Solution Review: Multiple Asynchronous Calls

This lesson will give a detailed review of how to call an asynchronous function multiple times.

WE'LL COVER THE FOLLOWING

^

Solution: Import the asyncio Library and Call the Asynchronous Coroutine

Solution: Import the **asyncio** Library and Call the Asynchronous Coroutine

- Import the library import asyncio
- Define the function

Asynchronous functions are declared with async def.

```
import asyncio
async def sum(n1, n2):
await asyncio.sleep(1)
return
```

- Call the asynchronous coroutine
 - 1. Create an event loop

```
loop = asyncio.get_event_loop()
```

2. Run async function and wait for completion

```
results = loop.run_until_complete(asyncio.gather(
    sum(n1, n2)
    sum(n1, n2)
    sum(n1, n2))
```

2 Close the loop

3. Close the loop

```
loop.close()
```

The following python code explains the concept.

```
import asyncio
                                                                                        6
async def sum(n1, n2):
   print('Sum numbers', n1, '+', n2)
   await asyncio.sleep(1)
   print('End Sum', n1, '+', n2)
    return n1 + n2
# Create event loop
loop = asyncio.get_event_loop()
# Run async function and wait for completion
results = loop.run_until_complete(asyncio.gather(
   sum(1, 2),
   sum(2, 3),
   sum(3, 4)
))
print(results)
# Close the loop
loop.close()
```

Now that you have an insight into asynchronous programming, let's move on to the quiz.