



python

```
10 from watson.common.contextmanagers import suppress
11
12
13 ACCEPTABLE_RETURN_TYPES = (str, int, float, bool)
14
15
16 class Base(ContainerAware, metaclass=abc.ABCMeta):
17     """The base class for all controllers.
18     Attributes:
19         __action__ (string): The last action that was
20         """
21
22     def execute(self, **kwargs):
23         method = self.get_execute_method(**kwargs)
24         self.__action__ = method
25         return method(**kwargs) or {}
26
27     def get_method(self, **kwargs):
```

Vad är Python?

- Skriptspråk först släppt 1991
- **Interpreter** – Compiler
- Kod oftast lätt att förstå
- Många resurser och bibliotek tillgängliga att ladda ner
- Används till allt möjligt
 - "Python är det näst bästa programmeringsspråket för alla ändamål"
 - **Jack-of-all-trades**
 - Web-Forskning-Automatisering-AI

Varför ska du lära dig Python?

- Bra hobby
- Bra i arbetslivet
- Bra start i din programmeringsresa
- Vad vill du göra?

Variabler

- En variabel är något som kan anta olika värden.
- Det finns fler olika typer av variabler i Python, som skapas på olika sätt.
- Variabler som deklarerar: `namn = värde` och kan innehålla ett värde.
 - **Strings:** Text, deklarerar: `namn = "Det här är en string"`
 - **Integers:** Hela siffror, deklarerar: `namn = 33`
 - **Float:** Tal med decimaler, deklarerar: `namn = 23.4`
 - **Boolean:** Binära variabler som är 0 eller 1, deklarerar: `namn = False/0`
- Variabler som deklarerar lite annorlunda och kan innehålla många värden.
 - **Lists:** Listor med flera variabler eller objekt i, deklarerar: `namn = ["Grej", 20, 8.0]`
 - **Tuples:** Som lists men kan inte ändras, deklarerar: `namn = (8, 9, 16.2, "j")`
 - **Dictionaries:** Listor med en key och ett värde. Värdet kan hämtas genom att specificera keyn, deklarerar: `namn = {"key": "value", "John": "Smith", "Stefan": "Berg"}`

Variablar

```
1 string_var = "This is a string"
2 integer_var = 23
3 float_var = 43.0
4 bool_var = True
5
6 print("string_var is", type(string_var))
7 print("integer_var is", type(integer_var))
8 print("float_var is", type(float_var))
9 print("bool_var is", type(bool_var))
```

```
string_var is <class 'str'>
integer_var is <class 'int'>
float_var is <class 'float'>
bool_var is <class 'bool'>
```

Variabler

```
1 list_var = ["Äpple", "Bannan", "Apelsin"]
2
3 print("1", list_var)
4 print("2", list_var[0])
5 print("3", list_var[1:])
6 print("4", list_var[0:2])
7
8 list_var.append("Gurka")
9 print("5", list_var)
10
11 list_var_2 = ["Annas", "Tomat"]
12 list_var_3 = list_var + list_var_2
13 print("6", list_var_3)
```

Kör: Test.py

```
1 ['Äpple', 'Bannan', 'Apelsin']
2 Äpple
3 ['Bannan', 'Apelsin']
4 ['Äpple', 'Bannan']
5 ['Äpple', 'Bannan', 'Apelsin', 'Gurka']
6 ['Äpple', 'Bannan', 'Apelsin', 'Gurka', 'Annas', 'Tomat']
```

Funktioner

```
1 # len() funktion som ger längden på en string
2 print(len("100-tal"))
3
4 # .upper() funktion som gör om en string till stora bokstäver
5 print("små bokstäver".upper())
6
7 # int() funktioner som gör om en string till en int
8 number = "100"
9 print(number, type(number))
10 number = int(number)
11 print(number, type(number))
```

Kör: Test.py

```
7
SMÅ BOKSTÄVER
100 <class 'str'>
100 <class 'int'>
```

Operationer

```
1 if name == "Edvin":
2     print("Du skrev det här programmet")
3 elif name == "Lucas":
4     print("Du är ledare på den här kursen")
5 else:
6     print("Dig känner jag inte")
7
8
9 x = 5 # Nu är x lika med 5
10
11 if x == 5: # Nu kollar vi om x är lika med 5
12     pass
13
14 x += 1 # Nu är x lika med sitt tidigare tal (5) plus 1.
15 x -= 1 # Nu är x lika med sitt tidigare tal (5) minus 1.
16
17 # Större/Minder än fungerar ungefär som i matten >, <, >=, <=
```


Operationer

```
1 age = 25
2
3 if age > 17:
4     print("Du är vuxen")
5 # Kommer printa då 25 är större än 17
6
7 if age <= 14:
8     print("Du får inte ta moppekörkort")
9 # Kommer inte printa då 25 inte är mindre eller lika med 14
10
11 if age % 5 == 0:
12     print("Din ålder kan delas på 5")
13 # Kommer printas då 25 mod 5 = 0 (mod är resten när man dividerar med ett tal (8 mod 5 = 3 då 5 + 3 = 8))
14
15 frukt = ["äpple", "banan", "melon", "apelsin"]
16 exempel = "chocklad"
17
18 if exempel in frukt:
19     print(exempel, "är gott i en fruktsallad")
20 elif exempel not in frukt:
21     print(exempel, "är nog inte så nyttigt")
22 # Här kommer andra printas, "chocklad är nog inte så nyttigt", då chocklad inte finns i listan frukt
23
24 if age != exempel:
25     print("Inte samma variabel")
26 # Kommer printas då chocklad inte är samma sak som 25
27
28 age1 = 25
29 age2 = "25"
30
31 if age1 == age2:
32     print("Samma ålder!")
33 # Kommer inte printas då stringen "25" inte är samma sak som integern 25
```

While loops

```
1 while(True):
2     print("Jag skriver det här förevigt")
3
4 numb = 0
5 while(numb <= 10):
6     numb += 1
7     print("numb = ", numb)
8
9 while(button.SPACEBAR_PRESSED):
10     jump()
11
12 while(x > 100):
13     # massa kod som ökar och minskar x i olika omgångar
14     if y < 10:
15         break
16     # massa mer kod som ökar och minskar x i olika omgångar
```

For loops

```
1 for x in range(0, 6):  
2     print(x)
```

Kör: Test.py

0
1
2
3
4
5

For loops

```
1 for y in range(10, 300, 50):  
2     print("y", y)  
3  
4 for m in range(8, 0, -1):  
5     print("m", m)
```

Kör: Test.py

```
y 10  
y 60  
y 110  
y 160  
y 210  
y 260  
m 8  
m 7  
m 6  
m 5  
m 4  
m 3  
m 2  
m 1
```

For loops

```
1 colors = ["Blå", "Röd", "Gul", "Grön"]  
2  
3 for col in colors:  
4     print("Du kan måla din bil", col)
```

Kör: Test.py

```
Du kan måla din bil Blå  
Du kan måla din bil Röd  
Du kan måla din bil Gul  
Du kan måla din bil Grön  
>>> |
```

For loops

```
1 for obj in ("a", "b", "k", 5, "edvin"):
2     print("Read", obj)
```

Kör: Test.py

Read a

Read b

Read k

Read 5

Read edvin

Vad nu?

- Skriptspråk först släppt 1991
- **Interpreter** – Compiler
- Kod oftast lätt att förstå
- Många resurser och bibliotek tillgängliga att ladda ner
- Används till allt möjligt
 - "Python är det näst bästa programmeringsspråket för alla ändamål"
 - **Jack-of-all-trades**
 - Web-Forskning-Automatisering-AI