Yes, **Python can handle very large numbers** natively without any special libraries.

Key Points:

- Python integers (`int`) have **arbitrary precision**, meaning they can grow as large as your memory allows.
- No need for special types like `long` (which existed in Python 2). In Python 3, all integers are of type `int` , and they automatically expand to accommodate big values.

Example:

python Copy Edit

```python
a = 10**100 # This is a 1 followed by 100 zeros print(a) print(type(a)) # <class 'int'>
```

Output:

javascript Copy Edit

```
10000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
<class 'int'>
```

Performance Note:

- While Python handles large numbers well, arithmetic operations on extremely large numbers may be slower than on small integers.
- For high-performance numerical computing, consider using **NumPy**, but its `int` types have fixed sizes (e.g., `int64`).

Let me know if you're dealing with large numbers in a specific context (e.g., finance, cryptography, NEPSE data, etc.)