

دستور کار آزمایشگاه مهندسی نرم افزار

دانشگاه صنعتی شریف

دانشکده مهندسی کامپیوتر

عنوان آزمایش: پوشش آزمون (Coverage)

• اهداف

- آشنایی دانشجویان با مفهوم پوشش آزمون با کمک یک برنامه ساده

• نیازمندی‌ها

- آشنایی با یک زبان برنامه‌نویسی مانند جاوا

• ابزارهای مورد استفاده

- IntelliJ IDEA (ترجیحاً نسخه 2018 به بعد)

- JUnit

• روال انجام آزمایش

این آزمایش به مفاهیم پوشش آزمون (Coverage) می‌پردازد. به‌طور کلی در این آزمایش منظور از پوشش آزمون یا تست آن است که مشخص کنیم موارد آزمون (Test Cases) تا چه اندازه کد اصلی برنامه را مورد آزمون قرار می‌دهند و در واقع چند درصد از کد برنامه توسط این موارد، اجرا و تست می‌شوند. پوشش آزمون می‌تواند توسط معیارهایی نظیر تعداد کلاس‌ها، متدها و خطوط بررسی شود و به‌صورت خودکار از طریق ابزارهایی قابل محاسبه است.

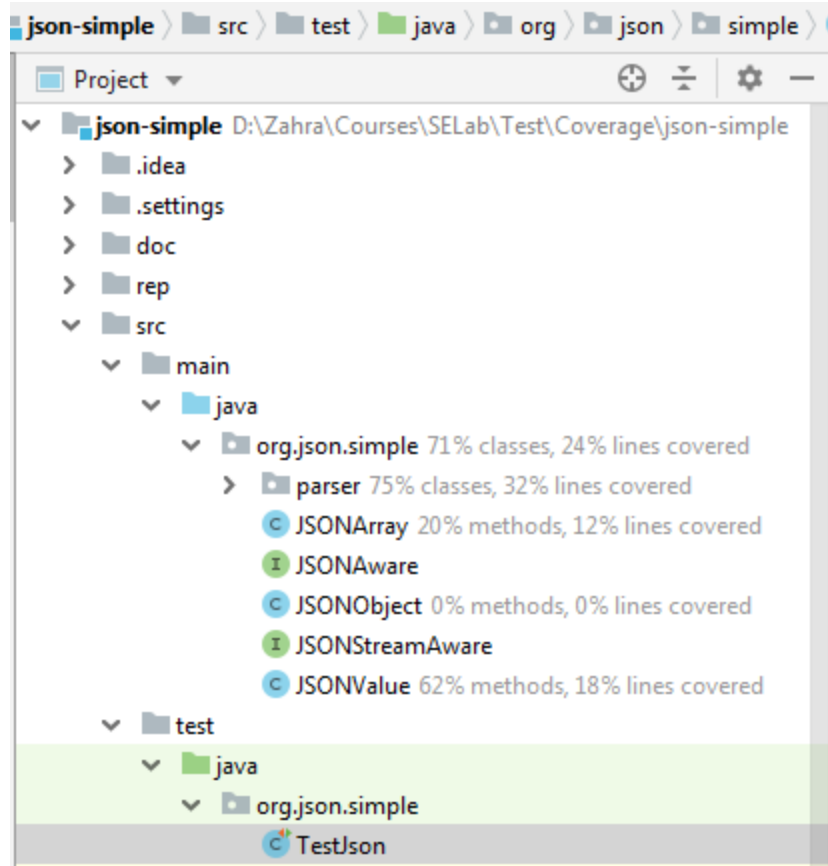
۱. پروژه json-simle را در محیط IntelliJ باز (import) کنید. در صورتی که فایل‌های برنامه با خطا مواجه شدند و کتابخانه Test را شناسایی نکرده‌اند باید JUnit4 را اضافه کرد. SDK هم از قبل باید نصب شده باشد. ساختار پروژه شامل دو پکیج اصلی main و test است که حاوی کدهای برنامه و کد کلاس تست می‌باشد.

۲. کلاس تست TestJson شامل متدهایی جهت تست کلاس‌های اصلی است ولی این متدها همه کلاس‌ها را تست نمی‌کنند. برای یافتن اینکه چند درصد از کلاس‌ها، متدها و خطوط برنامه اصلی توسط TestJson مورد تست قرار می‌گیرند بر روی کلاس تست، کلیک راست کرده و گزینه Run TestJson with Coverage را انتخاب می‌کنیم. (اگر درصدهای Coverage نشان داده نشده است باید قسمت Edit Configuration را از پنل Coverage کلیک کرد و سپس در قسمت مربوط به Coverage گزینه مربوط به ذخیره خروجی Coverage را انتخاب کرد.)

The screenshot shows the IntelliJ IDEA interface. On the left, the 'Project' view shows the 'test' directory. A right-click context menu is open over the 'Test' class, with 'Run TestJson with Coverage' selected. The main editor shows the source code of 'TestJson.java'. On the right, the 'Coverage: TestJson' tool window is open, displaying a table of coverage data for the package 'org.json.simple'.

Element	Class, %	Method, %	Line, %
parser	75% (3/4)	40% (20/49)	32% (172/531)
JSONArray	100% (1/1)	20% (5/25)	12% (22/175)
JSONObject	0% (0/1)	0% (0/9)	0% (0/41)
JSONValue	100% (1/1)	62% (5/8)	18% (21/114)

Below the table, it states: 'Tests passed: 2 of 2 tests - 78 ms'. The bottom status bar shows '29:22 CRLF UTF-8 Tab' and '10:54 AM'.



Coverage: TestJson x			
71% classes, 24% lines covered in 'all classes in scope'			
Element	Class, %	Method, %	Line, %
org.json.simple	71% (5/7)	32% (30/91)	24% (215/861)

Coverage: TestJson x

71% classes, 24% lines covered in package 'org.json.simple'

Element	Class, %	Method, %	Line, %
parser	75% (3/4)	40% (20/49)	32% (172/531)
JSONArray	100% (1/1)	20% (5/25)	12% (22/175)
JSONObject	0% (0/1)	0% (0/9)	0% (0/41)
JSONValue	100% (1/1)	62% (5/8)	18% (21/114)

Ant Build
Maven
Database

برای اینکه با روال کار بهتر آشنا شوید، می‌توانید با گذاشتن breakpoint بر روی یکی از متدهای تست، روند debug آن و پیمایش خطوط برنامه را مشاهده کنید. با استفاده از گزینه Generate Coverage Report (فلش زردرنگ در تصویر فوق) می‌توان گزارشی به فرمت html حاوی خلاصه درصدهای پوشش، خطوط پوشش داده شده (با رنگ سبز) و خطوطی که پوشش داده نشده (خطوط قرمز) را مشاهده کرد.

The screenshot shows an IDE with a project named 'json-simple'. The 'Coverage' window is open, displaying the same coverage data as the first image. A 'Generate Coverage Report' dialog box is open, showing the output directory 'TestCoverage\json-simple\rep' and the option 'Open generated HTML in browser' checked. The 'Run' window at the bottom shows the test results for 'TestJson'.

Run: TestJson x

Tests passed: 2 of 2 tests - 78 ms

TestJson (org.json.simple)

- testJSONArrayCollection: 31 ms
- testDecode: 47 ms

include patterns:
org\.json\.simple\.*
exclude patterns:=====decode=====

=====the 2nd element of array=====

10

Class transformation time: 0.072974771s for 499 classes or 1.462420260521042E-4s per class

Process finished with exit code 0

Coverage Report :: Summary - Windows Internet Explorer

D:\Zahra\Courses\SELab\Test\Coverage\json-simple\rep\index.html

Search Bing

Favorites Suggested Sites Web Slice Gallery

Coverage Report :: Summary

[all classes]

Overall Coverage Summary

Package	Class, %	Method, %	Line, %
all classes	71.4% (5/ 7)	33.3% (31/ 93)	25.1% (215/ 857)

Coverage Breakdown

Package	Class, %	Method, %	Line, %
org.json.simple	66.7% (2/ 3)	23.3% (10/ 43)	13.2% (43/ 326)
org.json.simple.parser	75% (3/ 4)	42% (21/ 50)	32.4% (172/ 531)

generated on 2019-05-12 10:56

Computer Protected Mode: Off 105%

با کلیک بر روی لینک هر یک از فایل‌های گزارش، خطوط پوشش داده شده (به رنگ سبز) و نشده (به رنگ قرمز) نمایش داده می‌شوند.

```

    public static void writeJSONString(Collection collection, Writer out) throws IOException{
        if(collection == null){
            out.write("null");
            return;
        }

        boolean first = true;
        Iterator iter=collection.iterator();

        out.write('[');
        while(iter.hasNext()){
            if(first)
                first = false;
            else
                out.write(',');

            Object value=iter.next();
            if(value == null){
                out.write("null");
                continue;
            }

            JSONValue.writeJSONString(value, out);
        }
        out.write(']');
    }

    public void writeJSONString(Writer out) throws IOException{
        writeJSONString(this, out);
    }

```

تمرین:

- روالی که برای پروژه json-simle ارائه شد را برای پروژه CodeCoverageProject نیز انجام دهید:
- ۱- برای به دست آوردن اعداد پوشش آزمون کافی است بر روی پکیج java در مسیر test کلیک راست کرده و گزینه Run All Tests with coverage را انتخاب کنید (نیازی به اجرای عادی پروژه نیست).
- ۲- با افزودن بخش‌هایی به کد تست، اعداد پوشش آزمون را در مورد تمامی کلاس‌های موجود در برنامه بهبود دهید. درصد افزایش اعداد پوشش آزمون مهم نیست ولی بخش‌هایی که به کد تست اضافه می‌شوند باید معنادار باشند و صرفاً یک فراخوانی ساده کلاس یا متد، بدون استفاده در بخش‌های دیگر کد کافی نیست.
- ۳- همراه با ارسال کل پروژه، لازم است گزارشی در مورد اعداد اولیه پوشش آزمون (به صورت screenshot)، بخش‌هایی که به کد تست اضافه کرده‌اید و میزان افزایشی که در اعداد پوشش آزمون به دنبال این تغییرات حاصل شده است را نیز تهیه و ارسال نمایید. توجه داشته باشید که در گزارش باید کاملاً مشخص باشد هر یک از کلاس‌ها چگونه مورد آزمون قرار گرفته‌اند (با درج تصویر از کد و میزان افزایش عدد پوشش آزمون نسبت به قبل به ازای هر کلاس).

- نحوه ارسال پروژه

علاوه بر تحویل پروژه از طریق گیت، یک پوشه با عنوان Coverage-studentID1-studentID2 (حاوی شماره دانشجویی اعضای گروه) ایجاد نموده و پروژه را به همراه گزارش در این پوشه قرار داده و آن را از طریق سامانه CW ارسال نمایید. آدرس مخازن پروژه خود را نیز در گزارش ذکر کنید.