Welcome to the CoGrammar

Skills Bootcamp: Using Built in Functions and Defining your own functions.

The session will start shortly...

Questions? Drop them in the chat. We'll have dedicated moderators answering questions.

CoGrammar

Cyber Security Session Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: **Questions**

Cyber Security Session Housekeeping cont.

- For all **non-academic questions**, please submit a query: **www.hyperiondev.com/support**

- We would love your **feedback** on lectures: **Feedback on Lectures**

- Find all the lecture **content** in you **Lecture Backpack** on GitHub.

Safeguarding & Welfare

We are committed to all our students and staff feeling safe and happy; we want to make sure there is always someone you can turn to if you are worried about anything.

If you are feeling upset or unsafe, are worried about a friend, student or family member, or you feel like something isn't right, speak to our safeguarding team:

Ian Wyles
Designated Safeguarding Lead

Simone Botes

Nurhaan Snyman

Rafiq Manan

Ronald Munodawafa

Tevin Pitts

Scan to report a safeguarding concern

or email the Designated Safeguarding Lead:
Ian Wyles
safeguarding@hyperiondev.com

CoGrammar HyperionDev

Stay Safe Series:

Mastering Online Safety One week at a Time

While the digital world can be a wonderful place to make education and learning accessible to all, it is unfortunately also a space where harmful threats like online radicalization, extremist propaganda, phishing scams, online blackmail and hackers can flourish.

As a component of this BootCamp the *Stay Safe Series* will guide you through essential measures in order to protect yourself & your community from online dangers, whether they target your privacy, personal information or even attempt to manipulate your beliefs.

Trustworthy Websites: How to Spot Secure Sites

When browsing online, it's crucial to identify secure and trustworthy websites. Look for URLs that start with HTTPS, as the 'S' indicates a secure connection. A padlock icon in the address bar also signifies a valid security certificate. Ensure the URL is spelled correctly and check for clear contact information, including a physical address and phone number. Additionally, legitimate websites provide privacy policies detailing how they handle your data. By following these guidelines, you can protect yourself from fraudulent sites and ensure a safer online experience.

Learning Objectives & Outcomes

- Identify and recall built-in Python functions such as print(), len(), and input().
- Describe the components of a function (defining, parameters, return statements).
- Create and call user-defined functions to perform specific operations.
- Examine the scope of variables within functions.
- Assess the efficiency and readability benefits of using functions.

SKILLS FOR LIFE
SKILLS BOOTCAMPS
Department for Education

CoGrammar

Using Built-in Functions & Defining Your Own Functions

October 2024

Functions

Just like a recipe provides a set of instructions to create a dish, a function provides a set of instructions to perform a specific task or calculation in a program

CoGrammar

Polls

Please have a look at the poll notification and select an option.

What does the len() function do in python?

A. Returns the length of a string or list
B. Finds the largest number in a list
C. Converts date into a string
D. Terminates a program

CoGrammar

Polls

Please have a look at the poll notification and select an option.

What built in function takes user input in python?

A. input()
B. len()
C. print()
D. sum()

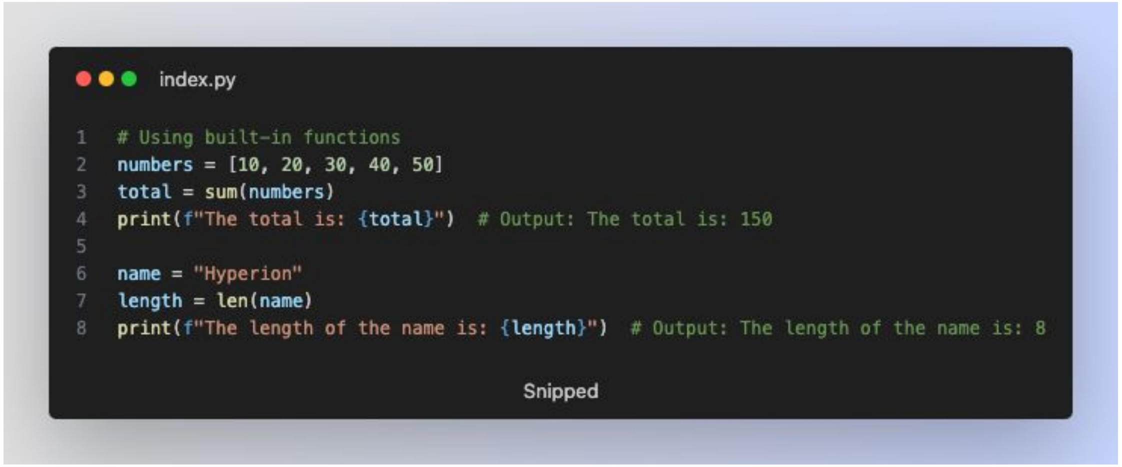
CoGrammar

Functions

- A function is a reusable block of code that performs a specific task.
- Functions are useful in the following ways:
    - Encapsulates logic
    - Makes code modular
    - Makes code organised
    - Makes code easier to read

Built-in Functions

- These are functions that are readily available for use without needing to define them. (e.g len(), input())



```python
# Using built-in functions
numbers = [10, 20, 30, 40, 50]
total = sum(numbers)
print(f"The total is: {total}")  # Output: The total is: 150

name = "Hyperion"
length = len(name)
print(f"The length of the name is: {length}")  # Output: The length of the name is: 8
```

CoGrammar

User-Defined Functions

- Functions that you create yourself to perform specific tasks that are not provided by the built-in functions.
- Parts of the function:
  - Function name
  - Function definition/body
  - Function Call
- We use the def keyword followed by the function name, parenthesis (which may include parameters), and a colon to create a function.

CoGrammar

Parameters vs Arguments

- Parameters are the variables listed inside the parentheses in the function definition. They act as placeholders for the values that will be passed to the function when it is called

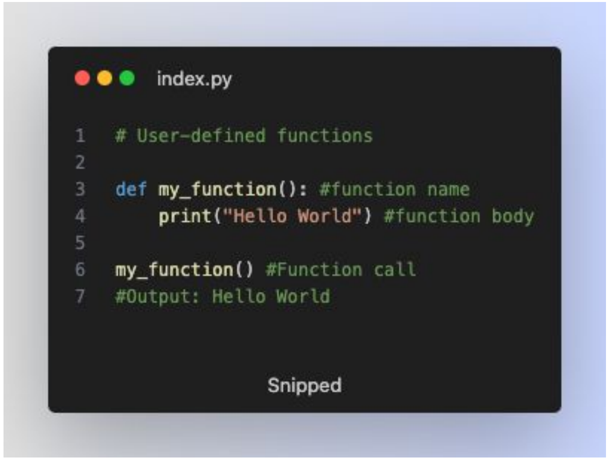
- Arguments are the actual values or data you pass to the function when calling it. These values replace the parameters defined in the function during the function call.

CoGrammar

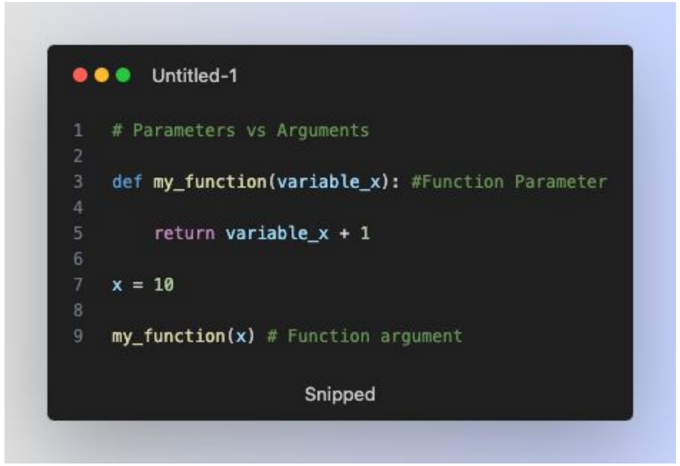
Let’s take a break

CoGrammar

User-defined Functions



```python
# User-defined functions

def my_function(): #function name
    print("Hello World") #function body

my_function() #Function call
#Output: Hello World
```

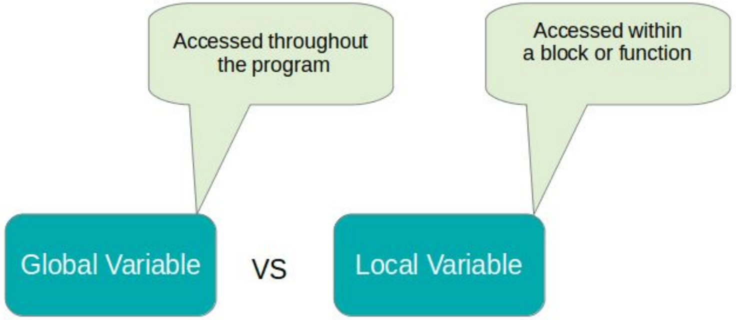


```python
# Parameters vs Arguments

def my_function(variable_x): #Function Parameter

    return variable_x + 1

x = 10

my_function(x) # Function argument
```

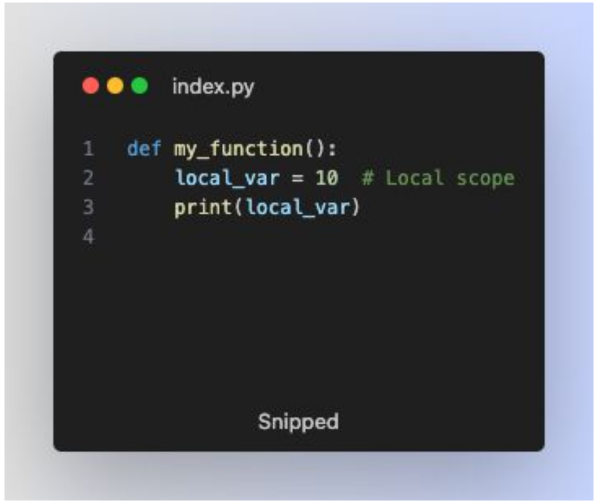
CoGrammar

Scopes in python

- Scope refers to the visibility and lifetime of variables in a program.

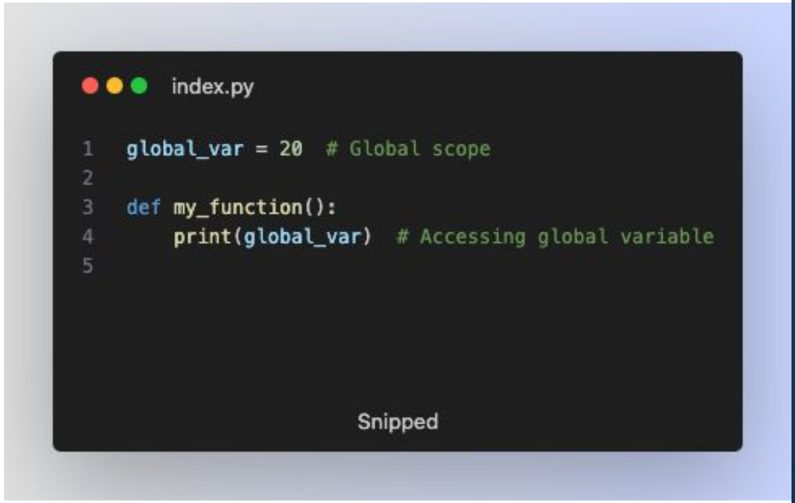

Accessed throughout the program
Accessed within a block or function
Global Variable
VS
Local Variable

Scopes in python


index.py

```python
def my_function():
    local_var = 10  # Local scope
    print(local_var)

```

Snipped


index.py

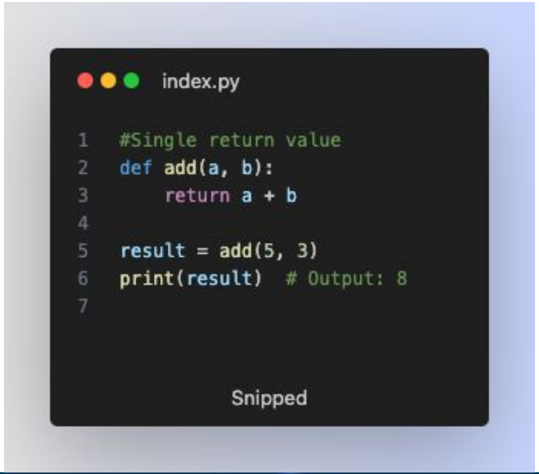
```python
global_var = 20  # Global scope

def my_function():
    print(global_var)  # Accessing global variable

```

Snipped

CoGrammar

Return Values in Functions

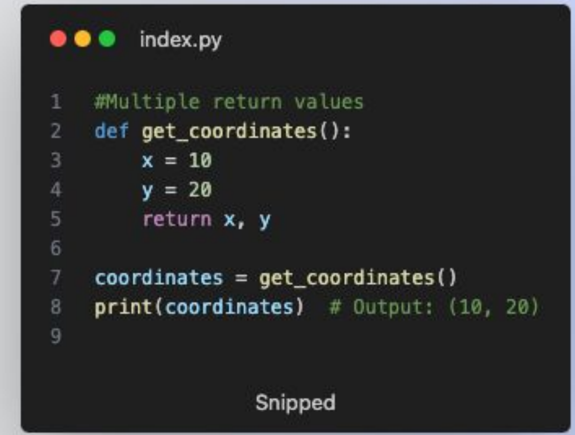
- A return value is the output that a function produces after it has finished executing.
- When a function reaches a return statement, it exits, and the value specified in the return statement is sent back to the caller.



```python
#Single return value
def add(a, b):
    return a + b

result = add(5, 3)
print(result)  # Output: 8
```

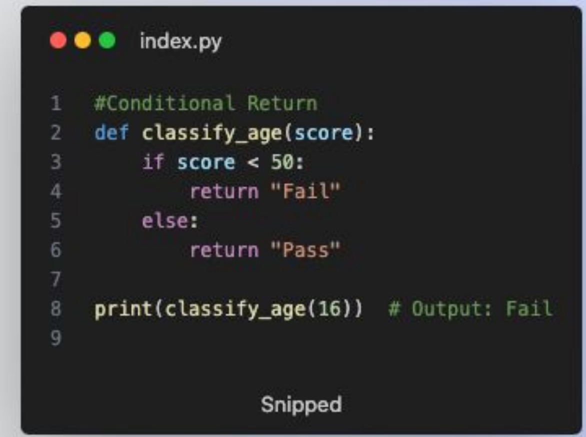
CoGrammar

Types of Return values.


index.py

```python
#Multiple return values
def get_coordinates():
    x = 10
    y = 20
    return x, y

coordinates = get_coordinates()
print(coordinates)  # Output: (10, 20)
```

Snipped


index.py

```python
#Conditional Return
def classify_age(score):
    if score < 50:
        return "Fail"
    else:
        return "Pass"

print(classify_age(16))  # Output: Fail
```

Snipped

CoGrammar

Polls

Please have a look at the poll notification and select an option.

Why would you create a user-defined function instead of using a built-in function

A. To avoid repetitive code
B. Built-in functions are too slow
C. Custom functions provide more flexibility and can handle specific tasks
D. Built-in functions are unreliable

Polls

Please have a look at the poll notification and select an option.

Which of the following scenarios demonstrates an understanding of function scope in Python?

A. A variable declared within a function can be accessed and modified directly from outside the function without any special declarations.
B. A nested function can access variables from its enclosing functions' scope, but those variables cannot be modified directly.
C. A global variable declared before a function can only be read inside the function but cannot be modified unless declared with the global keyword.
D. A function can be called before it is defined in the code as long as the function name is known.

CoGrammar

Conclusion & Summary

- Functions enhance code modularity, reusability, and readability.
- Built-in functions are efficient for common operations, while user-defined functions offer flexibility for specialised tasks.
- Proper use of functions is essential for writing clean and maintainable code.
- Functions in Python are like laws: they take inputs (arguments), process them using well-defined rules (the function body), and produce an outcome (return value).
- Just as laws apply equally to all, functions process inputs the same way every time, ensuring fairness and consistency in the results.

CoGrammar

Questions and Answers

CoGrammar

Thank you for attending

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

CoGrammar