

Flik 2: Installation och utvecklingsmiljöer



Interaktiv python

Start av python-tolken

```
$ python3
Python 3.7.3 (default, Aug 20 2019, 17:04:43)
[GCC 8.3.0] on linux
Type "help", "copyright" or "license" for more information.
>>> my_string = 'Hello World!'
>>> print(my_string)
Hello World!
>>>
```

help

```
>>> help(print)
Help on built-in function print in module builtins:

print(...)
    print(value, ..., sep=' ', end='\n', file=sys.stdout)

    Prints the values to a stream, sys.stdout by default.
    ...
```

Python skript/program

- Valfri editor som kan spara i ren text
 - Vid icke-ASCII, lägg till i början: `# -*- coding: utf-8 -*-`
 - I UNIX har man ibland som första rad: `#!/usr/bin/python3`
- Exekvera genom att köra Pythontolken med filnamnet som argument.
- Eller kör inifrån en IDE (utvecklingsplattform), t.ex.
 - Idle
 - PyCharm
 - Visual Studio Code

Installera Python

- Unix
 - Använd installationsmetod för respektive plattform (yum, yast, pkg_add, portage, HomeBrew)
 - Ladda ner källkoden från python.org och kompilera själv. Inte speciellt svårt...
 - Windows och Mac
 - Ladda ner förkompilerade binärer från python.org
 - Helt OK att bygga python från källkod på Mac OS X
 - Apple förinstallerar (i skrivande stund) Python 2
- `$ brew install python3`

Tilläggsmoduler

- Man kan installera tilläggsmoduler med pip (eller pip3)
 - pip3 install aiohttp
- Moduler kan ofta också installeras via systemets pakethanterare.
- Webbplats för moduler: <https://pypi.org>
- OBS! Det kan bli kladdigt att installera moduler "ad hoc", och eventuellt krävs administratörsrättigheter. Ett rekommenderat tillvägagångssätt är att skapa en "virtuell Pythonmiljö" och installera modulerna i den.

```
$ python3 -m venv labbenv
$ cd labbenv
$ source bin/activate
(labbenv) $ pip install aiohttp
```

Utvecklingsmiljöer

- Open Source IDE
 - Eclipse + PyDev, <http://pydev.sf.net>
 - **PyCharm**, <https://www.jetbrains.com/pycharm/>
- Editorer med textfärgning, indentering etc
 - notepad++, <http://notepad-plus.sourceforge.net>
 - **IDLE** (följer med windows-distributionen)
 - Emacs, <http://www.python.org/emacs>
 - Vi/vim, <http://vim.sourceforge.net/scripts/index.php>
- Kommersiella IDE (\$\$)
 - Wing IDE, <http://wingware.com>
 - **Komodo**, http://www.activestate.com/products/komodo_ide
- Övrigt
 - VisualStudio, <https://code.visualstudio.com>
 - Netbeans + Jython med Coyote, <http://coyote.dev.java.net>
- Se vidare
 - <http://wiki.python.org/moin/IntegratedDevelopmentEnvironments>
 - <http://wiki.python.org/moin/PythonEditors>

Python-varianter och dialekter

- CPython
 - "Vanliga" pythontolken i C
 - Kompilerar till bytecode
 - Använder ett globalt lås (GIL)
- Jython, <http://www.jython.org>, (<http://jythonbook.com>)
 - Skriven i Java
 - Körs i JVM
 - Kan importera javaklasser (`from java.util import Random`)
 - Kompilerar pythonkod till javaklasser med `jythonc`

```
C:\>jython
```

```
Jython 2.2b2 on java1.4.1
```

```
Type "copyright", "credits" or "license" for more information.
```

```
>>> from java.util import Random
```

```
>>> a=Random()
```

```
>>> a.nextFloat()
```

```
0.1187514066696167
```

Python-varianter och dialekter

- PyPy – snabbare (använder JIT)
- IronPython, <http://ironpython.net/>
 - Skriven i C# av Microsoft
 - Stöd för Python 3 är på gång(?)
 - .NET- och ASP.NET-integration
 - Kan importera klasser från .NET-ramverket. Jmf Jython ovan

```
C:\>ipy
```

```
IronPython 1.1 (1.1) on .NET 2.0.50727.42
```

```
Copyright (c) Microsoft Corporation. All rights reserved.
```

```
>>> import System
```

```
>>> class MyClass(System.Object):
```

```
>>> ...
```


Övning

■ 2.1 Starta en interaktiv Pythontolk

- via "Python Console" i PyCharm, eller
- i en terminal ("kommandoprompten" eller GIT-bash)

Sätt a till 5, därefter b till 3 och skriv sedan ut a+b.

```
>>> a = 5
>>> b = 3
>>> print(a + b)
8
>>>
```

■ 2.2 Starta PyCharm (eller någon editor med Pythonstöd).

- Skapa ett nytt projekt, t.ex. med namnet "labb".
- Skapa en ny Python-fil och skriv in kommandona ovan i filen.
- Exekvera (Run).

■ 2.3 Kör programmet från förra övningen i debuggern.

- Börja med att sätta en brytpunkt vid första raden (klicka bredvid radnumret).
- Välj "Debug", välj upprepade gånger "Step Over" (eller F8).

■ 2.4 Stäng nuvarande projekt, checka ut kursens källkod från Git.