# tests.py  
from operation import (  
 books, members,  
 add\_book, add\_member, borrow\_book, return\_book,  
 delete\_book, delete\_member  
)  
  
# Reset data (simple reset by clearing lists/dicts)  
books.clear()  
members.clear()  
  
# 1) Add book and member  
assert add\_book("ISBN001", "Python Basics", "Alice Author", "Non-Fiction", copies=3) is True  
member\_id = add\_member("John Doe", "john@example.com")  
assert isinstance(member\_id, int)  
  
# 2) Borrow book -> available should reduce  
assert borrow\_book(member\_id, "ISBN001") is True  
assert books["ISBN001"]["available\_copies"] == 2  
assert "ISBN001" in members[0]["borrowed\_books"]  
  
# 3) Prevent double-borrow by same member  
assert borrow\_book(member\_id, "ISBN001") is False # already borrowed  
  
# 4) Return book -> available increases  
assert return\_book(member\_id, "ISBN001") is True  
assert books["ISBN001"]["available\_copies"] == 3  
assert "ISBN001" not in members[0]["borrowed\_books"]  
  
# 5) Delete book only when no copies borrow\_book  
# borrow and attempt delete  
borrow\_book(member\_id, "ISBN001")  
assert delete\_book("ISBN001") is False # can't delete, one copy is borrowed  
# return and delete  
assert return\_book(member\_id, "ISBN001") is True  
assert delete\_book("ISBN001") is True  
  
# 6) Member delete safety  
assert delete\_member(member\_id) is True # member has no borrowed books now  
  
print("All asserts passed.")