在Python中使用pickle对对象进行序列化与反序列化的时候，如果对象A的其中一个域是另一个对象B的指针，并且对象A与对象B被序列化在不同的文件当中时，反序列化后,对象A的域是一个对象，而对象B会是另外一个新的对象。

示例代码:

import pickle  
class Test:  
 def \_\_init\_\_(self, name, target):  
 self.name = name  
 self.target = target

a = Test("A", "")  
b = Test("B", a)  
print(id(a), id(b.target))  
print(a.name, b.target.name)  
a.name = "ChangedA"  
print(a.name, b.target.name)  
print("========Now Dump And Reload=========")  
f = open("record", "wb")  
pickle.dump(a, f)  
f.close()  
f = open("record2", "wb")  
pickle.dump(b, f)  
f.close()  
f = open("record", "rb")  
a = pickle.load(f)  
f.close()  
f = open("record2", "rb")  
b = pickle.load(f)  
f.close()  
print(a.name, b.target.name)  
a.name = "ChangeAgainA"  
print(id(a),id(b.target))  
print(a.name, b.target.name)

输出的结果为：

2150559372680 2150559372680

A A

ChangedA ChangedA

========Now Dump And Reload=========

ChangedA ChangedA

2150559454600 2150559463856

ChangeAgainA ChangedA

🡪在序列化前，改变对象a的name属性时，b对象的target域所对应的name属性也会改变。当序列化以及反序列化以后，b.target与对象a不是同一个对象。从而改变a的name属性时，b的target域中的name属性不会变化。