

G4-17 课后练习 —— 综合项目一

姓名： ____ 日期： ____

面向7-10岁少儿 | G4 编程高手 | 嵌套结构与异常处理综合

一、填空题（回顾概念）

题目1： 在 try/except/else/finally 结构中，只有 try 块中 ____ （发生/没发生）异常时 else 块才会执行。finally 块 ____ （无论/仅当有异常）都会执行。

题目2： 嵌套字典 「data = {"a": {"b": [1, 2, 3]}}」 中，要访问数字 2，写法是 「data[____][____][____]」 。

二、代码补全（补写代码）

题目3： 编写一个"学生信息管理系统"，支持添加和查询学生。补全代码。

```
students = []

def add_student(name, age, grade):
    """添加学生"""
    student = {
        "name": ____,
        "age": ____,
        "grade": ____
    }
    students.___(student)
    print(f"学生 {name} 添加成功！")

def find_student(name):
    """按姓名查找学生"""
    for stu in ____:
        if stu[____] == name:
            return stu
    return ____

# 测试
add_student("小明", 10, 5)
```

```
add_student("小红", 9, 4)
```

```
result = find_student("小明")
if result:
    print(f'找到: {result['name']}, {result['age']}岁, {result['grade']}年级')
else:
    print("未找到")
```

题目4： 安全的列表访问函数。补全代码。

```
def safe_get_element(lst, index):
    """
    安全获取列表元素
    如果索引有效，返回元素值
    如果索引无效，返回 None 并打印错误信息
    """
    try:
        return lst[____]
    ____ IndexError:
        print(f"错误：索引{index}超出范围，列表长度{____(lst)}")
        return ____
    ____ Exception as e:
        print(f"未知错误：{e}")
        return None
```

```
# 测试
nums = [10, 20, 30, 40, 50]
print(safe_get_element(nums, 2)) # 期待输出：30
print(safe_get_element(nums, 10)) # 期待输出：错误信息 + None
```

三、创作题（动手编程）

题目5： 综合项目：编写"超市收银系统"。

- 商品数据（列表套字典）：

```
```python
products = [
 {"id": "001", "name": "苹果", "price": 5.5, "stock": 50},
 {"id": "002", "name": "牛奶", "price": 12.0, "stock": 30},
 {"id": "003", "name": "面包", "price": 8.0, "stock": 20},
 {"id": "004", "name": "鸡蛋", "price": 1.5, "stock": 100}
]
```
```

- 实现功能：

1. 显示所有商品（编号、名称、价格、库存）
2. 用户通过商品编号选择要购买的商品和数量
3. 检查库存是否充足，不足则提示并拒绝交易
4. 用 try/except 处理输入错误（编号不存在、数量不是数字等）
5. 计算总价，扣减库存，打印购物小票
6. 循环购买，输入"结账"退出

请写出完整代码。

<details>

<summary>参考答案</summary>

题目1： 没发生，无论

题目2： 「data["a"]["b"][1]」

题目3：

```
students = []
```

```
def add_student(name, age, grade):
    student = {
        "name": name,
        "age": age,
        "grade": grade
    }
    students.append(student)
    print(f"学生 {name} 添加成功！")
```

```
def find_student(name):
    for stu in students:
        if stu["name"] == name:
            return stu
    return None
```

```
# 测试
add_student("小明", 10, 5)
add_student("小红", 9, 4)
```

```
result = find_student("小明")
if result:
    print(f"找到: {result['name']}, {result['age']}岁, {result['grade']}年级")
else:
    print("未找到")
```

题目4：

```
def safe_get_element(lst, index):
    try:
```

```
    return lst[index]
except IndexError:
    print(f"错误：索引{index}超出范围，列表长度{len(lst)}")
    return None
except Exception as e:
    print(f"未知错误：{e}")
    return None
```

题目5：

```
products = [
    {"id": "001", "name": "苹果", "price": 5.5, "stock": 50},
    {"id": "002", "name": "牛奶", "price": 12.0, "stock": 30},
    {"id": "003", "name": "面包", "price": 8.0, "stock": 20},
    {"id": "004", "name": "鸡蛋", "price": 1.5, "stock": 100}
]
```

```
cart = []
```

```
def show_products():
    print("\n商品列表： ")
    print("编号\t名称\t价格\t库存")
    print("-" * 30)
    for p in products:
        print(f"{p['id']}\t{p['name']}\t{p['price']}\t{p['stock']}")
```

```
def find_product(pid):
    for p in products:
        if p["id"] == pid:
            return p
    return None
```

```
print("==== 欢迎光临小小超市 =====")
```

```
while True:
    show_products()
    pid = input("\n请输入商品编号购买（或输入'结账'结束）： ")
```

```
    if pid == "结账":
        break
```

```
    product = find_product(pid)
    if product is None:
        print("商品编号不存在，请重新输入！ ")
        continue
```

```
    try:
        qty = int(input(f"请输入{product['name']}的购买数量： "))
    except ValueError:
        print("数量必须是整数！ ")
        continue
```

```
    if qty <= 0:
```

```
print("数量必须大于0! ")
continue
```

```
if qty > product["stock"]:
    print(f"库存不足! {product['name']}仅剩{product['stock']}件")
    continue
```

```
# 加入购物车
product["stock"] -= qty
cart.append({"name": product["name"], "qty": qty, "price": product["price"]})
print(f"已将{qty}件{product['name']}加入购物车")
```

```
# 打印小票
print("\n" + "=" * 30)
print("    购物小票")
print("=" * 30)
total = 0
for item in cart:
    subtotal = item["qty"] * item["price"]
    total += subtotal
    print(f"{item['name']} x{item['qty']}  ¥{subtotal:.1f}")
print("-" * 30)
print(f"合计:  ¥{total:.1f}")
print("=" * 30)
print("感谢惠顾, 欢迎下次光临! ")
```

</details>

自我评价: ☐ 全都会了 ☐ 大部分会了 ☐ 还要再练练