

Föreläsning 5

Programmeringsteknik

DD1310

- Doctest
- Dictionary
- Filhantering

Översikt	<h2>Modulen <code>doctest</code></h2> <ul style="list-style-type: none">• Man kan använda modulen <code>doctest</code> för att testa programmet på ett mer systematiskt sätt.• Så här gör man:<ol style="list-style-type: none">1. Skriver programmets funktioner2. Skriv testfallen i en fil3. Kör filen med testfall4. Vid fel åtgärdar man felen i funktionerna5. Gör om punkt 3 och 4 tills man får rätt
Testning(<code>doctest</code>)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	<h2>Exempel</h2> <p>Betrakta funktionen <code>kant</code> som definierats i filen <code>pythago.py</code>. Funktionen beräknar kanten av en rätvinklig triangel givet hypotenusan och den andra kanten.</p> <pre>def kant(hyp, knt1): a = knt1**2 b = hyp**2 knt2 = (a-b)**0.5 return knt2</pre>
Testning(<code>doctest</code>)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	<h2>Exempel</h2> <p>I filen <code>pythago.doctest</code> skriver vi följande testfall:</p> <pre>>>> import pythago >>> pythago.kant(5, 3) 4.0</pre>
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	<h2>Exempel</h2> <p>Sedan kör vi testfallet med följande kommando:</p> <pre>python3 -m doctest pythago.doctest</pre> <p>Datorn kommer med meddelandet på nästa bild!</p>
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	<h2>Exempel</h2> <pre>***** File "pythago.doctest", line 2, in pythago.doctest Failed example: pythago.kant(5, 3) Expected: 4.0 Got: (2.4492935982947064e-16+4j) ***** 1 items had failures: 1 of 2 in pythago.doctest ***Test Failed*** 1 failures.</pre>
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	<h2>Åtgärda felet</h2> <pre>def kant(hyp, knt1): a = knt1**2 b = hyp**2 knt2 = (a-b) **0.5 return knt2</pre>
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	<h2>Öppna fil</h2> <p>En fil kan öppnas för olika ändamål:</p> <ul style="list-style-type: none"> För läsning: <code>file = open("filename.txt", "r")</code> För skrivning (skriver över befintlig fil) <code>file = open("filename.txt", "w")</code> För tillägg till slutet av filen: <code>file = open("filename.txt", "a")</code> Ytterligare alternativ "r+", "w+", "a+" fungerar som utan plus fast man kan både skriva och läsa.
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	<h2>Läs från öppen fil</h2> <ul style="list-style-type: none"> Hela filen: <code>contents = file.read()</code> Bestämt antal tecken: <code>fiveChars = file.read(5)</code> En rad: <code>oneLine = file.readline()</code> Alla rader till en lista: <code>allLines = file.readlines()</code>
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt
Testning(doctest)
Filhantering
Dictionary (Hashtabell)
Sammanfattning

Skriv till öppen fil och stäng

- Skriv sträng

```
file.write("Hejsan hoppsan")
```
- Elementen i en lista:

```
file.writelines(stringlist)
```
- När du läst eller skrivit klart från/till en fil så stänger du den:

```
file.close()
```

Översikt
Testning(doctest)
Filhantering
Dictionary (Hashtabell)
Sammanfattning

Dictionary (Hashtabell)

- En hashtabell kan ses som en lista som indexen är inte förbestämd utan man väljer ett namn även för indexen:

Nycklar:	5	1	12
Element:	"maj"	"jan"	"dec"

Översikt
Testning(doctest)
Filhantering
Dictionary (Hashtabell)
Sammanfattning

Skapa hashtabell

- Skapa med måsvingar {}, och par av nyckel och värde, enligt nedan:

```
Ht = {1:"jan", 12:"dec", 5:"maj"}
```
- Skapa med dict():

```
ht=dict()
ht[1]="jan"
ht[12]="dec"
ht[5]="maj"
```

Översikt	<h2>Identifiera element</h2> <ul style="list-style-type: none"> Element identifieras med nyckelobjekt, men det måste finnas ett objekt lagrat med given nyckel: <pre>print(ht[1]) print(ht[5]) print(ht[15])</pre> <p>Vad skrivs ut av ovanstående?</p> <p>jan maj</p> <p>ht[15] orsakar att programmet kraschar!</p>
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	<h2>Kontrollerad åtkomst</h2> <ul style="list-style-type: none"> Alternativ 1: Kontrollera förekomst innan åtkomst. <pre>if 15 in ht: x = ht[15]</pre> Alternativ 2: Använd get() med default-värde: <pre>x = ht.get(15, "FINNS EJ!") y = ht.get(15)</pre> <p>Vilka värden tilldelas <code>x</code> och <code>y</code> om 15 inte finns i tabellen?</p> <p><code>x</code> blir "FINNS EJ!" <code>y</code> blir None</p>
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	<h2>Ytterligare operationer</h2> <ul style="list-style-type: none"> Antalet element i tabellen: <pre>len(ht)</pre> Ta bort element: <pre>del ht[12]</pre> Kopiera hela tabellen: <pre>htcopy = ht.copy()</pre>
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Översikt	
Testning(doctest)	
Filhantering	
Dictionary (Hashtabell)	
Sammanfattning	

Sammanfattning

- Dictionary, är väldigt lämplig för snabba sökningar
- Filer kan öppnas för olika ändamål, läsning, skrivning och tilläggning av data
- Använd doctest för alla dina funktioner (dock inte för de som har med input och print) för att göra en systematisk testning av ditt program.
