

Life After Code In Place 2025

Week 1



WELCOME!

Intro

Intro

- LaCiP not sanctioned by CiP or Stanford

Intro

- LaCiP not sanctioned by CiP or Stanford
- No certificates / recommendations / job offers

Intro

- LaCiP not sanctioned by CiP or Stanford
- No certificates / recommendations / job offers
- Active participation and the 🖐️

Intro

- LaCiP not sanctioned by CiP or Stanford
- No certificates / recommendations / job offers
- Active participation and the 🖐
- Assignment links two weeks in advance on main forum

Intro

- LaCiP not sanctioned by CiP or Stanford
- No certificates / recommendations / job offers
- Active participation and the 🖐️
- Assignment links two weeks in advance on main forum
- Preview topics and problems before LaCiP sessions

Intro

- LaCiP not sanctioned by CiP or Stanford
- No certificates / recommendations / job offers
- Active participation and the 🖐️
- Assignment links two weeks in advance on main forum
- Preview topics and problems before LaCiP sessions
- Recordings on YouTube

Intro

- LaCiP not sanctioned by CiP or Stanford
- No certificates / recommendations / job offers
- Active participation and the 🖐️
- Assignment links two weeks in advance on main forum
- Preview topics and problems before LaCiP sessions
- Recordings on YouTube
- Questions in main forum (put LaCiP keyword in topic)

Intro

- LaCiP not sanctioned by CiP or Stanford
- No certificates / recommendations / job offers
- Active participation and the 🖐️
- Assignment links two weeks in advance on main forum
- Preview topics and problems before LaCiP sessions
- Recordings on YouTube
- Questions in main forum (put LaCiP keyword in topic)
- Professionalism and code of conduct

Intro

- LaCiP not sanctioned by CiP or Stanford
- No certificates / recommendations / job offers
- Active participation and the 🖐️
- Assignment links two weeks in advance on main forum
- Preview topics and problems before LaCiP sessions
- Recordings on YouTube
- Questions in main forum (put LaCiP keyword in topic)
- Professionalism and code of conduct
- Questions, suggestions, feedback: karelcip2024@gmail.com

Schedule

Schedule

Week 1 FizzBuzz, Follow Me, June 27th, 7 am PST
Baby Snake

Schedule

Week 1 **FizzBuzz, Follow Me,** **June 27th, 7 am PST**

Baby Snake

Week 2 **Data Analysis, GitHub,** **July 3rd, 4 pm PST**

Jupyter Notebook

Quiz Time

Quiz Time

- On the next slide you will see an image of a location

Quiz Time

- On the next slide you will see an image of a location
- Write the NAME of the location, as well as the CITY and the COUNTRY where it's located, in the Zoom chat

Quiz Time

- On the next slide you will see an image of a location
- Write the NAME of the location, as well as the CITY and the COUNTRY where it's located, in the Zoom chat
- The first person to answer correctly will get a \$5 Amazon gift certificate!

Quiz Time



LOCATION NAME, CITY AND COUNTRY

Quiz Time



FORBIDDEN CITY, BEIJING, CHINA

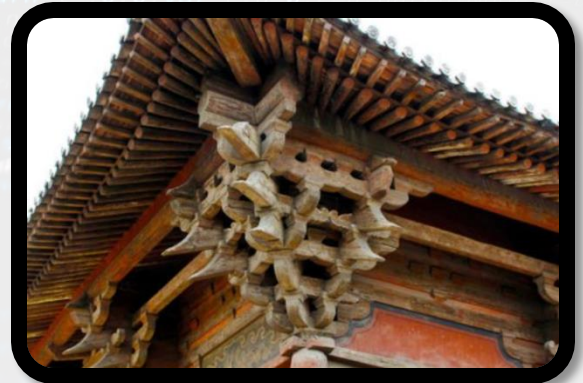
Quiz Time



There are no bird droppings on the roof because of the sloping lapis tiles that make it impossible for birds to perch on the smooth, sloping surface of the roof.

The Forbidden City has 999.5 rooms. Legend has it that a palace with 1,000 rooms can only exist in Heaven.

The unique interlocking brackets called dougong helped withstand earthquakes. They do not require nails or glue.



The Plan

The Plan

- Functions recap (15 min)
- FizzBuzz problem (15 minutes)

The Plan

- Functions recap (15 min)
- FizzBuzz problem (15 minutes)
- Canvas methods recap (10 minutes)
- Follow Me! problem (10 minutes)

The Plan

- Functions recap (15 min)
- FizzBuzz problem (15 minutes)
- Canvas methods recap (10 minutes)
- Follow Me! problem (10 minutes)
- Baby Snake problem (1 hour)

Functions

Functions

FUNCTION



Functions

FUNCTION

ACTION(S)

Functions

FUNCTION

IN



ACTION(S)

Functions

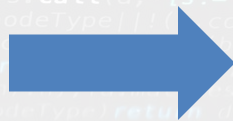
FUNCTION

IN

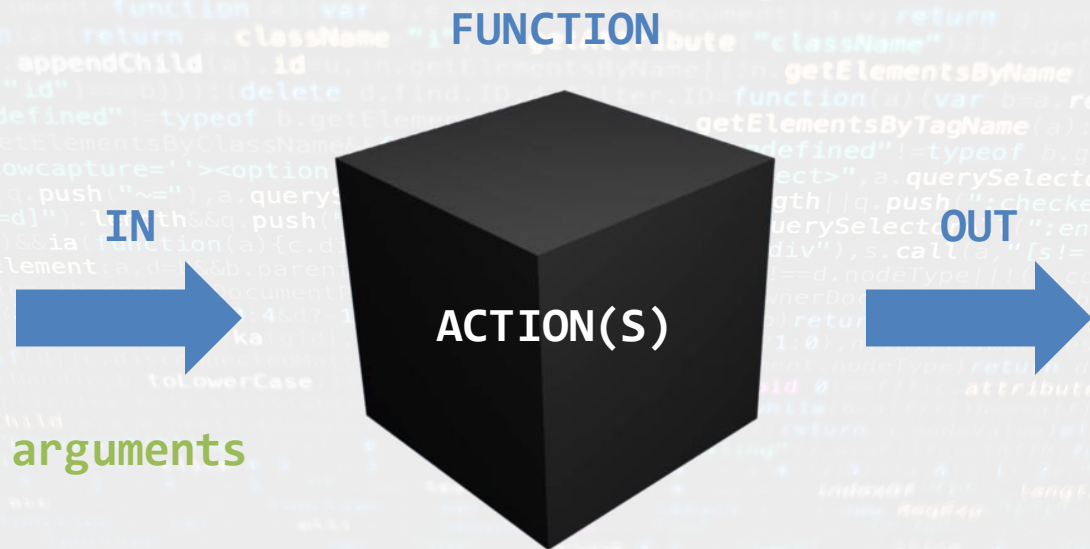


ACTION(S)

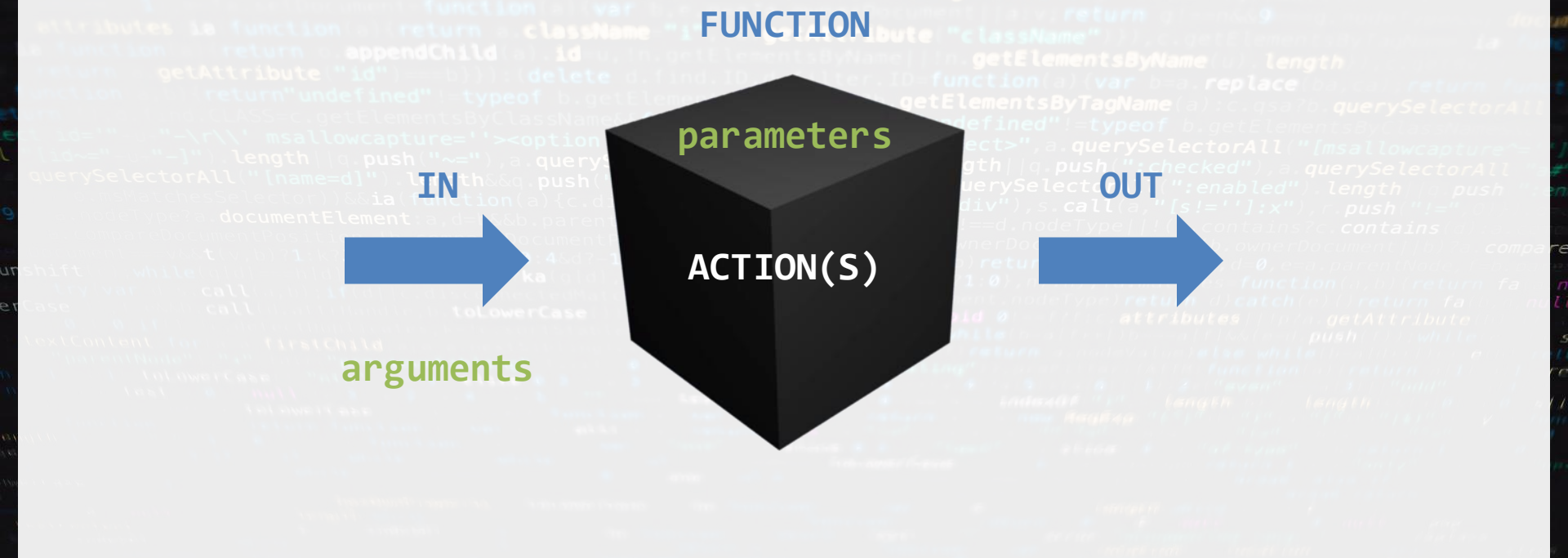
OUT



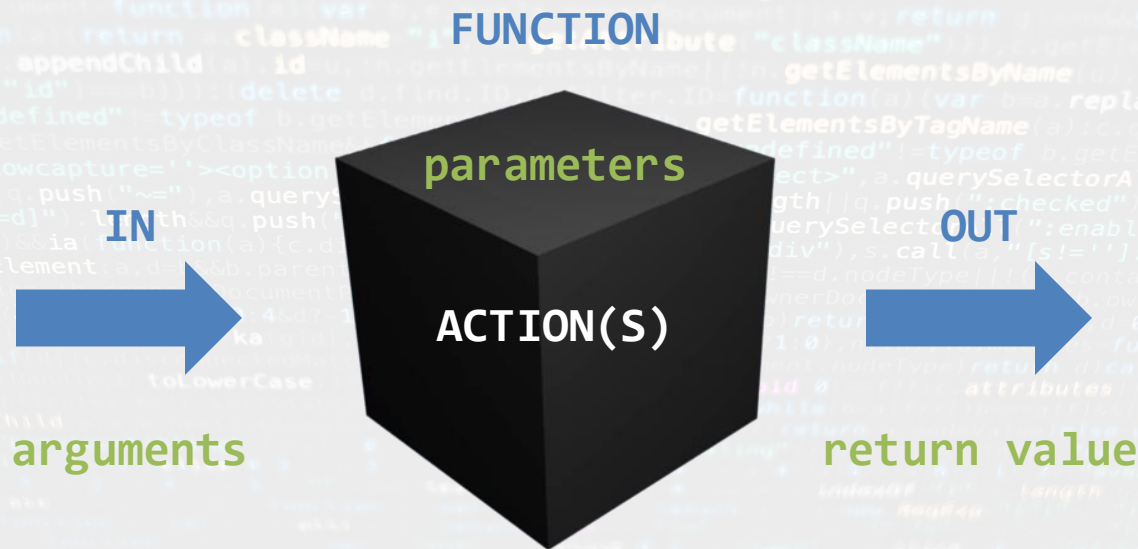
Functions



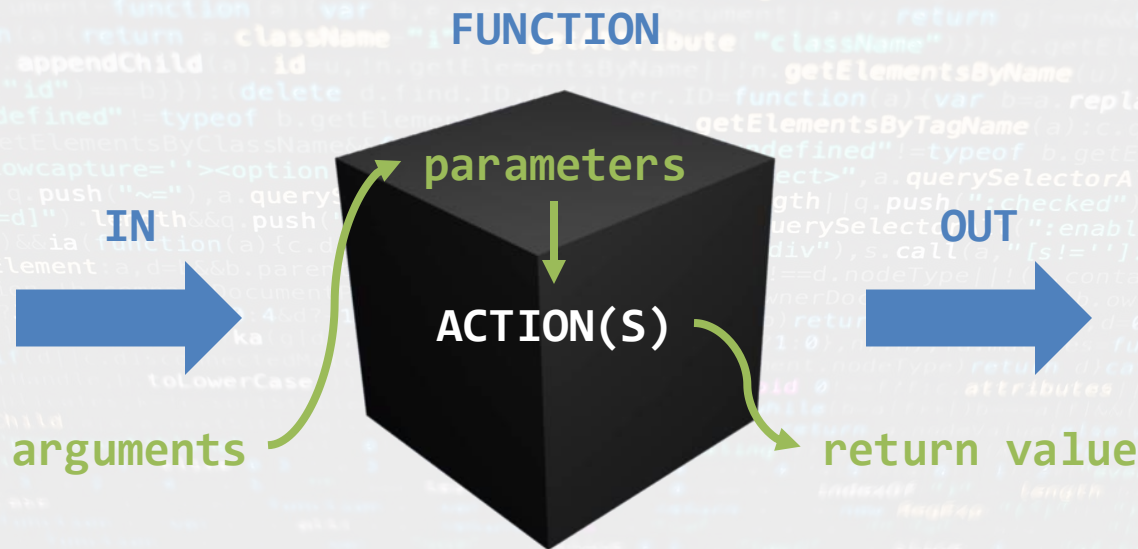
Functions



Functions



Functions



Functions

function definition

```
def power(num, pow):
```

```
    result = 1
```

```
    for i in range(pow):
```

```
        result = result * num
```

```
    return result
```

function call

```
print(power(2, 8))
```

num and pow are parameters

result is the return value

2 and 8 are arguments

Functions

```
def print_hello():    # no parameters  
    print("Hello!")  # no return value
```



ACTION(S)

Functions

```
def print_hello():    # no parameters  
    print("Hello!")  # no return value
```



ACTION(S)

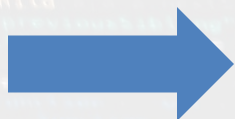


Hello!

Functions

```
def print_hello(first, last): # parameters
    print(f"Hello, {first} {last}!") # no return value
```

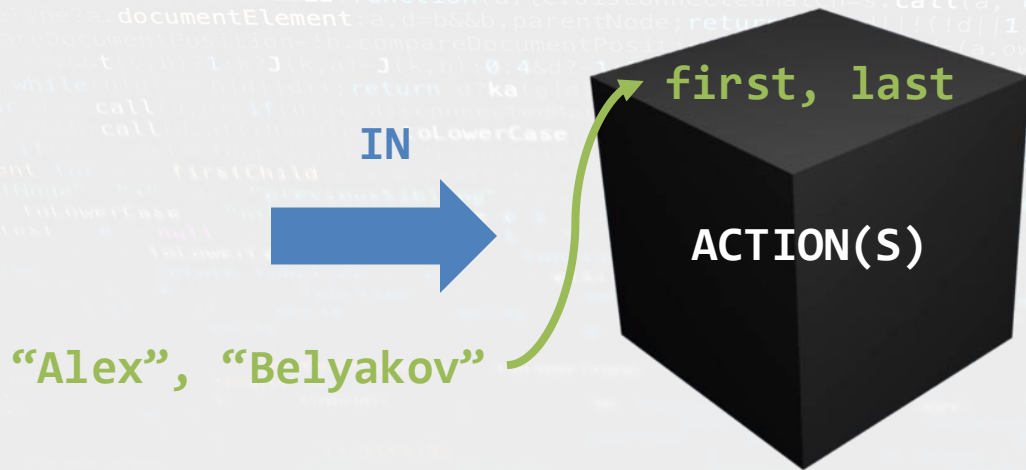
IN



“Alex”, “Belyakov”

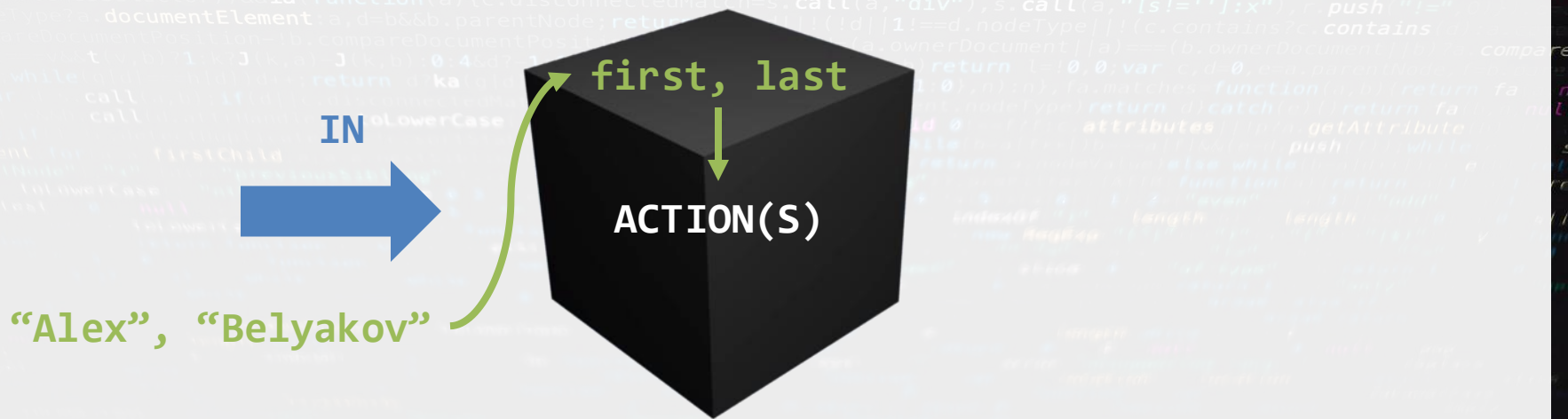
Functions

```
def print_hello(first, last): # parameters
    print(f"Hello, {first} {last}!") # no return value
```



Functions

```
def print_hello(first, last): # parameters
    print(f"Hello, {first} {last}!") # no return value
```



Functions

```
def print_hello(first, last): # parameters
    print(f"Hello, {first} {last}!") # no return value
```



Functions

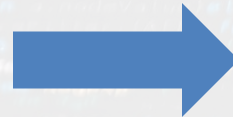
```
def throw_2d6():  
    die1 = random.randint(1, 6)  
    die2 = random.randint(1, 6)  
    return die1 + die2
```

no parameters

return value



OUT



Functions

```
def throw_2d6():  
    die1 = random.randint(1, 6)  
    die2 = random.randint(1, 6)  
    return die1 + die2
```

no parameters

return value



OUT



Functions

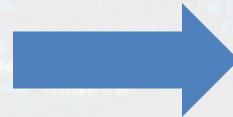
```
def throw_2d6():  
    die1 = random.randint(1, 6)  
    die2 = random.randint(1, 6)  
    return die1 + die2
```

no parameters

return value



OUT

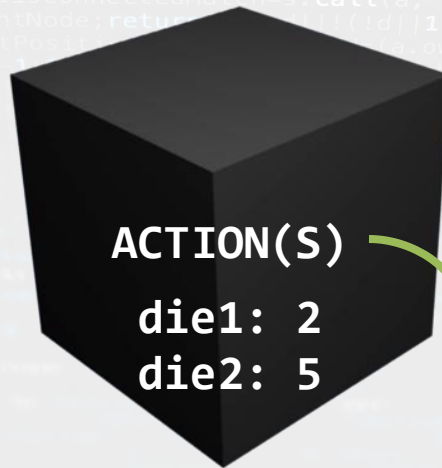


Functions

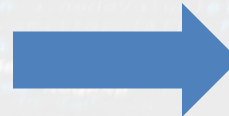
```
def throw_2d6():  
    die1 = random.randint(1, 6)  
    die2 = random.randint(1, 6)  
    return die1 + die2
```

no parameters

return value



OUT



7

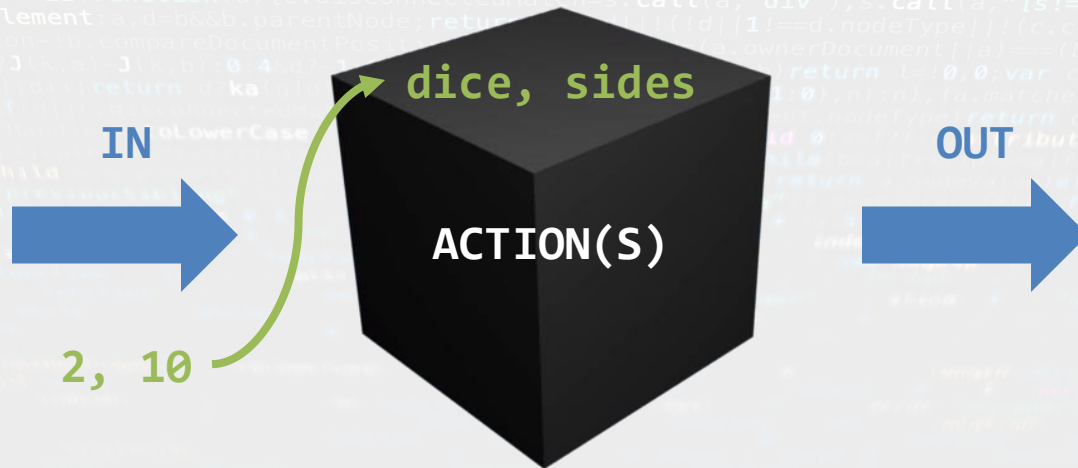
Functions

```
def throw_dice(dice, sides): # parameters
    total = 0
    for i in range(dice):
        total += random.randint(1, sides)
    return total # return value
```



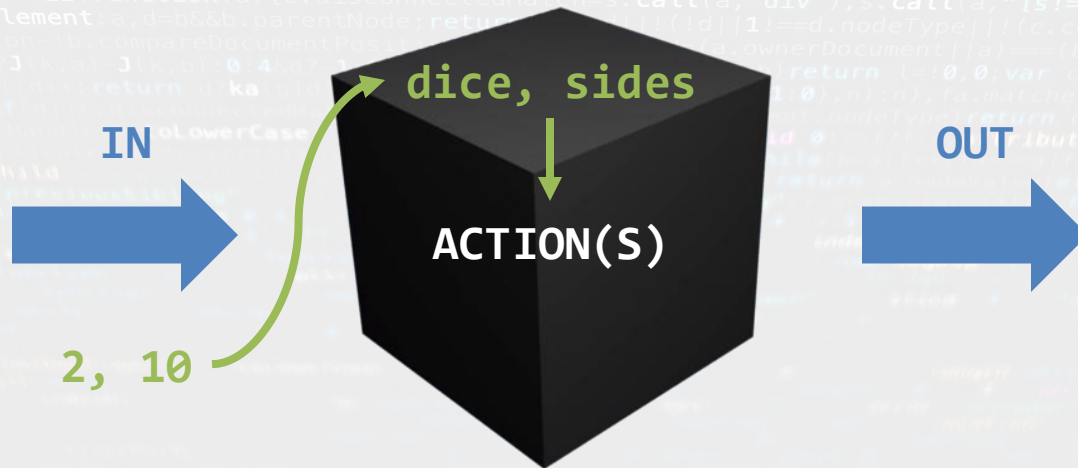
Functions

```
def throw_dice(dice, sides): # parameters
    total = 0
    for i in range(dice):
        total += random.randint(1, sides)
    return total # return value
```



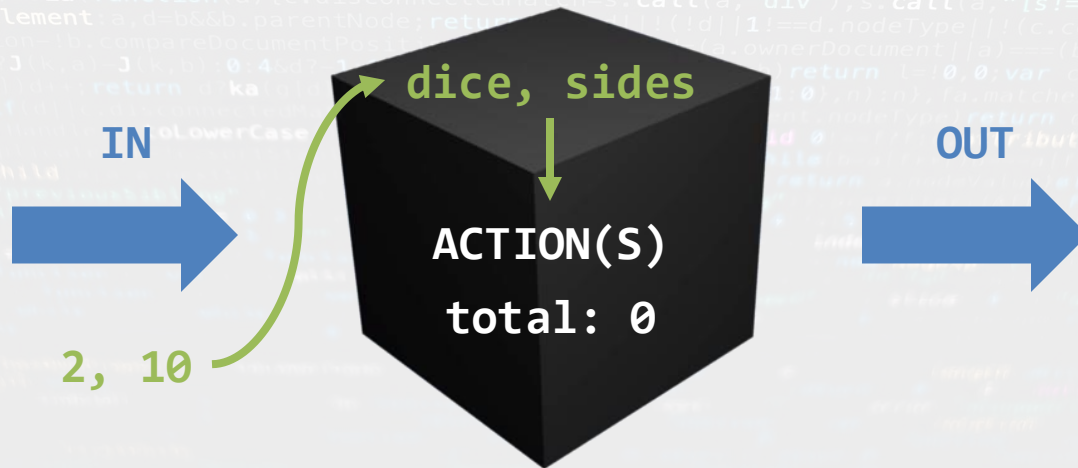
Functions

```
def throw_dice(dice, sides): # parameters
    total = 0
    for i in range(dice):
        total += random.randint(1, sides)
    return total # return value
```



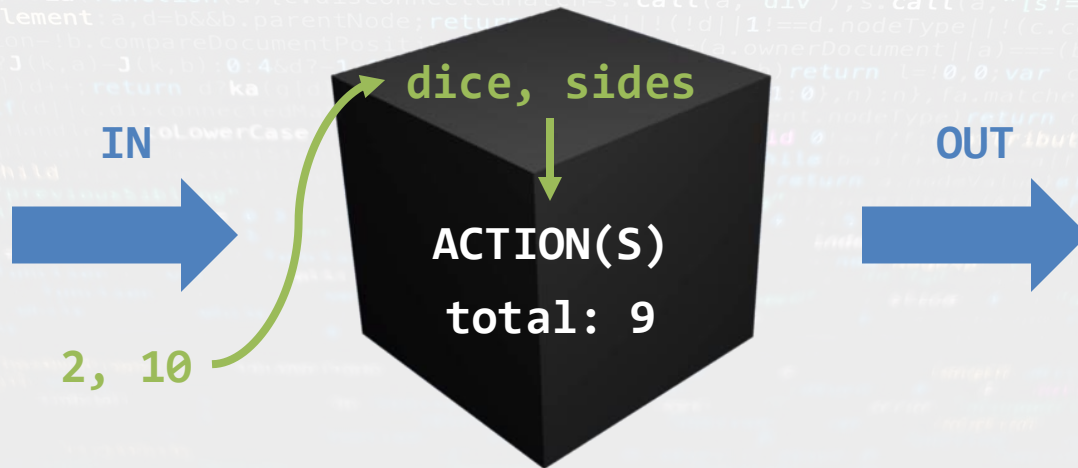
Functions

```
def throw_dice(dice, sides): # parameters
    total = 0
    for i in range(dice):
        total += random.randint(1, sides)
    return total # return value
```



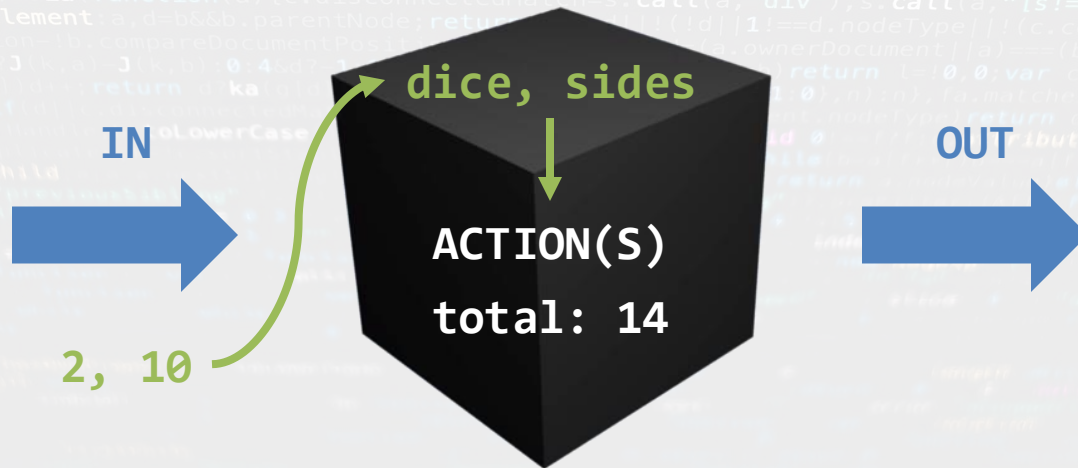
Functions

```
def throw_dice(dice, sides): # parameters
    total = 0
    for i in range(dice):
        total += random.randint(1, sides)
    return total # return value
```



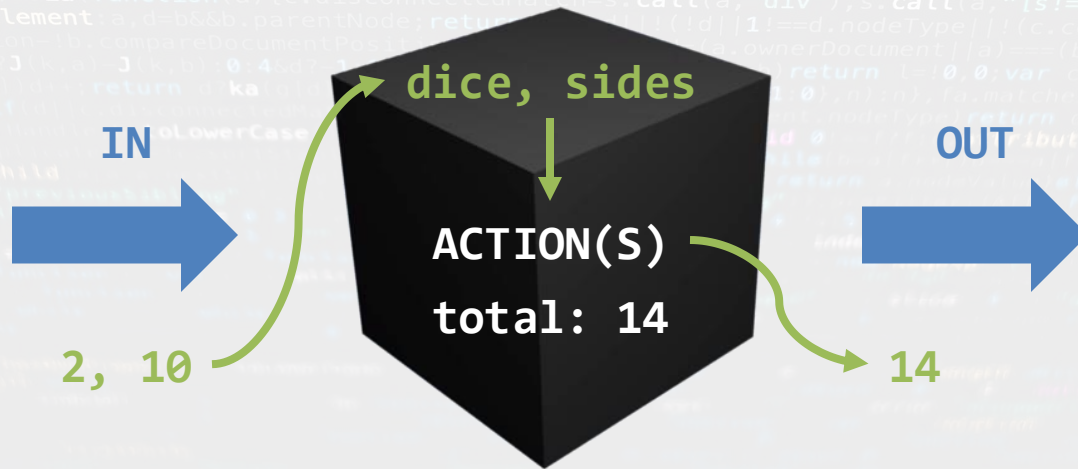
Functions

```
def throw_dice(dice, sides): # parameters
    total = 0
    for i in range(dice):
        total += random.randint(1, sides)
    return total # return value
```



Functions

```
def throw_dice(dice, sides): # parameters
    total = 0
    for i in range(dice):
        total += random.randint(1, sides)
    return total # return value
```



Fizzbuzz Problem

Fizzbuzz Problem

- For every number from 1 to `MAX_VALUE`:
 - Print “`Fizz`” if number divisible by 3
 - Print “`Buzz`” if number divisible by 5
 - Print “`Fizzbuzz`” if number divisible by both 3 and 5
 - Otherwise print the number itself

Fizzbuzz Problem

1

2

Fizz

4

Buzz

Fizz

7

8

Fizz

Fizzbuzz Problem

1 Buzz

2 11

Fizz Fizz

4 13

Buzz 14

Fizz Fizzbuzz

7 16

8 17

Fizz

Fizzbuzz Problem

- Create a `fizzbuzz()` function that takes in a number as a parameter and returns the appropriate value to be printed
- Call function multiple times inside a loop

Fizzbuzz Problem



Canvas Methods

Vicki's Quiz Time 1—just checking if you're still awake?

```
3 def main():
4     # modify this starter code to call fizzbuzz
5     # on every number from 1 to MAX_VALUE
6     to_say = fizzbuzz(10)
7     print(to_say)
8
9 def fizzbuzz(n):
10     """
11     Takes in a positive integer (n) and returns
12     what the player should say at that value.
13     Here are a few examples:
14     fizzbuzz(3) returns "Fizz"
15     fizzbuzz(15) returns "Fizzbuzz"
16     fizzbuzz(2) returns 2
17     """
18     if n%3==0 and n%5==0: # if n%3 and n%5==0: would also work
19         print("Fizzbuzz")
20         return("Fizzbuzz")
21     elif n%3==0:
22         print("Fizz")
23         return("Fizz")
24     elif n%5==0:
25         print("Buzz")
26         return("Buzz")
27     else:
28         print(n)
29         return n
30
31 if __name__ == '__main__':
32     main()
```

A.

```
% python main.py
Buzz
```

B.

```
% python main.py
Buzz
Buzz
```

C.

```
% python main.py
Fizzbuzz
Fizzbuzz
```

D.

```
% python main.py
%
```

Vicki's Quiz Time 2—still awake?

```
3 def main():
4     # modify this starter code to call fizzbuzz
5     # on every number from 1 to MAX_VALUE
6     to_say = fizzbuzz(10)
7     #print(to_say)
8
9 def fizzbuzz(n):
10     """
11     Takes in a positive integer (n) and returns
12     what the player should say at that value.
13     Here are a few examples:
14     fizzbuzz(3) returns "Fizz"
15     fizzbuzz(15) returns "Fizzbuzz"
16     fizzbuzz(2) returns 2
17     """
18     if n%3==0 and n%5==0: # if n%3 and n%5==0: would also work
19         print("Fizzbuzz")
20         return("Fizzbuzz")
21     elif n%3==0:
22         print("Fizz")
23         return("Fizz")
24     elif n%5==0:
25         print("Buzz")
26         return("Buzz")
27     else:
28         print(n)
29         return n
30
31 if __name__ == '__main__':
32     main()
```

A.

```
% python main.py
Buzz
```

B.

```
% python main.py
Buzz
Buzz
```

C.

```
% python main.py
Fizzbuzz
Fizzbuzz
```

D.

```
% python main.py
%
```


Vicki's Quiz Time 3:

```
def main():  
    # modify this starter code to call fizzbuzz  
    # on every number from 1 to MAX_VALUE  
    to_say = fizzbuzz(10)  
    #print(to_say)
```

```
def fizzbuzz(n):  
    """
```

Takes in a positive integer (n) and returns
what the player should say at that value.

Here are a few examples:

fizzbuzz(3) returns "Fizz"

fizzbuzz(15) returns "Fizzbuzz"

fizzbuzz(2) returns 2

```
    """
```

```
    if n%3==0 and n%5==0: # if n%3 and n%5==0: would also work  
        #print("Fizzbuzz")  
        return("Fizzbuzz")  
    elif n%3==0:  
        #print("Fizz")  
        return("Fizz")  
    elif n%5==0:  
        #print("Buzz")  
        return("Buzz")  
    else:  
        #print(n)  
        return n
```

```
if __name__ == '__main__':  
    main()
```

A.

```
% python main.py  
Buzz
```

B.

```
% python main.py  
Buzz  
Buzz
```

C.

```
% python main.py  
Fizzbuzz  
Fizzbuzz
```

D.

```
% python main.py  
%
```

Canvas Methods

```
x = canvas.get_mouse_x() # get x coordinate of mouse  
y = canvas.get_mouse_y() # get y coordinate of mouse
```

Canvas Methods

`x = canvas.get_mouse_x()` # get x coordinate of mouse

`y = canvas.get_mouse_y()` # get y coordinate of mouse

`canvas.move(obj, x, y)` # move object BY x, y pixels

`canvas.moveto(obj, x, y)` # move object TO x, y coordinates

Canvas Methods

`x = canvas.get_mouse_x()` # get x coordinate of mouse

`y = canvas.get_mouse_y()` # get y coordinate of mouse

`canvas.move(obj, x, y)` # move object BY x, y pixels

`canvas.moveto(obj, x, y)` # move object TO x, y coordinates

`time.sleep(s)` # pause program for s seconds

(requires import time)

Follow Me! Problem

Follow Me! Problem

- You have the continuously updating **x** and **y** coordinates of the mouse in a loop
- Draw a blue circle of size **BALL_DIAMETER** that follows the mouse, centered on the cursor
- Bonus assignment: make it so the circle doesn't go out of bounds

Follow Me! Problem



Baby Snake Problem

- Set up the game world
- Animate the snake
- Handle key presses
- Detect collisions with walls
- Hitting and moving the goal
- Optional: Increase speed after each goal
- Optional: Keep track of points
- Optional: Add obstacles to the board

Baby Snake Problem

