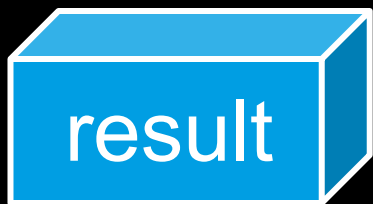
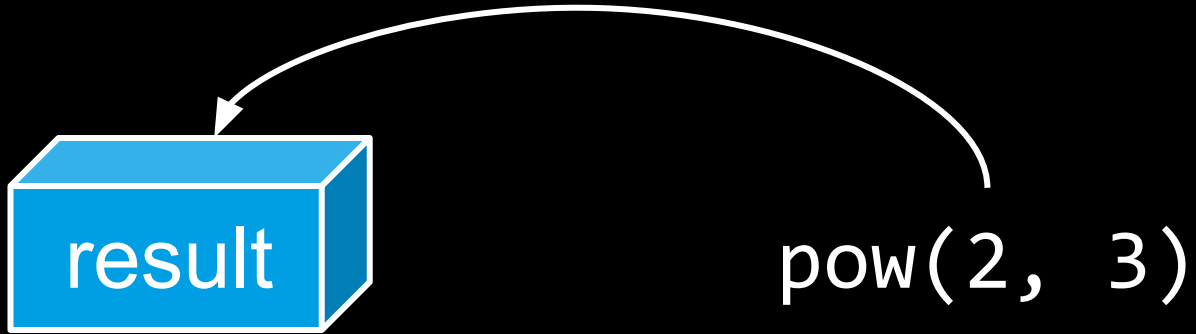
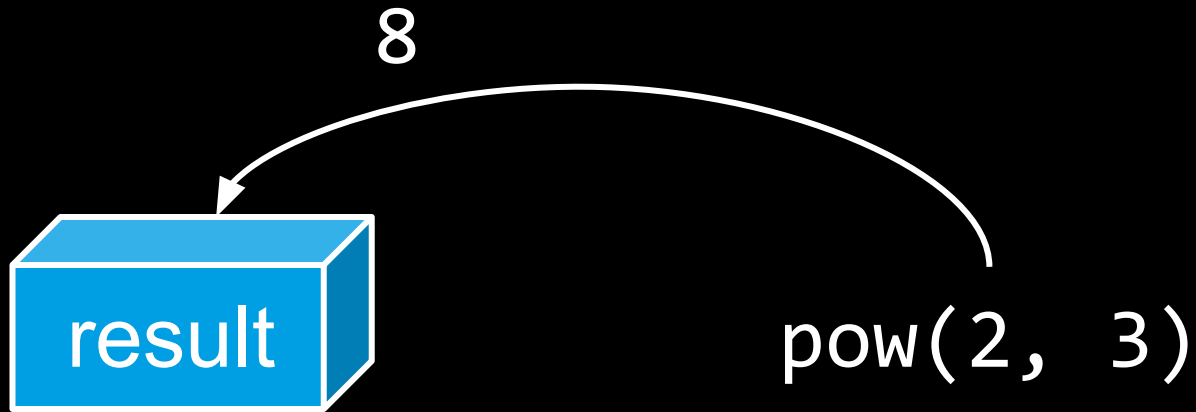


VARIABLES AND DATA TYPES


[BACK](#)







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

 = 4

result = pow(2, )

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

constants

PI = 3.14159

UID = "ZeW"

naming variables

use_underscores_for_spaces
start_with_small_letter
only_0123456789_and_letters
english_and_speaking_names

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

strings

== != > < >= <=

+

*

in / not in

[1] / [1:4]

strip()

capitalize()

title()

data types

integer

integer
float } numeric

integer
float
boolean

}

numeric

integer
float
boolean
string

} numeric

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)
```