# Python 3.8 & Core Concepts — Q&A (Set 15)

## Q1. What are the new features added in Python 3.8?

- Assignment expressions (walrus operator :=).  
- Positional-only parameters (with /).  
- f-string debugging support (f"{expr=}").  
- math.prod(), math.isqrt(), statistics.fmean().  
- Shared memory in multiprocessing.  
- TypedDict in typing.  
- Many optimizations and minor syntax/runtime features.

## Q2. What is monkey patching in Python?

Monkey patching = dynamically modifying classes or modules at runtime.  
Example:  
class A:  
 def greet(self): print('Hello')  
  
def new\_greet(self): print('Hi, patched!')  
  
A.greet = new\_greet  
A().greet() # Hi, patched!  
  
Useful for testing but risky for maintainability.

## Q3. Difference between shallow copy and deep copy?

- Shallow copy: new container, but inner objects are shared references.  
 import copy  
 lst = [[1,2],[3,4]]  
 shallow = copy.copy(lst)  
 shallow[0][0] = 99 → lst changes too.  
  
- Deep copy: recursively copies inner objects.  
 deep = copy.deepcopy(lst)  
 deep[0][0] = 7 → lst unaffected.

## Q4. What is the maximum possible length of an identifier?

In Python 3.x, identifiers are practically unlimited (bounded by memory). PEP 8 recommends ≤ 79 chars for readability, but technically you can use much longer.

## Q5. What is generator comprehension?

A generator comprehension creates a generator (lazy iterator) instead of a list.  
Example:  
gen = (x\*\*2 for x in range(5))  
next(gen) → 0  
next(gen) → 1  
  
Syntax is like list comprehension but uses parentheses (). Advantage: memory efficient, produces items on demand.