

Input/Output

Programming 1

2024/2025

Outline

input

Ta emot text

Ta emot heltal

Att ta emot decimaltal

Output

print

sep

end

f-strängar

Input

Att ta emot användarvärden

- För att göra ett program *interaktivt* behöver användaren kunna ge *input*

```
1 namn = input("Skriv ditt namn")  
2 print("Hej", namn)
```

Input

Att ta emot tal

- Testa att skriva följande bit kod, kör den och se vad som händer

```
1 tal = input("Skriv ett tal: ")
2 nytt_tal = tal*2
3 print("Om man dubblar ditt tal får man:", nytt_tal)
```

Input

Att ta emot tal

- ▶ Python antar **alltid** att man ger text
- ▶ För att konvertera från text till ett heltal:

```
1 tal = input("Skriv ett tal: ")  
2 tal = int(tal)
```

- ▶ Nu kan du använda det som ett heltal
- ▶ `int` är en förkortning av *integer* som är engelska för heltal

Input

Att ta emot decimaltal

- ▶ Testa ditt program igen, med förändringen att du *konverterar* till ett heltal
- ▶ Testa nu att skriva in talet 1.2

Input

Att ta emot tal

- ▶ `int` klarar bara av att konvertera text som bara innehåller siffror till tal
- ▶ För att konvertera från text till ett decimaltal (flyttal):

```
1 tal = input("Skriv ett tal: ")
2 tal = float(tal)
```

- ▶ Nu kan du använda det som ett decimaltal

Outline

input

Ta emot text

Ta emot heltal

Att ta emot decimaltal

Output

print

sep

end

f-strängar

Output

print

- ▶ För att skriva ut saker använder du `print()`
- ▶ För att skriva ut flera saker kan du skriva `print(a, b, c)`

Output

sep

- ▶ När du skriver ut flera saker kan du påverka vad som ska hända mellan bitarna.
- ▶ Du kan styra vad kommatecknena ska visas som. Standard är ett mellanslag
- ▶ Du lägger till , `sep="..."` innan den avslutande parantesen

```
1 print("Hej", "på", "dig", sep="!")
```

Output

end

- ▶ Ibland vill du inte att det ska göras en radbrytning efter en `print()`
- ▶ Då kan du lägga till , `end="..."` innan den avslutande parantesen

```
1 print("Hej", end=" ")
2 print("på", end="?")
3 print("dig", end="!")
```

Outline

input

- Ta emot text

- Ta emot heltal

- Att ta emot decimaltal

Output

- print

- sep

- end

f-strängar

f-strängar

Formatera strängar

- ▶ När du vill göra en utskrift som ska innehålla mycket text och variabler kan du använda en f-sträng
- ▶ Det är viktigt att sätta ett `f` framför det första citattecknet.
- ▶ Sen markerar du varje variabel med måsvingar.

```
1 a = 12
2 b = 6
3 print(f"Det finns {b} positiva heltalsdelare till talet {a}")
```