

CMPSC 100 SPRING 2021

Functions



COURSE INFORMATION

- Grades – incoming.
 - Probably Wednesday.
 - Underpromise, over-deliver.
- As posted to #general, assignment and quiz are posted.
 - Much excite.
- This week's assignments features a different kind.
 - I'll show it to you now.
 - cc: Tyler for the suggestion
 - Address your fan mail to him; hate mail to me

Functions


Named procedures that accept
parameters.

Use case:

I can add $2 + 2$ once very easily.

What if I want to repeat
it 120 times, and it's not
always $2 + 2$?

function definition



```
def add(addend1, addend2):  
→ total = addend1 + addend2  
  return total
```



return statement

def keyword

parameters

def add(addend1, addend2)

function name



arguments

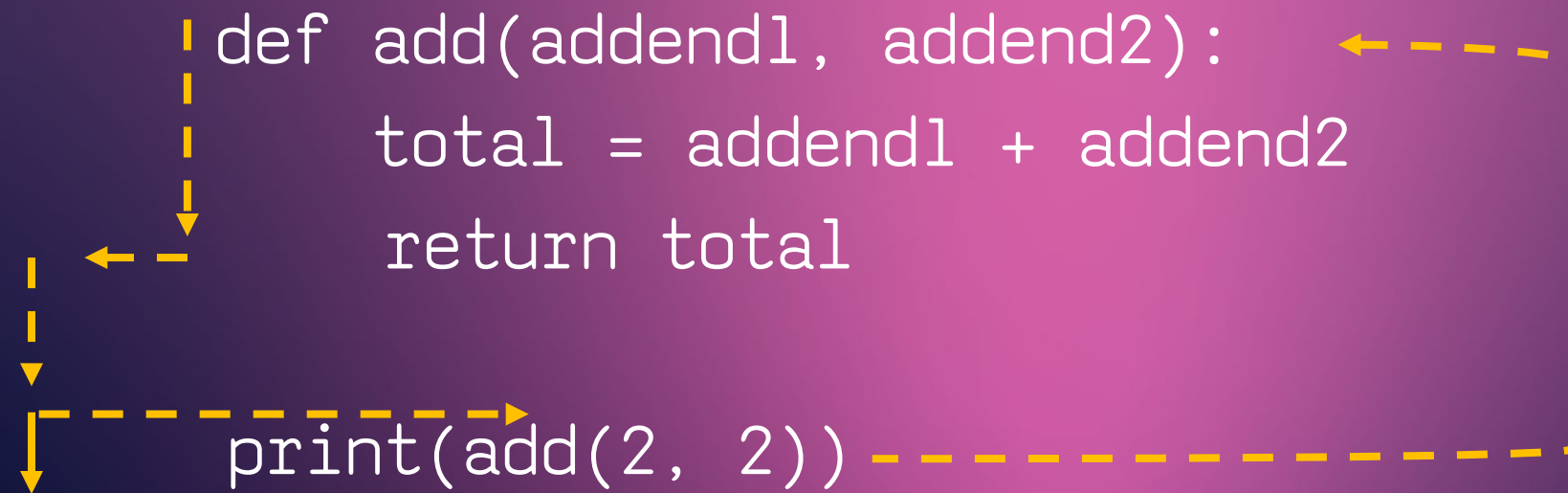


add(2, 2)



function call

```
def add(addend1, addend2):  
    total = addend1 + addend2  
    return total  
  
print(add(2, 2))
```

A diagram illustrating the execution flow of a function call. A dashed yellow arrow starts from the function definition line 'def add(addend1, addend2):', goes down, then left, then down again to point at the function call 'print(add(2, 2))'. Another dashed yellow arrow starts from the function call, goes right, then up, then left to point at the first parameter 'addend1' in the function definition. A third dashed yellow arrow starts from the function call, goes right, then up, then left to point at the second parameter 'addend2' in the function definition. A fourth dashed yellow arrow starts from the function call, goes right, then up, then left to point at the 'return total' line in the function definition.

> 4

What if I don't know how many
numbers I'm going to add?

```
def add (*addends):  
    total = 0  
    for addend in addends:  
        total += addend  
    return total
```

default value



```
def best_cat_name(name="Ulysses"):  
    print(f"The best cat name is: {name}")
```

```
best_cat_name("Bert") # <- - lies, but we need an example
```

```
best_cat_name()
```

GROUP INSTRUCTIONS

- Your task is to complete the code in today's final activity as a group
- I will group you quasi-randomly
- Dylan and I will be checking in on groups
 - We'll likely start with an "explain the task" conversation