**Funciones:**

* **apunte\_7540.pdf,** capítulo 3.
* Funciones anónimas lambda:

lambda [lista\_parámetros]: Expresión\_devuelta

De esta forma se crea una función anónima, esto es, una función que no tiene identificador. Expresión\_devuelta debe de ser una expresión, no una declaración (es decir, no “if xx:...”, “print xxx”, etcétera) y además no puede contener múltiples líneas. Ejemplo:

totalfactura = **lambda** base, iva: base + ((base \* iva) / 100)

print(totalfactura(1000, 21))

**Módulos:**

* **TutorialPython3.pdf** página 37
* P**ython-principiantes.pdf , página 33 ...**

**Localización de los módulos:**

Cuando incluimos la declaración import para la carga del módulo correspondiente no se le dice a Python dónde buscar. Por defecto Python buscará primero en el directorio donde tengamos el programa principal ó donde resida el intérprete Python..

Una manera de ver las rutas por las cuales Python intenta cargar los módulos mediante import es cargando el módulo de sistema sys (en el intérprete), y a continuación ejecutar sys.path. Esto nos devolverá una lista con todas las rutas a las que Python accede para buscar los módulos requeridos.

import sys

print (sys.path)

**El listado completo de la librería estándar para Python 2.6.6 se puede encontrar en:**

* <https://docs.python.org/3/library/index.html>

**Módulos empaquetados y espacios de nombres**

En Python, cada uno de nuestros **archivos .py se denominan módulos**. Estos módulos, a la vez, pueden formar parte de **paquetes**. Un paquete, es una carpeta que contiene archivos .py. Pero, para que una carpeta pueda ser considerada un paquete, debe contener un archivo de inicio llamado **\_\_init\_\_.py**. Este archivo, no necesita contener ninguna instrucción. De hecho, puede estar completamente vacío.

En el módulo a importar, que contiene las funciones, si queremos que sea importado y ejecutado desde él mismo, pero que al ser importado, solo se ejecuten las funciones que tiene, hay que añadir el siguiente cómo al final del mismo:

**if** \_\_name\_\_==**"\_\_main\_\_"**:

main()

1. **Import Math-** ¿Qué resultados se obtendrán al evaluar las siguientes expresiones Python? Observa la prioridad de paréntesis y operadores.

a) int(exp(2 \* log(3)))

b) round(4\*sin(3 \* pi / 2))

c) abs(log10(.01) \* sqrt(25))

d) round(3.21123 \* log10(1000), 3)

Resultado:

int(exp(2 \* log(3))) = 9

round(4\*sin(3 \* pi / 2))= -4

abs(log10(.01) \* sqrt(25))= 10.0

round(3.21123 \* log10(1000), 3)= 9.634

1. **import Sys** - Muestra algunos datos del sistema de tu equipo, por ejemplo: el número máximo de un entero, el copyright de python, la versión de la api, el estilo de float, el juego de caracteres utilizado, ...

MÁXIMO número entero 2147483647

VERSIÓN API 1013

COPIYRIGHT Copyright (c) 2001-2014 Python Software Foundation.

All Rights Reserved.

Copyright (c) 2000 BeOpen.com.

All Rights Reserved.

Copyright (c) 1995-2001 Corporation for National Research Initiatives.

All Rights Reserved.

Copyright (c) 1991-1995 Stichting Mathematisch Centrum, Amsterdam.

All Rights Reserved.

FLOAT STYLE short

JUEGO DE CARACTERES USADOS utf-8

1. **Importar módulo**. Crea un paquete o carpeta y en ella guarda un archivo.py o módulo. En dicho archivo realiza las siguientes funciones:
   1. Dado un lado que devuelva el área de un cuadrado.
   2. Dadas las base y altura de un triángulo, devuelva el área de un triángulo.
   3. Dado un prompt o texto de mensaje de entrada, devuelve un número tecleado, convertido a entero, real y string. Devuelve 3 datos.

Desde un archivo principal importa y utiliza dichas funciones para obtener este resultado:

1. Área de cuadrado

2. Área de triángulo

Teclea una opción (1-2:) 2

Base:3

Altura:5

Área del triángulo es: 7.50

1. **Lambda.** realiza cuatro funciones **lambda**, a las que pasando dos números, realicen una suma, una resta, una multiplicación y una división de dichos números, respectivamente. Escribir dichos resultados.

Teclea un número:5

Teclea un número:6

Suma 5.00 + 6.00: 11.00

Resta 5.00 - 6.00: -1.00

Multiplicación 5.00 \* 6.00: 30.00

División 5.00 / 6.00: 0.83