



# CoGrammar

## SE PORTFOLIO SESSION 2



**SKILLS  
FOR LIFE**

**SKILLS BOOTCAMPS**



Department  
for Education

# Software Engineering Lecture Housekeeping

---

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly.  
**(FBV: Mutual Respect.)**
- 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 Open Classes.  
You can submit these questions here: [Open Class Questions](#)

## Software Engineering Lecture Housekeeping cont.

---

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

# Progression Criteria

## ✓ **Criterion 1: Initial Requirements**

- Complete 15 hours of Guided Learning Hours and the first four tasks within two weeks.

## ✓ **Criterion 2: Mid-Course Progress**

- Software Engineering: Finish 14 tasks by week 8.
- Data Science: Finish 13 tasks by week 8.

## ✓ **Criterion 3: Post-Course Progress**


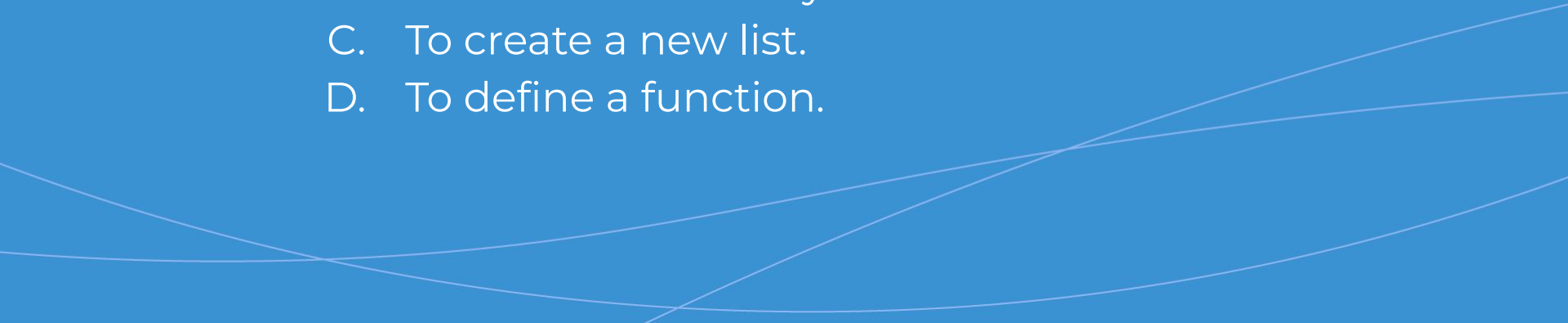
- Complete all mandatory tasks by 24th March 2024.
- Record an Invitation to Interview within 4 weeks of course completion, or by 30th March 2024.
- Achieve 112 GLH by 24th March 2024.

## ✓ **Criterion 4: Employability**

- Record a Final Job Outcome within 12 weeks of graduation, or by 23rd September 2024.


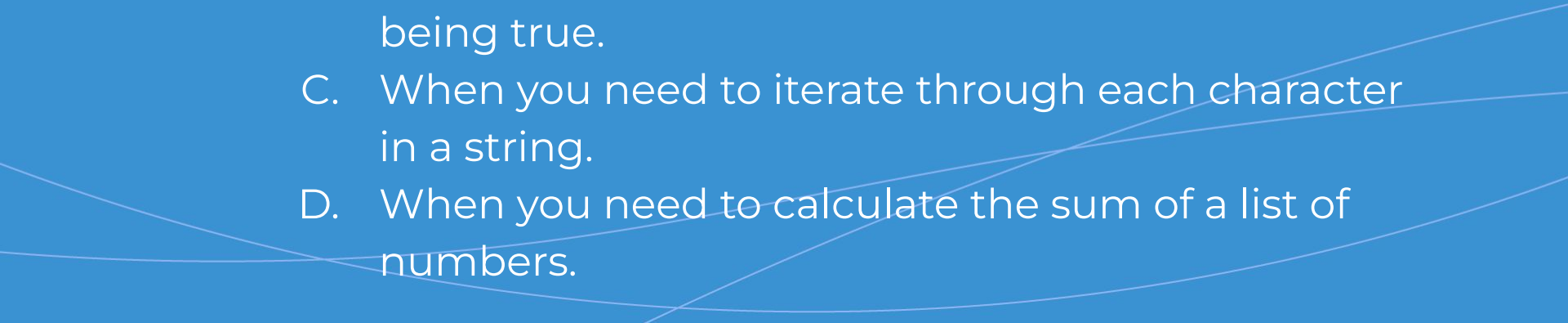


# What is the primary purpose of a for loop in Python?

- 
- A. To repeat a block of code a fixed number of times.
  - B. To check the validity of a condition.
  - C. To create a new list.
  - D. To define a function.
- 

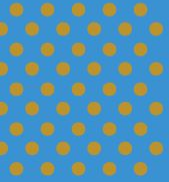
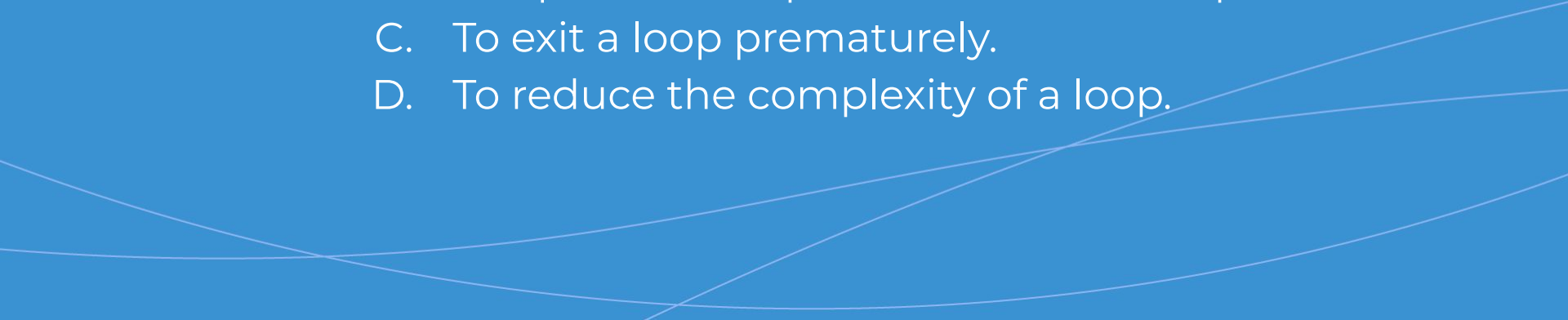


# When is a while loop most appropriate to use?

- 
- A. When you know exactly how many times you need to iterate.
  - B. When you need to iterate based on a condition being true.
  - C. When you need to iterate through each character in a string.
  - D. When you need to calculate the sum of a list of numbers.
- 



# What are nested loops primarily used for in Python?

- 
- A. To repeat a block of code within another loop.
  - B. To optimise the performance of a loop.
  - C. To exit a loop prematurely.
  - D. To reduce the complexity of a loop.
- 

## Recap of Week 2: Iteration

### For Loops

- Used for iterating over items of a collection (like lists or strings) in the order that they appear.

### While Loops

- Execute as long as a specified condition is true, useful for repeated actions until a condition changes.

### Nested Loops

- A loop inside another loop, enabling the handling of multi-layered data structures.



## Recap of Week 2: Iteration

### Controlling Loop Execution

- **'break'**: Immediately exits the loop, typically used to exit early when a condition is met.
- **'continue'**: Skips the rest of the current loop iteration and moves to the next one, often used to skip over certain items.

```
# Example of a nested loop
for row in range(3): # Outer loop
    for col in range(3): # Inner loop
        print(f"Cell ({row}, {col})")

# Use of break and continue
for number in range(10):
    if number == 5:
        break # Exit loop
    if number % 2 == 0:
        continue # Skip even numbers
    print(number)
```

## Paws n Cart Part II

- **Background:** Valentina, a passionate pet lover, creates a specialized online shopping cart for pet products.
- **Challenge:** To revamp the cart, making it dynamic and user-centric, allowing pet owners to add or remove items as their shopping needs evolve.
- **Objective:** Upgrade the cart using loop constructs:
  - Implement "while" loops for continuous item management, mirroring real-world shopping patterns.
  - Use "for" loops to display current cart items, enhancing user experience.
  - Tackle common shopping scenarios, like changing orders and managing multiple items.

## Paws n Cart Part II

- **Programming Needs:**
  - Dynamic item addition/removal using "while" loops.
  - Efficient item display and management with "for" and nested loops.
  - Robust handling of user input and cart modifications.

# Iterating over Pet Products

```
pet_products = ["Dog Food", "Cat Toy", "Fish Bowl"]  
for product in pet_products:  
    print(product)
```

We initialise a 'pet\_products' array containing 3 items, then loop over each product to print it out for display.

# Demo: Using While Loops for Dynamic Cart Management

- The loop continues to prompt the user for input until they choose to finish. It allows adding and removing items from the cart, demonstrating a real-world application of while loops for user interaction:

```
# Add/Remove Items Using While Loop
while True:
    action = input("Enter 'add' to add item, 'remove' to remove item, or 'done' to finish: ")
    if action == 'done':
        break
    elif action == 'add':
        item = input("Enter item to add: ")
        cart.append(item)
    elif action == 'remove':
        if cart:
            item = input("Enter item to remove: ")
            if item in cart:
                cart.remove(item)
    else:
        print("Your cart is empty.")
```

# Demo: Trace Table for Debugging

- A trace table helps visualize how the state of the cart list changes with each iteration of the while loop. It aids in understanding the flow of the program and the impact of each loop iteration on the cart's contents:

Iteration	Action	Input	Cart Contents
1	'add'	'Dog Food'	['Dog Food']
2	'add'	'Cat Toy'	['Dog Food', 'Cat Toy']
3	'done'	-	['Dog Food', 'Cat Toy']

# Valentina's Shopping Cart

Your challenge is to develop a dynamic shopping cart system using Python loops for Valentina's online pet store.

Here is the example data to simulate cart interactions:

Pet Food Items
'Catnip'
'Dog Leash'
'Fish Food'
'Bird Seeds'
'Hamster Wheel'

## Step-by-Step Tasks:

1. **Data Structure:** Initialize an array to represent the list of pet food items in the store.
2. **For Loop Display:** Create a for loop to display all the pet food items available for purchase.
3. **While Loop for Cart Management:** Write a while loop to let users add items to their cart or remove them before checkout.
4. **Nested Loops for Quantity Management:** Use nested loops to allow users to specify quantities for each item (even though you might only simulate this with print statements).

## Advanced Challenge:

- Build upon the cart functionality to enable users to modify the quantity of items in their cart, ensuring that your program can handle these changes accurately.

# Summary

---

## Fundamentals of Iteration

- ★ While loops are ideal for situations where the number of iterations is not predetermined, such as continuous user input.
- ★ Loops within loops useful to handle multi-layered data, crucial for complex data analysis tasks.
- ★ Explored how to use break to exit loops and continue to skip to the next iteration, providing greater control over loop execution.

## Skills Gained

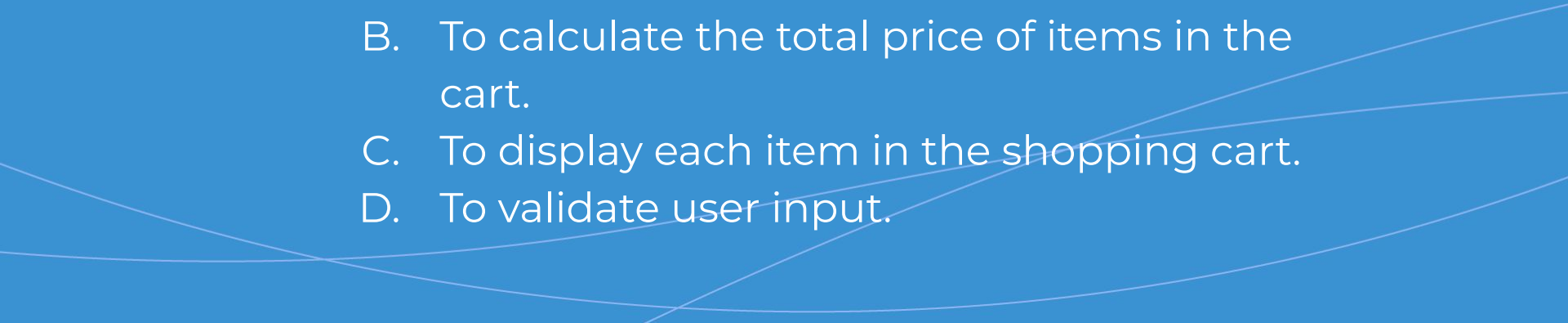
- ★ Enhanced problem-solving skills through the implementation of loop constructs.
- ★ Improved debugging techniques and understanding of program flow with trace tables.





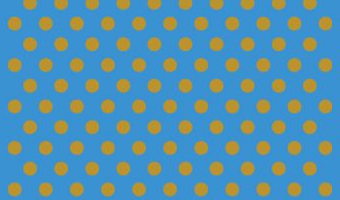
# In the "Paws n Cart Part 2" case study, how can a for loop be effectively used?



- A. To continuously prompt the user until they decide to exit.
  - B. To calculate the total price of items in the cart.
  - C. To display each item in the shopping cart.
  - D. To validate user input.
- 



# Which of the following is true about using if statements within a loop?

- A. If statements can only be used outside of loops.
  - B. Using if statements within a loop can help in making decisions based on each element in the loop.
  - C. If statements within loops always terminate the loop.
  - D. If statements negate the need for a loop.
- 



# Questions and Answers

Questions around the Case Study

