

UCSE DAR

Sistemas Operativos I

2014

Introducción

¿Qué es Python?

Python es un lenguaje de programación poderoso y fácil de aprender. Sus características lo hacen un lenguaje ideal para scripting y desarrollo rápido de aplicaciones en diversas áreas y sobre la mayoría de las plataformas.

Caraterísticas

- Gratis y Libre
- Maduro (+18 años)
- Fácil de aprender
- Multiplataforma
- Compila a bytecode interpretado
- Multiparadigma
- Tipos de datos de alto nivel
- Intérprete interactivo

Baterías incluídas

Servicios del sistema, subprocessos, sockets, internacionalización, base de datos, threads, formatos zip, expresiones regulares, XML, HTML, XHTML, XML-RPC, email, clientes HTTP, FTP, SMTP, NNTP, POP3, IMAP4, servidores HTTP, SMTP, herramientas MIME, serializador y deserializador de objetos, debugger, profiler, random, curses, logging, compilador, decompilador, CSV, interfaz gráfica incorporada, matemática real y compleja, criptografía, introspección, unit testing, doc testing, etc, etc, etc...

Si lo estandar no alcanza

- Bases de datos: MySQL, PostgreSQL, MS SQL, Informix, DB/2, SQLite ...
- Interfaces gráficas: Qt, GTK, win32, wxWidgets, Cairo ...
- Frameworks Web: Django, web2py, Pyramid, Zope, Plone, webpy ...
- Y un montón más de temas: PIL, PyGame, SymPy, Numpy ...

Python es productivo

- Empinada curva de aprendizaje
- Código corto y legible
- Nos concentramos en nuestro problema
- Podemos enlazar con código C
- No tiene ciclos de compilación y enlace

Programando en Python

Hola mundo!!

```
In []: print "Hola Mundo"
```

Tipos de Datos

Números Enteros

```
In []: a = 2  
      b = 5  
      c = 15425
```

```
In []: print a * b
```

```
In []: print b / a # Operación entre enteros retorna enteros
```

```
In []: print c ** b
```

```
In []: print c ** b ** b
```

Números Flotantes

```
In []: f = 2.0 # Escribiendo en formato decimal deja de ser entero
```

```
In []: print a * f
```

```
In []: print c / f # Operación con un número flotante retorna flotante
```

```
In []: print c / a # Operación entre enteros retorna entero
```

Números Complejos

```
In []: nc = 3 + 4j
```

```
In []: print nc + nc
```

```
In []: print nc * nc
```

Funciones Útiles

```
In []: # Valor absoluto
print abs(123.45)
# Conversión de tipo
print float(5)
print int("52") + 8
# Redondeo
print round(123.45678, 1)
print round(123.45678, 2)
print round(123.45678, 4)
```

Cadenas

```
In []: # Definición
s = 'Texto1'
t = "Texto2"
u = """Texto multilinea
de ejemplo"""
```

```
In []: s
```

```
In []: u
```

```
In []: print u
```

Operaciones con cadenas

```
In []: # Concatenar
print s + s
```

```
In []: # Replicar
print s * 5
```

```
In []: # Operaciones del "objeto" cadena
s = "Texto"
print s
print s.strip() + "#"
print s.startswith('T')
print s.lower()
print s.upper()
```

Las cadenas son listas

```
In []: s = "Texto de ejemplo"
print s[0] # Subindice
print s[-1] # Subindice desde el final
print s[1:3] # Slice (rebanada)
print s[2:] # Slice sin final
print s[:4] # Slice sin inicio
print s[::2] # Slice con paso
print s[::-1] # Slice con paso negativo
```

Las cadenas son inmutables

No pueden modificarse

```
In []: s = "Texto"
s[0] = 'S'
```

```
In []: # Para reemplazar se hace una copia modificada
s = "Texto"
print s
s = "S" + s[1:]
print s
```

Formateando datos

```
In []: a = 5
b = "Textos"
c = 1.5555
print """Datos con formato:
a: %d\tb: %s\tc: %.3f""" % (a, b, c)
```

Listas

```
In []: tres = 3.0
lista = [1, "dos", tres]
print lista
```

```
In []: # Agregar elementos
      lista.append('texto')

      lista.insert(0, 'cero')
      print lista
```

```
In []: # Eliminar elementos
      del lista[2]
      print lista
```

```
In []: # Contactenar
      lista = lista + [4, 5, 6]
      print lista
```

```
In []: # Algunas funciones del "objeto" lista
      lista.reverse()
      print lista
      lista.sort()
      print lista # asdasd
      #print lista.index(4)
      #del lista[lista.index(4)]
      print lista
```

```
In []: # Listas por comprensión
      lista = [e for e in range(10) if e % 2 == 0]
      print lista
```

```
In []: # Generando cadenas desde listas
      ",".join([str(e) for e in lista]) # Funcion "join" del "objeto" cadena
```

Tuplas

```
In []: tupla = (1, 2, 3) # La diferencia con las listas es que las tuplas son
      inmutables
```

Conjuntos

```
In []: # Definiendo un conjunto
      c1 = set([1, 2, 3])
      c2 = set([3, 4, 5])
```

```
In []: # Algunas Operaciones
      print c1 & c2 # Intersección
      print c1 | c2 # Unión
      print c1 - c2 # Diferencia
```

Diccionarios

```
In []: # Son tablas HASH (clave -> valor)
      alumnos = {
          '35353535': 'Juan Pérez',
          '45454545': 'Jorge García',
      }
```

```
In []: # Obteniendo valores
      print alumnos['35353535']
```

```
In []: # Claves disponibles
      print alumnos.keys()
```

```
In []: # Valores disponibles
      print alumnos.values()
      # Vista de cadenas con "unicode"
```

```
In []: # Agregando elementos
      alumnos['35353535'] = 'Nuevo alumno'
      print alumnos
```

Estructuras de Control

if / else / elif

```
In []: a = 5
      if a < 5:
          print "Es chico"
      elif a < 10:
          print "Es intermedio"
      else:
          print "Es grande"
```

```
In []: # Operadores: <, >, <=, >=, ==, !=, is, in, and, or, not
      print 1 in (1, 2, 3)
      print True and False
      print False or False
```

for

```
In []: # El for es un iterador de elementos
      lista = ["asd", "ffff", 3, 4]
      for e in lista:
          print e

      # NO USA INDICES PARA ACCEDER A LOS ELEMENTOS
```

while

```
In []: n = 5
      while n > 0:
        print n
        n -= 1
```

Funciones para tener a mano

dir

Lista las funciones y atributos de un objeto

```
In []: dir('texto')
```

```
In []: dir(1)
```

```
In []: dir([])
```

```
In []: dir({})
```

help

Muestra la ayuda de una función o módulo

```
In []: help("".zfill)
```

```
In []: help([].insert)
```

```
In []: help("".join)
```

Mas info

- <http://www.python.org> (<http://www.python.org>)
- <http://www.python.org.ar> (<http://www.python.org.ar>)
- <http://docs.python.org.ar/tutorial/index.html> (<http://docs.python.org.ar/tutorial/index.html>)