

#### **Funciones**



- Python provee funciones primitivas de las cuáles hoy aprendimos cinco:
  - Valor absoluto y potencia y las operaciones para cambiar el tipo a entero, entero largo y número de punto flotante
- Las funciones en Python siguen el concepto matemático de función:
  - Tienen un nombre
  - Se le entregan argumentos sobre los cuales se aplica
  - Siempre devuelven un valor (MEDIANTE RETURN)

## **Funciones importadas**



- Podemos extender la funcionalidad de Python importando funciones
  - Desde bibliotecas llamadas módulos
  - Hay dos formas para importarlas/usarlas:

Forma de importar	Forma de usar
import <nombre módulo=""></nombre>	<nombre modulo="">.<nombre función="">(parámetros)</nombre></nombre>
Ej: import math	Ej: math.sqrt(2)
<b>from &lt;</b> nombre módulo> <b>import &lt;</b> nombre función>	<nombre función="">(parámetros)</nombre>
Ej: from math import sqrt	Ej: sqrt(2)

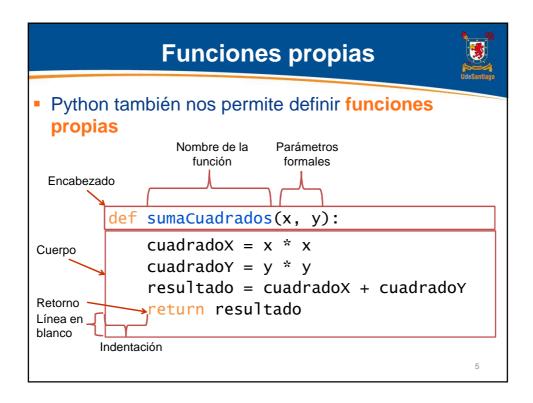
3

### Funciones importadas: Módulo math



 El módulo math contiene varias funciones matemáticas y trigonométricas, por ejemplo:

Nombre	Descripción
sin(x)	Seno de x, con x expresado en radianes
cos(x)	Coseno de x, con x expresado en radianes
tan(x)	Tangente de x, con x expresado en radianes
exp(x)	Número e elevado a x
log(x)	Logaritmo natural (base $e$ ) de x
log10(x)	Logaritmo en base decimal de x
sqrt(x)	Raíz cuadrada de x
degrees(x)	Convierte a grados un ángulo x expresado en radianes
radians(x)	Convierte a radianes un ángulo x expresado en grados.



### Parámetros formales y reales



- Recordemos:
  - Al declarar una función, indicamos los parámetros formales en el encabezado, y los usamos en el cuerpo de la función
  - Al invocar la función, lo hacemos con su nombre y especificamos los parámetros actuales
  - Al momento de la ejecución, se asocia cada parámetro formal a un parámetro actual, de acuerdo a sus posiciones
  - Por esto, el número de parámetros actuales a especificar en la invocación, debe ser igual al número de parámetros formales declarados para la función

# Parámetros formales y reales



- En el ejemplo:
  - Se declara la función sumaCuadrados
  - Que tiene dos parámetros formales: x e y
  - Podemos invocarla, por ejemplo, con la expresión sumaCuadrados (2, 3)
  - El parámetro formal x toma el valor 2 y el parámetro formal y toma el valor 3
  - Se ejecuta cada sentencia en el orden que aparecen:

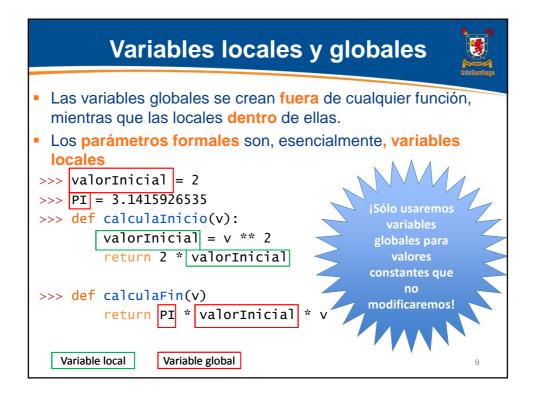
```
cuadradoX = 2 * 2 = 4
cuadradoY = 3 * 3 = 9
resultado = 4 + 9 = 13
return 13
```

7

### Variables locales y globales



- Las variables tienen cierta visibilidad
  - Técnicamente, a esta visibilidad se le llama alcance
  - Hay esencialmente dos tipos de alcances:
    - las variables locales sólo son visibles en el cuerpo de la función donde se crean (con una sentencia de asignación)
    - las variables globales son visibles por todas las sentencias que siguen a su creación, excepto donde existe una variable local con el mismo nombre
  - Si una variable es local o global, depende de dónde es creada



### Para la próxima clase



- Probar funciones nativas y del módulo math
- Revisar guía de ejercicios propuestos
- Comenzaremos a usar distintas funciones importadas y propias para unirlas en un programa

