

# به نام خدا

## تمرین کامپیوتری اول – تست نرم افزار – بخش تئوری

امیرحسین راحتی 810100144

علی ممتحن 810100213

<https://github.com/AmirhosseinRHT/Software-Testing>

لینک مخزن پروژه :

e58fb809736714104c4de94347a24b761a382ef4

شناسه آخرین کامیت:

### • پاسخ سوالات بخش تئوری

(۱) در State Verification ، ما وضعیت نهایی سیستم را پس از اجرای عملیات بررسی می کنیم. یعنی، نتیجه و خروجی نهایی مورد ارزیابی قرار می گیرد. نه جزئیات هر مرحله از اجرا

در Behavior Verification ، رفتار و تعاملات بین اجزای سیستم مورد بررسی قرار می گیرد. یعنی اینکه آیا متدها با پارامترهای درست و به ترتیب صحیح فراخوانی شده اند.

در State Verification معمولاً از Stub استفاده می کنیم چون فقط نیاز داریم داده های از پیش تعیین شده برگردانده شوند .

در Behavior Verification از Mock استفاده می کنیم زیرا نیاز به بررسی دقیق تعاملات و فراخوانی ها داریم.

(۲) Test Spy ها ابزارهایی هستند که رفتار و تعاملات حین اجرای برنامه را ثبت می کنند تا بعداً بتوانیم آن ها را بررسی کنیم. انواع اصلی Test Spy ها عبارتند از:

a. Method Call Spy: فراخوانی متدها و پارامترهای آن ها را ثبت می کند. می تواند به صورت partial هم باشند . یعنی بخشی از یک تابع این خاصیت را داشته باشد و بقیه تابع به کار خودی خود بپردازد

b. Property Spy: دسترسی و تغییرات property ها را ثبت می کند

c. Mock Spy: ترکیبی از mock و spy است که به تست کننده امکان می دهد تا علاوه بر ثبت فراخوانی ها، رفتار خاصی را نیز تعریف کند که در صورت نیاز فعال شود.

d. Event Spy: event ها و پارامترهای آن ها را ثبت می کند

Test Spy ها برای موارد زیر استفاده می شوند:

- بررسی ترتیب فراخوانی متدها
- ثبت پارامترهای ورودی
- بررسی تعداد دفعات فراخوانی یک متد

(۳)

a. وقتی ایجاد fixture زمان بر است - وقتی fixture تغییر نمی کند و بین تست ها مشترک است - وقتی حجم داده های fixture زیاد است

b. مزایای Lazy Setup : مصرف منابع کمتر - اجرای سریع تر تست ها - خوانایی بیشتر کد - کاهش فشار به سیستم برای لود دیتای بلا استفاده - کاهش زمان اجرا

معایب Lazy Setup : پیچیدگی بیشتر در مدیریت - احتمال تکرار کد - نیاز به کنترل بیشتر و checking بیشتر در کد  
c. استفاده از الگوی Builder برای ساخت fixture - ایجاد یک نسخه جدید از fixture برای هر تست - استفاده از ترنزکشن ها برای بازگرداندن وضعیت قبلی : در این روش ، با تعیین یک شرط ، خود برنامه در صورت برآورده شدن آن شرط ، برنامه را به حالت قبل اجرای آن متد میبرد (در اسپرینگ وجود دارد) - پیاده سازی مکانیزم Rollback برای بازگشت به حالت اولیه

## ● باگ های کد

(۱) پارامتر های متد های Reservation Controller چک نمیشد که مقادیر درستی داشته باشند.