

WELCOME TO PYTHON
VARIABLES AND DATA TYPES
FUNCTIONS
COLLECTIONS
CONDITIONALS
LOOPS
DEBUGGING AND ERROR HANDLING

The slides are meant as visual support for the lecture.
They are neither a documentation nor a script.

Please do not print the slides.

Comments and feedback at n.meseth@hs-osnabrueck.de

WELCOME TO PYTHON

[BACK](#)

```
print("hello, world")
```

use functions (or *commands*)

built-in functions

```
print()  
input()  
...
```

functions from built-in modules

```
math.sqrt()  
time.sleep()  
sys.exit()  
...
```

external modules

```
requests.get()  
...
```

arguments / parameter

comments


```
# this is a comment  
print("hello, world")
```

```
# this is a comment  
print("hello, world") # another comment
```

```
'''
```

```
a multi-line comment  
for longer descriptions  
'''
```

```
print("hello, world")
```

bugs

syntax errors

runtime errors

user input

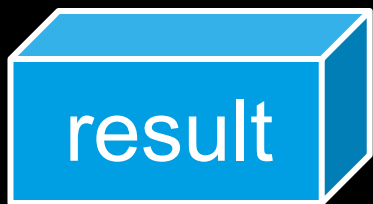
```
user_name = input("What's your name? ")
```

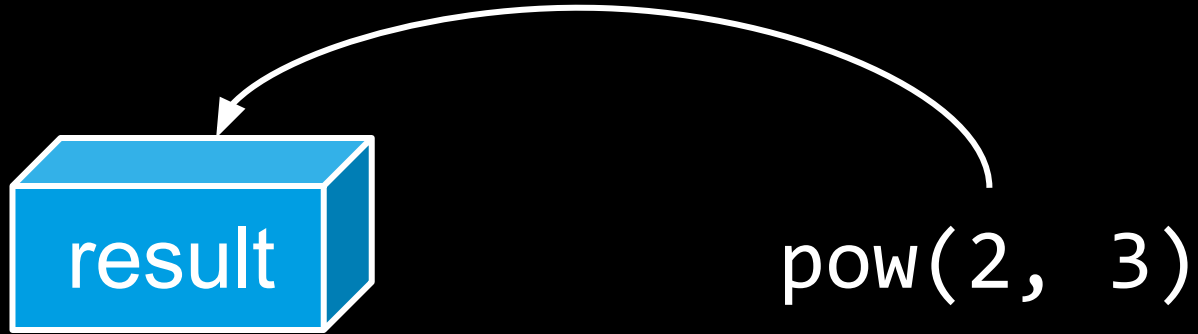

function's return values

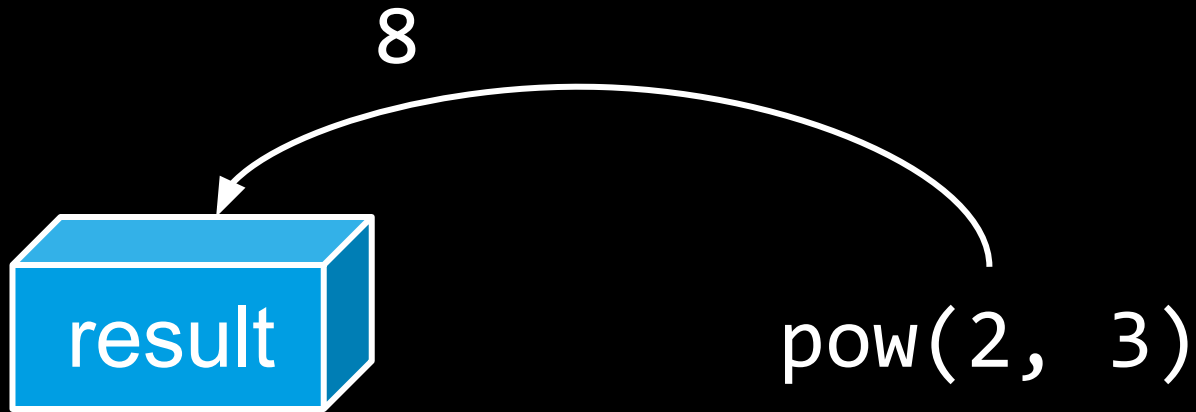
```
result = pow(2, 3)
```

VARIABLES AND DATA TYPES

[BACK](#)







```
exp = 4  
result = pow(2, exp)
```

 = 4

result = pow(2, )


```
exp = 4  
result = pow(2, exp)  
print(result)
```

operators

math

$$5 + 5$$

$$9 - 8$$

$$2 / 1$$

$$6 * 7$$

$$5 // 2$$

$$10 \% 3$$

$$2^{**}3$$

logic

$2 == 1$

$2*2 > 1+3$

$2*2 >= 1+3$

"A" < "B"

"A" < "B" and $2 == 1$

"A" < "B" or $2 == 1$

data types

integer

float

boolean

string

```
strip()  
capitalize()  
title()
```

format strings

```
print(f"Hello {name}")
```

comments

step 1: determine exponent

step 2: calculate power

step 3: print result

problem solving → problem decomposition

step 1: determine exponent

step 2: calculate power

step 3: print result

```
# step 1: determine exponent
```

```
exp = 4
```

```
# step 2: calculate power
```

```
# step 3: print result
```

```
# step 1: determine exponent
```

```
exp = 4
```

```
# step 2: calculate power
```

```
result = pow(2, exp)
```

```
# step 3: print result
```

```
# step 1: determine exponent
```

```
exp = 4
```

```
# step 2: calculate power
```

```
result = pow(2, exp)
```

```
# step 3: print result
```

```
print(result)
```

FUNCTIONS

[BACK](#)

create functions

```
def greet():  
    print("hello")
```


parameters

```
def greet(name):  
    print(f"hello {name}")
```

format strings

parameter default values

```
def greet(name="world"):
    print(f"hello {name}")
```

return a result

```
def make_greeting(name):  
    greeting = f"hello {name}"  
    return greeting
```

COLLECTIONS

[BACK](#)

CONDITIONALS

[BACK](#)

```
if <condition>:
```

```
...
```

```
if <condition>:
```

```
    ...
```

```
else:
```

```
    ...
```

```
if <condition>:  
    ...  
elif <condition>:  
    ...
```

LOOPS

[BACK](#)

while loop

for loop

DEBUGGING AND ERROR HANDLING