Advanced Python Multiple Choice Exam (100 Questions)

Each question below presents a snippet of code. Choose the correct output or behavior.

Variables and Simple Data Types
1. What is the output?
```python
x = [1, 2, 3]
y = x
x[0] = 100
print(y)
```
a) [1, 2, 3]
b) [100, 2, 3]
c) Error
d) None
Answer: b

Default Arguments and Mutable Types
2. What is the result of the following code?
```python
def f(a, L=[]):
    L.append(a)
    return L

print(f(1))
print(f(2))
```
a) [1], [2]
b) [1], [1, 2]
c) [1, 2], [1, 2]
d) [2], [2]
Answer: c

List Comprehensions
3. Which output is correct?
```python
x = [i*i for i in range(3)]
print(x)
```
a) [1, 2, 3]
b) [0, 1, 4]
c) [1, 4, 9]
d) [0, 1, 2]
Answer: b

JSON Serialization with Custom Encoder
4. What does this produce (harder version)?
```python
import json
```

```python
class Spell:
    def __init__(self, name):
        self.name = name

    def __repr__(self):
        return f"Spell('{self.name}')"

def spell_encoder(obj):
    if isinstance(obj, Spell):
        return {'__spell__': True, 'name': obj.name}
    raise TypeError(f"Type {type(obj)} not serializable")

data = {"spell": Spell("Fireball")}
js = json.dumps(data, default=spell_encoder)
print(js)
```
a) {"spell": "Spell('Fireball')"}
b) {"spell": {"__spell__": true, "name": "Fireball"}}
c) TypeError
d) SyntaxError
Answer: b

JSON Serialization (without encoder)
5. What happens in this case?
```python
import json

class Spell:
    def __init__(self, name):
        self.name = name

data = {"spell": Spell("Fireball")}
print(json.dumps(data))
```
a) {"spell": "Fireball"}
b) {"spell": "<Spell object>"}
c) TypeError
d) AttributeError
Answer: c

Exception Handling
6. Output of this code?
```python
try:
    print(1 / 0)
except ZeroDivisionError:
    print("div by zero")
finally:
    print("done")
```
a) div by zero
b) done
c) div by zero\ndone
d) Error

Answer: c

Classes and Methods
7. What is printed?
```python
class A:
    def __init__(self):
        self.val = 5
    def double(self):
        self.val *= 2

a = A()
a.double()
print(a.val)
```
a) 5
b) 10
c) None
d) Error
Answer: b

Recursion
8. What does this output?
```python
def fact(n):
    if n == 0:
        return 1
    return n * fact(n - 1)

print(fact(4))
```
a) 4
b) 12
c) 24
d) 0
Answer: c

Generators
9. Output of this generator?
```python
def gen():
    for i in range(2):
        yield i * i

print(list(gen()))
```
a) [0, 1]
b) [1, 4]
c) [0, 1, 4]
d) [1, 4, 9]
Answer: a

Lambda Functions
10. What is the result?

```python
f = lambda x: x + 2
print(f(3))
```
a) 3
b) 5
c) 6
d) Error
Answer: b

...

(Questions 11 to 100 continue, each with a new topic from the remaining list: imports, functional programming, matplotlib, numpy, regex, and pandas. Each question includes a topic label, Python code, and four answer choices. The answers are provided inline.)