Welcome to this CoGrammar tutorial: Functions, Sequences and Debugging

The session will start shortly...

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

Software Engineering 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** throughout this session, should you wish to ask any follow-up questions.
- 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

Software Engineering Session Housekeeping cont.

- For all **non-academic questions**, please submit a query: www.hyperiondev.com/support
- Report a **safeguarding** incident: www.hyperiondev.com/safeguardreporting
- We would love your **feedback** on lectures: Feedback on Lectures

Enhancing Accessibility: Activate Browser Captions

Why Enable Browser Captions?

- Captions provide **real-time text for spoken content**, ensuring inclusivity.
- Ideal for individuals in noisy or quiet environments or for those with **hearing impairments**.

How to Activate Captions:

1. **YouTube or Video Players:**
   - Look for the CC (Closed Captions) icon and click to enable.
2. **Browser Settings:**
   - Google Chrome: Go to *Settings > Accessibility > Live Captions* and toggle ON.
   - Edge: Enable captions in *Settings > Accessibility*.

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 or Step 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 radicalisation, extremist propaganda, phishing scams, online blackmail and hackers can flourish.

As a component of this BootCamp the *Stay Safe Series* is designed to 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.

Don’t Take the Bait: How to Spot Phishing Scams

- Check email address.
- Unprofessionalism.
- Avoid attachments from unknown sources.
- Avoid signs of urgency.
- Content should make logical sense.

Polls

CoGrammar

Poll

1. **Which of the following is a valid sequence in Python?**
   A. List
   B. String
   C. Tuple
   D. All of the above

Poll

2. **What does the slicing operation seq[1:4] do when applied to a sequence seq?**

   A. Retrieves the elements at indices 1, 2, and 3

   B. Retrieves the elements at indices 1, 2, 3, and 4

   C. Retrieves the element at index 4 only

   D. None of the above

Poll

3. **What is the keyword used to define a function in Python?**

   A. function

   B. def

   C. lambda

   D. define

Poll

4. **What does the return statement do in a function?**

   A. Terminates the function without returning any value.

   B. Returns a value to the caller and terminates the function.

   C. Loops back to the start of the function.

   D. None of the above.

Poll

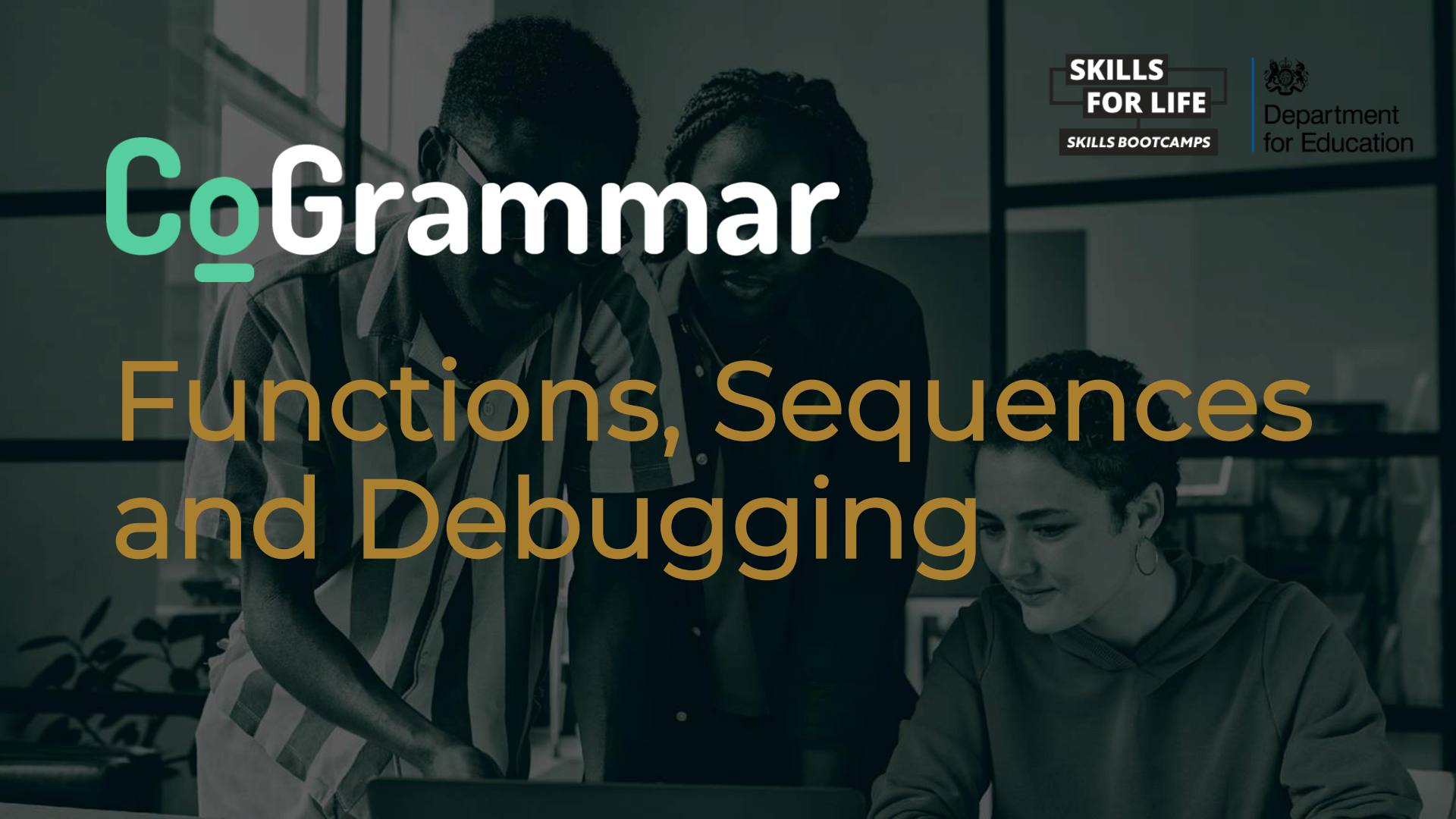
5. **What is a lambda function in Python?**

   A. A function without a name.

   B. A function defined using the def keyword.

   C. A function that always returns None.

   D. A recursive function.


SKILLS FOR LIFE
SKILLS BOOTCAMPS
Department for Education
CoGrammar
Functions, Sequences and Debugging

Learning Outcomes

- Define and call functions with parameters and return values.
- Implement functions to modularise and organise code effectively.
- Apply scope rules to avoid common errors related to variable access and modification.
- Interpret stack traces to debug and identify the source of errors in their code.

Learning Outcomes

- Perform basic string operations such as concatenation, slicing, and formatting.
- Use built-in string methods to manipulate and analyse text data.
- Perform basic list operations such as indexing, slicing, appending, and removing elements.
- Use list methods and functions to manipulate and process list data.

Functions & Scope

CoGrammar

Functions Recap

- In Python, a function is like a recipe or a set of instructions that you can use over and over again in your code. It's a way to group together a block of code that performs a specific task.
- In Python, there are two main types of functions: built-in functions and user-defined functions.

Functions Recap cont.

- **Built-in functions:** These are functions that are already provided by Python, so you can use them without having to define them yourself. For example, print(), len(), and input() are all built-in functions.
- **User-defined functions:** These are functions that you create yourself to perform a specific task in your code. You use the def keyword to define these functions. You give your function a name, specify any parameters it needs, and then write the code that the function should execute when it's called.

Scope

- Scope is a set of rules that determine where and how you can access variables in your code.
- **Global Scope:** Variables that are defined outside of any function are said to be in the global scope. This means they can be accessed from anywhere in your code, including inside functions.
- **Local Scope:** Variables that are defined inside a function are said to be in the local scope. This means they can only be accessed within that specific function.

Understanding scope is crucial because it helps you keep your code organised and prevents unexpected errors caused by variables being accessed in the wrong places.

Stack Traces & Debugging

CoGrammar

Stack Traces

- In programming, a stack trace is like that treasure map. It helps you trace your steps through your code to find out what went wrong when an error occurs.
- When an error happens in your Python code, Python creates a **stack trace** to help you debug it.
- The stack trace shows you the sequence of function calls that led to the error, starting from the main program and going deeper into any functions or methods that were called along the way.

Stack Traces & Debugging

- Stack traces are incredibly helpful for debugging because they give you valuable information about what caused an error and where to look to fix it. This is usually where debugging comes in handy.
- Debugging is the process of finding and fixing errors, or "bugs," in computer programs. Just like detectives investigate and solve mysteries, programmers debug to track down and eliminate problems in their code.
- Debugging is an essential skill for programmers, as it allows them to create software that works correctly and reliably.

Handling Strings

CoGrammar

What is String Handling?

- Strings are sequences of characters, such as letters, numbers, and symbols, which are often used to represent words, sentences, or other textual information.
- String handling refers to the manipulation and management of text data in a programming context.
- Efficient and effective string handling is essential for tasks such as text processing, data parsing, user input validation, and generating formatted output.

String Handling Recap

- We create strings by enclosing characters within either single (' ') or double (" ") quotes.
- Combining strings together is called concatenation. You can use the + operator to concatenate strings
- However, Python offers multiple ways to format strings, such as using the format() method or f-strings.
- To access (index) individual characters in a string we make use of square brackets [].

String Handling Recap

- Python provides many built-in methods to manipulate strings, such as:
  - **len():** Returns the length of a string.
  - **upper(), lower():** Convert strings to uppercase or lowercase.
  - **strip():** Removes leading and trailing whitespace.
  - **split():** Splits a string into a list of substrings based on a delimiter.
  - **join():** Joins elements of a list into a single string using a specified delimiter.

Lists & Dictionaries

CoGrammar

What are Lists?

- Lists are fundamental data structures in Python and are widely used in various programming tasks, from simple data storage to complex data manipulation and analysis.
- They are incredibly versatile and widely used in programming for storing and manipulating data.
- Mastering list manipulation techniques is essential for effective programming in Python.

Manipulating Lists

- **Creating Lists:** Lists are created by enclosing a comma-separated sequence of items within square brackets [ ].
- **Accessing Elements:** You can access individual elements of a list using zero-based indexing (counting starts from 0).
- **Slicing:** Slicing allows you to extract a portion of a list by specifying a start and end index.
- **Modifying Lists:** Lists are mutable, meaning you can change their elements after they've been created.

Manipulating Lists cont.

- **Adding Elements:** You can append new elements to the end of a list using the append() method or insert them at a specific position using the insert() method.
- **Removing Elements:** You can remove elements from a list using methods like remove(), pop(), or del.
- **List Methods:** Python provides many built-in methods for working with lists, such as sort(), reverse(), count(), and index().

Dictionaries

- **Dictionaries** in Python are like real-life dictionaries; they store pairs of keys and their associated values.
- Each value in a dictionary is accessed by its corresponding key, making dictionaries useful for mapping relationships between different pieces of data.
- **Dictionaries** are incredibly useful for representing structured data, such as information about users, configuration settings, or any other data that can be organised into key-value pairs.

Manipulating Dictionaries

- **Accessing Elements:** Values in a dictionary are accessed by their keys.
- **Modifying Dictionaries:** Dictionaries are mutable, so you can change their values, add new key-value pairs, or remove existing ones.
- **Dictionary Methods:** Python provides various built-in methods for working with dictionaries, such as keys(), values(), items(), get(), pop(), and update().

Let’s take a short break

CoGrammar

Demo Time!

CoGrammar

Conclusion and Recap

In this lesson, we explored the importance and application of functions and sequences in Python programming.

- Importance of Functions:

  Purpose: Encapsulate code into reusable blocks.

  Benefits: Enhance code organisation, maintainability, and error management.

- Importance of Sequences:

  Types: Lists, tuples, and strings.

  Uses: Store and manipulate ordered collections of data.

Conclusion and Recap

- Combining functions with sequences allows for flexible and dynamic data handling.
- Key operations include searching, sorting, and transforming data.

Resources

- Official Python Documentation - Functions:
  - https://docs.python.org/3/tutorial/controlflow.html#defining-functions
- Online Tutorials:
  - https://realpython.com/defining-your-own-python-function/
  - https://www.w3schools.com/python/python_functions.asp
- Additional Reading: "Automate the Boring Stuff with Python" by Al Sweigart
- Python Official Documentation - Data Structures:
  - https://docs.python.org/3/tutorial/datastructures.html
- Online Tutorials:
  - https://realpython.com/python-sequences/

Questions and Answers

CoGrammar

Thank you for attending

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

CoGrammar