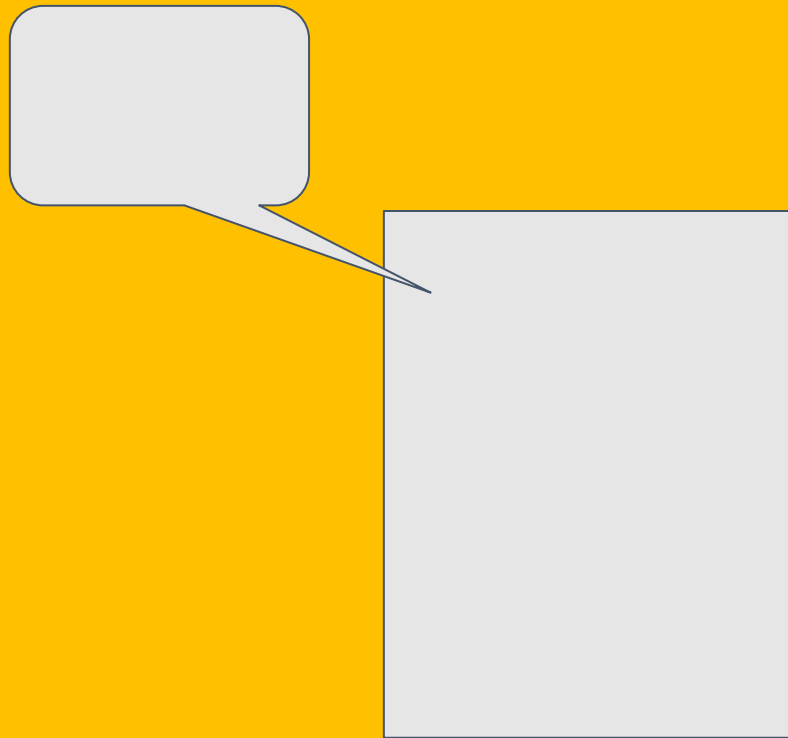
# huvudprogram

```
def main():  
    lista = elevlista("provdata.csv")  
    lista.sort()  
    for elev in lista:  
        print(elev)
```

## urval - skriv bara ut godkända

```
def main():  
    lista = elevlista("provdata.csv")  
    lista.sort()  
    for elev in lista:  
        if elev.betyg >="E":  
            print(elev)
```

moduler



# moduler

- En modul kan hämtas till programmet med en import-sats

```
import random  
x = random.randint(1, 6)
```

- Enstaka delar ur modulen kan importeras:

```
from random import randint  
x = randint(1, 6)
```

- Alla delar ur modulen kan importeras:

```
from random import *  
x = randrange(10)
```

# csv-modulen

Modulen csv underlättar inläsning från csv-filer

```
import csv
```

# csv-modulen

Modulen csv underlättar inläsning från csv-filer

Exempel från dokumentationen:

```
import csv

with open('eggs.csv', newline='') as csvfile:
    spamreader = csv.reader(csvfile, delimiter=' ', quotechar='|')
    for row in spamreader:
        print(row)
```

Variabeln row blir här en *lista* (som om man gjort split)

<https://docs.python.org/3/library/csv.html>

# DictReader i csv-modulen

Med DictReader kan man få en dictionary istället för en lista

```
import csv
with open('names.csv', newline='') as csvfile:
    reader = csv.DictReader(csvfile)
    for row in reader:
        print(row['first_name'], row['last_name'])
```

Variabeln row blir här en *dictionary* med första raden som nycklar

# egna moduler

En egen programfil, t ex *elevmodul.py*, med klasser och funktioner kan importeras på samma sätt

```
import ElevEnklare
```



# testkod i egna moduler

Ofta vill man skriva en main-funktion i modulen för att testa den. Men den ska bara köras om vi anropar modulen som ett program, inte när den importeras.

```
if __name__ == "__main__":  
    main()
```