RPG Unit Testing Made Easy in VS Code

Alan Seiden · Principal, Seiden Group

SEIDEN GROUP

IBM i

- Presentation based on material from Sanjula Ganepola (IBM)
  - Software Developer – IBM i App Dev & AI
    Sanjula.Ganepola@ibm.com

- Additional content from Patrick Behr

SEIDEN GROUP

Agenda

- Benefits of Unit Testing
- RPGUnit Testing Tool
- VS Code "IBM i Testing" Extension
- Command line tool (CLI) for IBM i Testing
  - How to run tests outside VS Code?
  - How to run tests in a CI/CD pipeline?
- Can IBM Bob help with unit testing?

Benefits of Unit Testing

- Small, concrete steps
- You know when you're done
- Simpler program design
- All code is tested

- INSPIRES CONFIDENCE AND COURAGE

Test Phases

1. Environment Setup
2. Call the programs/functions being tested
3. Verify results
4. Cleanup

xUnit

Test automation frameworks

SEIDEN GROUP

Test automation frameworks

SUnit - Smalltalk

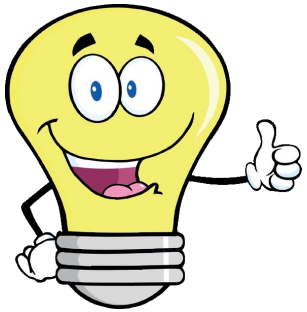
JUnit - Java

PHPUnit – PHP

PyUnit - Python

NUnit - .NET

RPGUnit - RPG



~~RPGUnit~~

iRPGUnit

ILEUnit

IBMiUnit

SEIDEN GROUP

1. Environment setup

- setupSuite( )

  Runs once before the start of any tests

- setup( )

  Runs once before each of your tests

2. Call the program/function being tested

This can be the tricky part…
Getting the program/procedure into a state where it can be easily tested

SEIDEN GROUP

3. Verify results

- Verify return values
- Verify spooled files
- Verify data

Tests Use **Assertions** with boolean pass/fail results
(more information to come)

SEIDEN GROUP

4. Teardown (clean everything up)

- teardown( )

  Runs once after each of your tests

- teardownSuite( )

  Runs once after all tests have run

Assertions using assert()

```
actualValue = AddTwoNumbers(2 : 2);
expectedValue = 4;

assertEqual(expectedValue : actualValue : 'Error message');

 if expectedValue <> actualValue;
    sendFailedMessage('Error message');
    failedTests += 1;
 else;
    passedTests += 1;
 endif;
```

Example Assertions with assert()

Syntax

```
assert( booleanExpression [: 'message' );
```

Example

```
// ✅ Pass
assert( 1=1 : 'Impossible to fail' );

// ❌ Fail
assert( 1=0 : 'Obviously wrong' );
```

Example Assertions with assertEqual()

Syntax

```
assertEqual( 'expected' : 'actual' [: 'message' );
```

Example

```
// ✅ Pass
assertEqual( 'Hello' : 'Hello' );
assertEqual( 123 : 123 );

// ❌ Fail
assertEqual( 123.45 : 123.46 );
assertEqual( D'2024-09-16' : D'2023-01-11' );
assertEqual( T'11.22.33' : T'22.33.44' );
assertEqual( Z'2024-09-16-11.22.33.123456' : Z'2024-01-11-11.22.33.123456' );
```


SEIDEN GROUP

Assertions are Helpful in Test Cases

- A **test case** contains testing code, including assertions
  - Example coming up
- A **test suite** contains multiple test cases

Test Cases

SEIDEN GROUP

Test Case Using Assertions

```
testAddTwoNumbers {

        actualValue = AddTwoNumbers(2 : 2);
        expectedValue = 4;


        assertEqual(expectedValue : actualValue : 'Error message');
        assertEqual(expectedOther : actualOther : 'Error message');
        assertEqual(…etc…);
}
```

SEIDEN GROUP

Test Suite

SEIDEN GROUP

Test Suite with Multiple Test Cases

```
ArithmeticTests {

        setup{…}

        testAddTwoNumbers{…}
        testSubtract{…}
        testDivide{…}
        testMultiply{…}

        teardown{…}
}
```

environment setup

call test, verify results

call test, verify results

call test, verify results

call test, verify results

environment cleanup

SEIDEN GROUP

RPGUnit / iRPGUnit

https://sourceforge.net/projects/irpgunit/

💡 Add the copybook: /include qinclude,TESTCASE

SEIDEN GROUP

Naming requirements

iRPGUnit supports both *SRVPGM and *PGM programs.
At this time, the IBM i Testing Extension supports only service programs.

A **test case** is an exported procedure

- starts with the name test
  Examples:
  test_getEmployeeDetail
  test_getDeptDetail
  testIsPalindrome
  testFactorial

A **test suite** is a service program containing test case procedures

- Test suite program has no naming requirements

VS Code IBM i Testing Extension naming requirements:

- Local source files: must be suffixed with .test.rpgle, .test.sqlrpgle, .test.cblle, or .test.sqlcblle
- Source members: recommended: prefix or suffix with T, _T, T_, etc.

SEIDEN GROUP

Example Includes for RPGUnit

```
**free

ctl-opt  nomain;

/include rpgunit/qinclude,TESTCASE

/include SRVMATH_H
```

SEIDEN GROUP

Test Case Procedure in RPGUnit

```
// ----------------------------------------------------------------
// Add two numbers
// ----------------------------------------------------------------
dcl-proc  test_AddTwoNumbers  export;

   dcl-s actualValue packed(7);
   dcl-s expectedValue  packed(7);

   actualValue = AddTwoNumbers(2 : 2);
   expectedValue = 4;

   iEqual( expectedValue : actualValue);

end-proc  test_AddTwoNumbers;
```

SEIDEN GROUP

Test case (full code)

```
**free
ctl-opt  nomain;
/include rpgunit/qinclude,TESTCASE
/include SRVMATH_H

dcl-proc  test_AddTwoNumbers  export;
   dcl-s actualValue packed(7);
   dcl-s expectedValue  packed(7);

   actualValue = AddTwoNumbers(2 : 2);
   expectedValue = 4;

   iEqual(expectedValue : actualValue);

end-proc  test_AddTwoNumbers;
```

SEIDEN GROUP

Assertion type "assert" in RPGUnit

```
dcl-proc  assert;
   condition      ind  const;
   messageIfFalse  varchar(16384) const;
end-proc  assert;


assert( a > b : 'A is not greater than B' );

assert( %found(FILE) : 'Missing record in file FILE' );
```

SEIDEN GROUP

Assertion type "aEqual" in RPGUnit

```
dcl-proc  aEqual;
  expectedValue   char(32565) const;
  actualValue     char(32565) const;
  fieldname       varchar(64) const options(*nopass: *omit);
end-proc  aEqual;

aEqual('a' : 'b');
// Expected 'a', but was 'b'.

aEqual('a' : 'b' : 'Switch');
// Switch: Expected 'a', but was 'b'.
```

SEIDEN GROUP

Assertion type "iEqual" in RPGUnit

```
dcl-proc  iEqual;
  expectedValue   zoned(31)   const;
  actualValue     zoned(31)   const;
  fieldname       varchar(64) const options(*nopass: *omit); end-proc  iEqual;
end-proc  iEqual;

iEqual(5 : 6);
// Expected 5, but was 6.

iEqual(5 : 6 : 'Widget');
// Widget: Expected 5, but was 6.
```

Assertion type "nEqual" in RPGUnit

```
dcl-proc  nEqual;
   expectedValue  ind  const;
   actualValue    ind  const;
   fieldName      varchar(64) const;
end-proc  nEqual;

nEqual(*ON : *OFF);
// Expected '1', but was '0'.

nEqual (*ON : *OFF : 'Alarm');
// Alarm: Expected '1', but was '0'.
```

SEIDEN GROUP

Utility to get joblog errors

```
monitor;
  a = 10;
  b = 0;
  c = a / b;

  // ❌ Fail: Only runs if exception was not caught
  fail('Division by zero did not raise an error.');
on-error;
  msgInfo = getMonitoredMessage(*ON);
endmon;

// ✅ Pass
aEqual( 'MCH1211' : msgInfo.Id );
```

💡Add the copybook: /include qinclude,TESTCASE

Compile and execution commands

Compile Command:

```
RPGUNIT/RUCRTRPG TSTPGM(MYUSER/TEMPDET) SRCSTMF('/home/MYUSER/builds/
ibmi-company_system/qtestsrc/empdet.test.sqlrpgle') TGTCCSID(*JOB)
DBGVIEW(*SOURCE) COPTION(*EVENTF) RPGPPOPT(*LVL2) INCDIR('''/home/
MYUSER/builds/ibmi-company_system''')
```

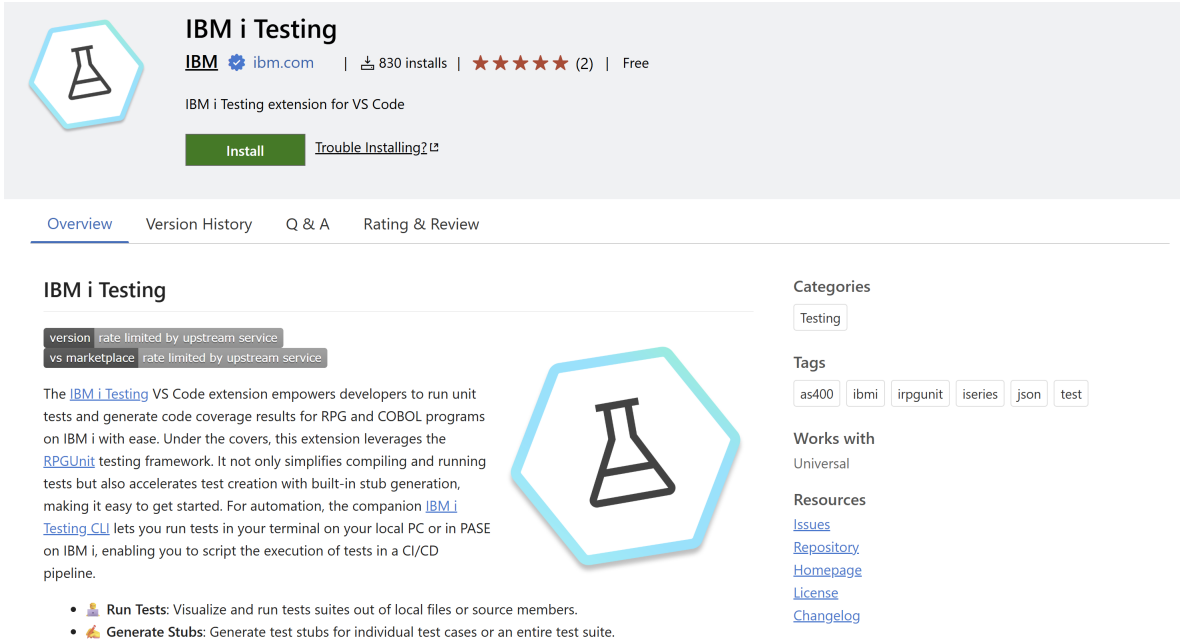
Execution Command:

```
RPGUNIT/RUCALLTST TSTPGM(MYUSER/TEMPDET) ORDER(*REVERSE) DETAIL(*ALL)
OUTPUT(*ALLWAYS) LIBL(*CURRENT) JOBD(*DFT) RCLRSC(*NO) XMLSTMF('/tmp/
testing/vscode-ibmi-testing/RPGUNIT/TEMPDET_1751646914682.xml')
```

IBM i Testing Extension for VS Code

SEIDEN GROUP

Installable via the Marketplace



https://marketplace.visualstudio.com/items?itemName=IBM.vscode-ibmi-testing

SEIDEN GROUP

Additional Extensions

Extensions from VS Code Marketplace or Open VSX Registry:
IBM i Testing ***(Required)***
Code for IBM i ***(Required - 2.16.0 minimum)***
RPGLE ***(Required - 0.31.0 minimum)***
COBOL *(Optional - Install if writing COBOL unit tests)*

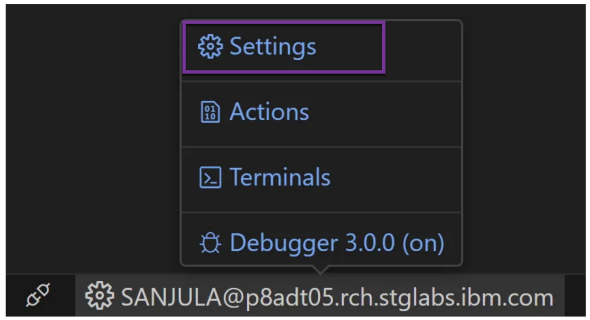
Once you have these, you can install iRPGUnit easily from VS Code
(Instructions coming up)

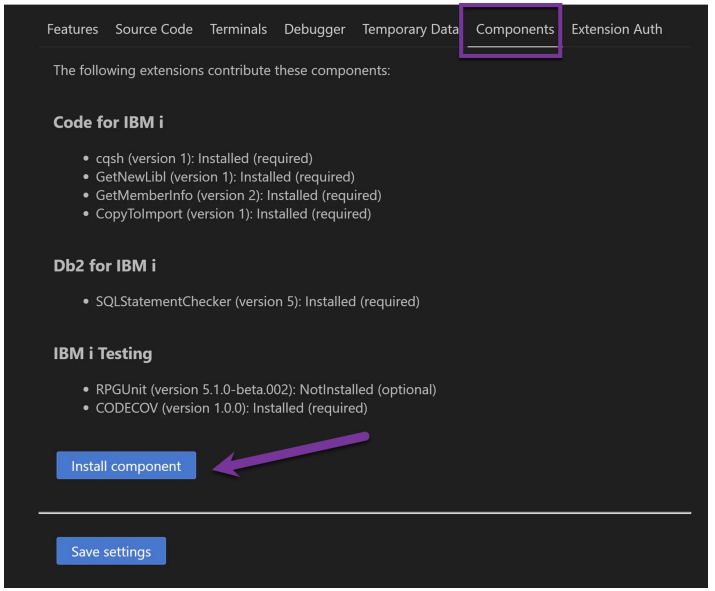
https://codefori.github.io/docs/developing/testing/overview/

SEIDEN GROUP

Install RPGUnit library and CODECOV

Hover over Settings, then click "Components" →





https://codefori.github.io/docs/developing/testing/overview/

SEIDEN GROUP

Extension lets you generate test cases

SEIDEN GROUP

Result

```
1  **free
2
3  ctl-opt nomain ccsidcvt(*excp) ccsid(*char : *jobrun);
4
5  /include qinclude,TESTCASE
6  /include 'qrpgleref/empdet.rpgleinc'
7
8  dcl-proc test_getEmployeeDetail export;
9    dcl-pi *n extproc(*dclcase) end-pi;
10
11   dcl-s empno char(6);
12   dcl-ds actual likeDs(employee_detail_t) inz;
13   dcl-ds expected likeDs(employee_detail_t) inz;
14
15   empno = '';
16
17   actual = getEmployeeDetail(empno);
18
19   expected.found = *off;
20   expected.name = '';
21   expected.netincome = 0.0;
22
23   nEqual(expected.found : actual.found : 'found');
24   assert(expected.name = actual.name : 'name');
25   assert(expected.netincome = actual.netincome : 'netincome');
26 end-proc;
```


SEIDEN GROUP

Now edit your test case

qtestsrc > empdet.test.sqlrpgle > ...

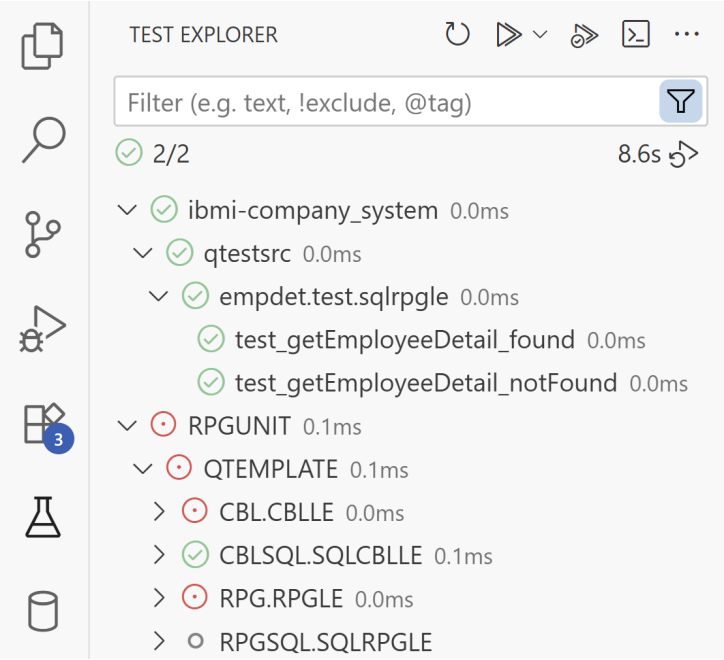
```
1   **free
2
3   ctl-opt nomain ccsidcvt(*excp) ccsid(*char : *jobrun);
3+  ctl-opt nomain ccsidcvt(*excp) ccsid(*char : *jobrun) bnddir('APP');
4
5   /include qinclude,TESTCASE
6   /include 'qrpgleref/empdet.rpgleinc'
7
8   dcl-proc test_getEmployeeDetail export;
9     dcl-pi *n extproc(*dclcase) end-pi;
10
11    dcl-s empno char(6);
12    dcl-ds actual likeDs(employee_detail_t) inz;
13    dcl-ds expected likeDs(employee_detail_t) inz;
14
15    empno = '';
15+   empno = '000010';
16
17    actual = getEmployeeDetail(empno);
18
19    expected.found = *off;
20    expected.name = '';
21    expected.netincome = 0.0;
19+   expected.found = *on;
20+   expected.name = 'CHRISTINE I HAAS';
21+   expected.netincome = 57970;
22
23    nEqual(expected.found : actual.found : 'found');
24    assert(expected.name = actual.name : 'name');
25    assert(expected.netincome = actual.netincome : 'netincome');
26  end-proc;
```


SEIDEN GROUP

Generate more test cases (showing preview mode)

SEIDEN GROUP

See tests in Text Explorer



SEIDEN GROUP

See tests in code editor

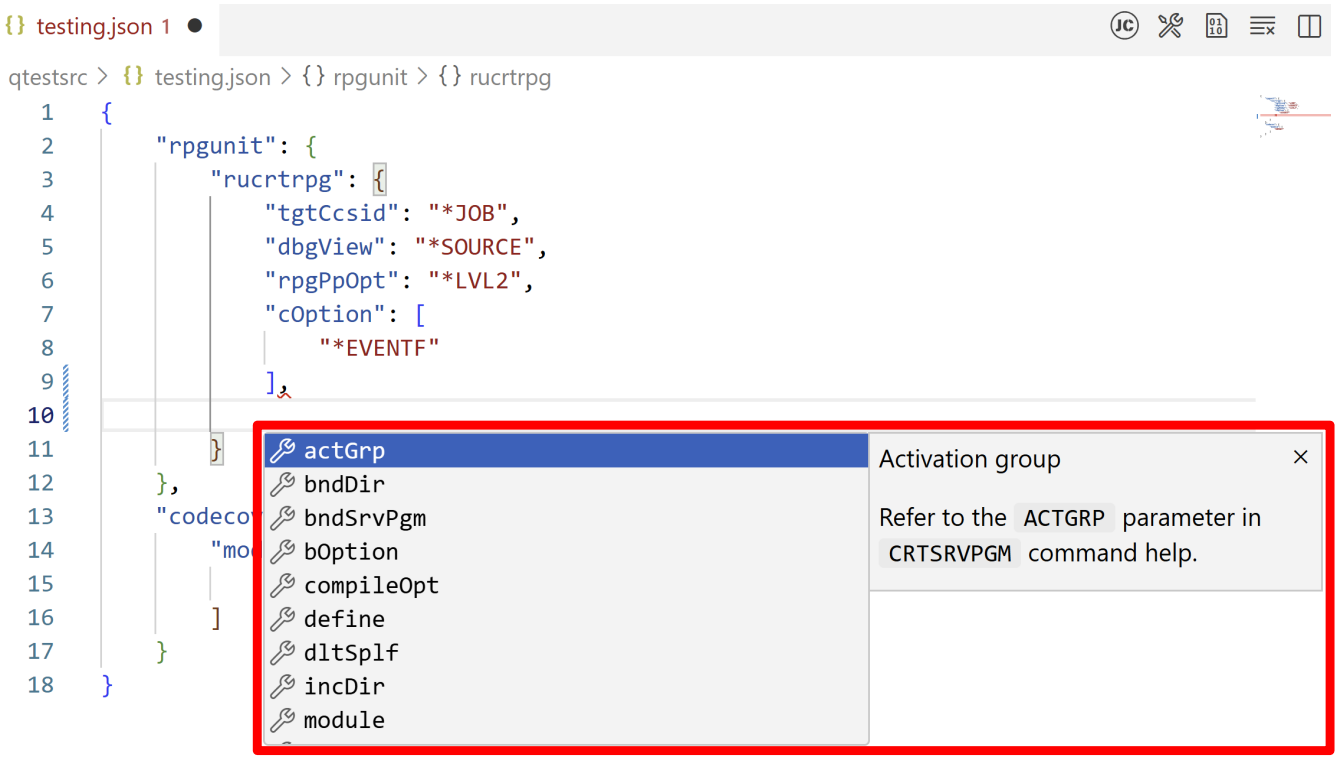
empdet.test.sqlrpgle M

qtestsrc > empdet.test.sqlrpgle > test_getEmployeeDetail_notFound

```
61  dcl-proc test_getEmployeeDetail_notFound export;
62    dcl-pi *n extproc(*dclcase) end-pi;
63
64    dcl-s empno char(6);
65    dcl-ds actual likeDs(employee_detail_t) inz;
66    dcl-ds expected likeDs(employee_detail_t) inz;
67
68    empno = '111111';
69    actual = getEmployeeDetail(empno);
70
71    expected.found = *on;
72
73    nEqual(expected.found : actual.found : 'found');
74  end-proc;
75
76  dcl-proc test_getDeptDetail_found export;
77    dcl-pi *n extproc(*dclcase) end-pi;
```

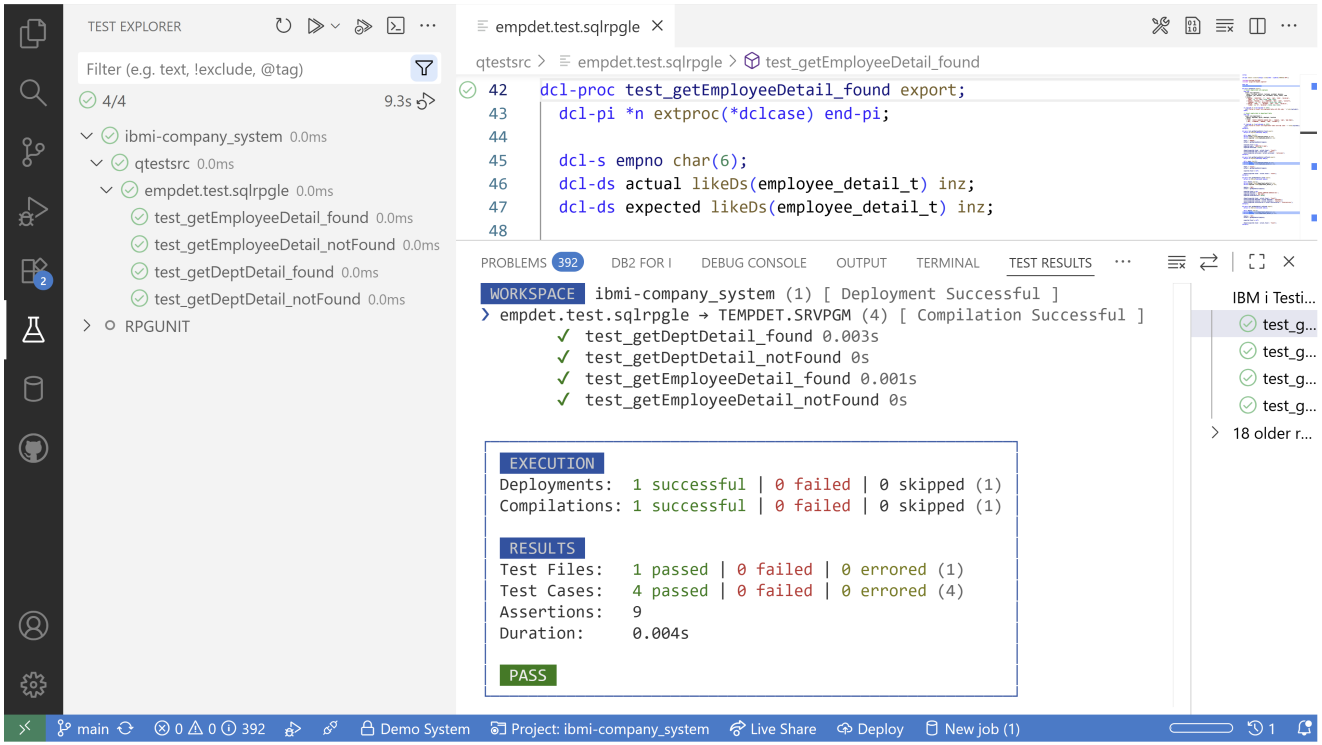

SEIDEN GROUP

Configure tests



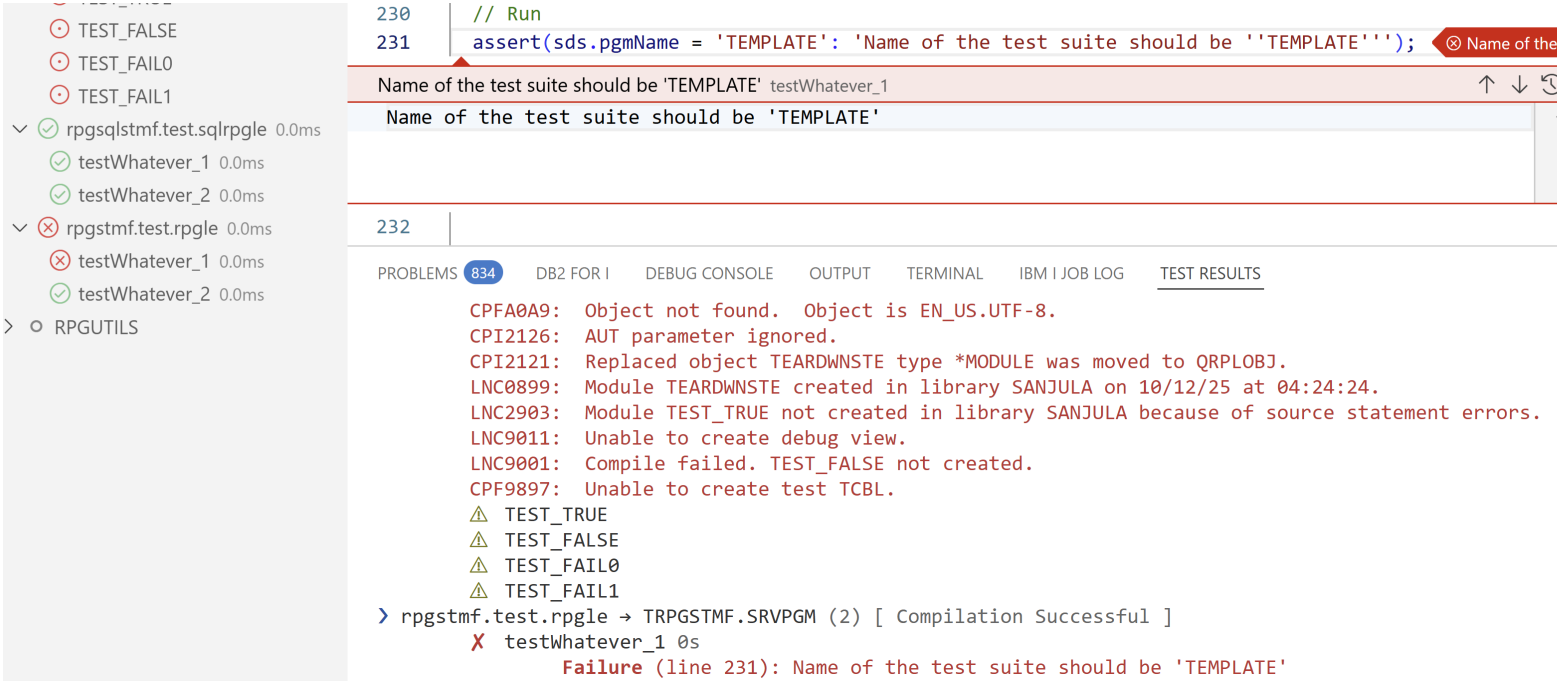
SEIDEN GROUP

Run tests and view results



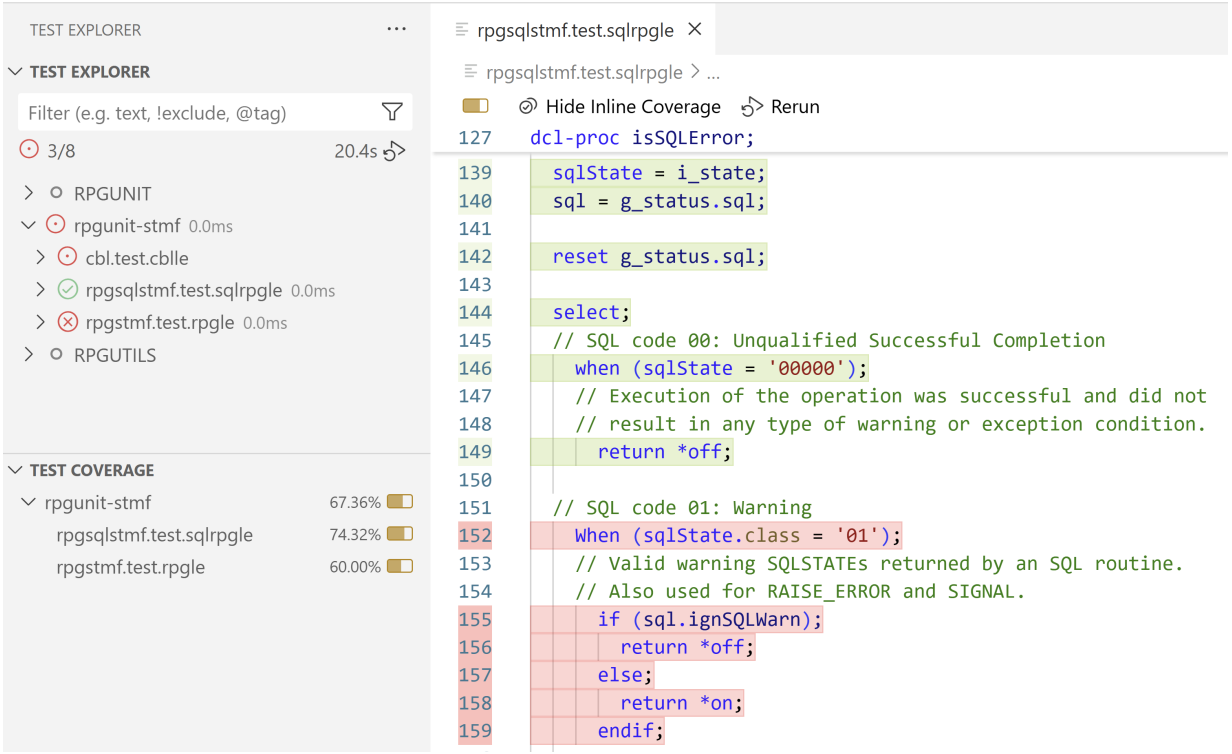
SEIDEN GROUP

View failures and errors

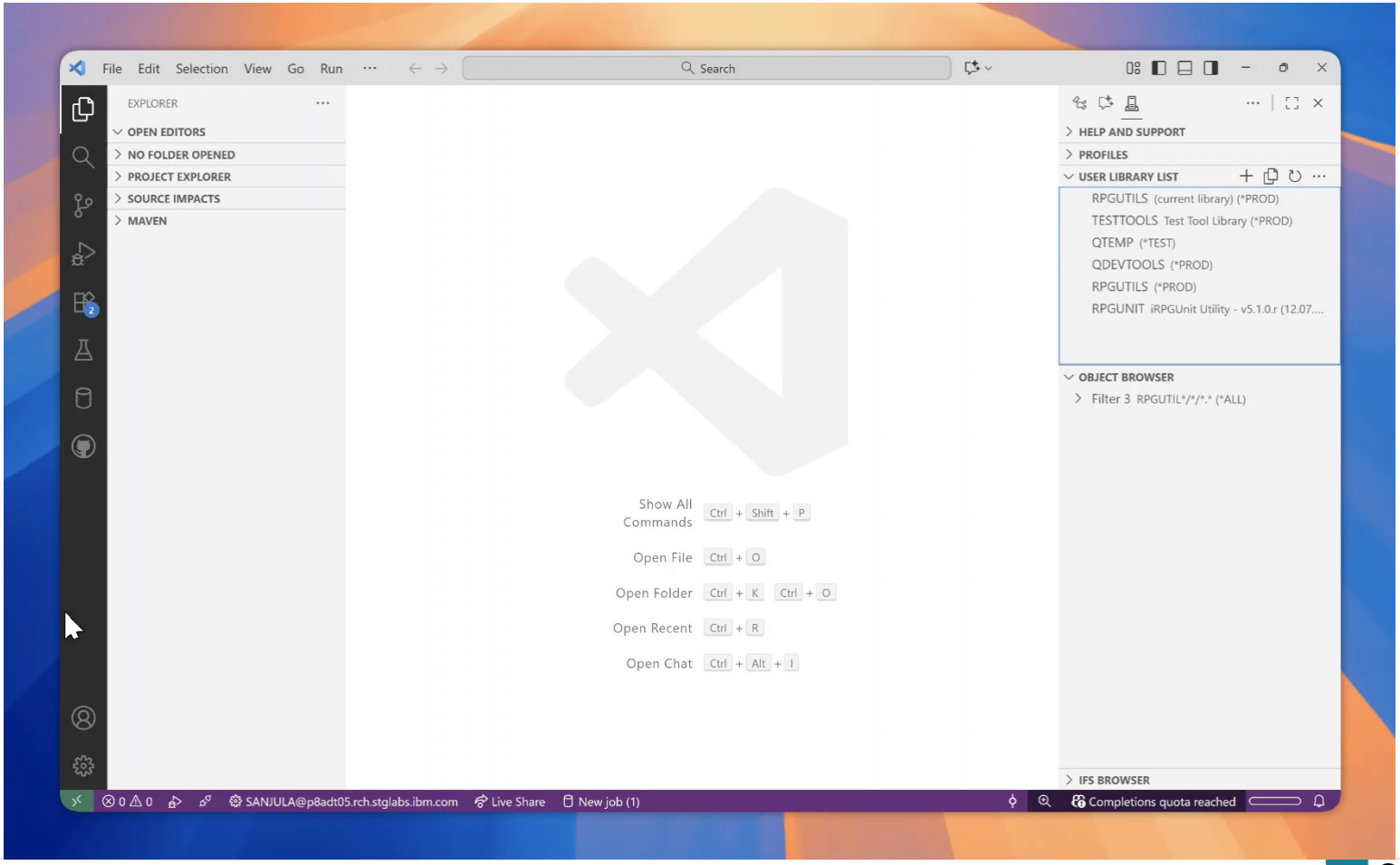


SEIDEN GROUP

Run tests with code coverage

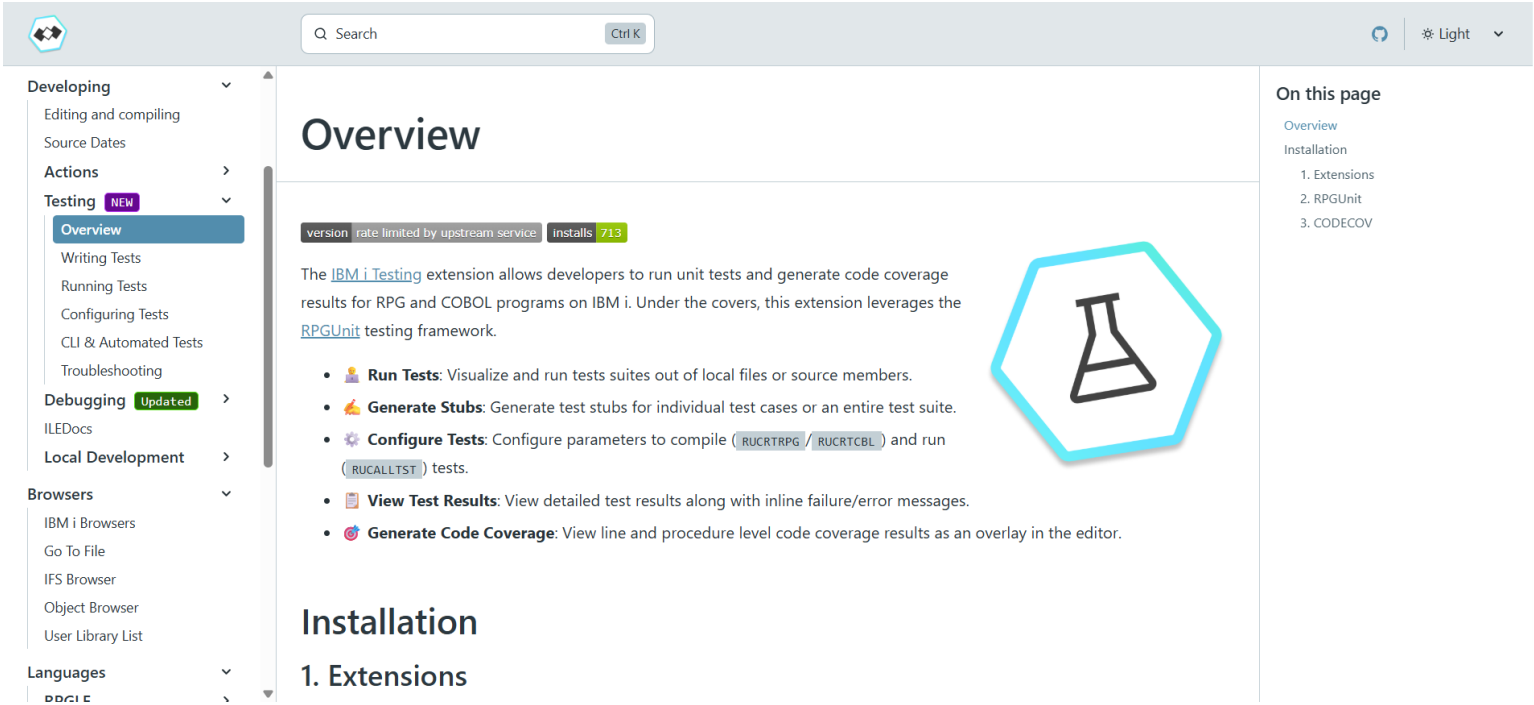


SEIDEN GROUP



SEIDEN GROUP

Documentation for IBM i Testing Extension



https://codefori.github.io/docs/developing/testing/overview/

SEIDEN GROUP

IBM i Testing CLI

SEIDEN GROUP

Installable via NPM (Node package manager)

npm Search packages Search Sign Up Sign In

@ibm/itest TS

1.2.3 • Public • Published a month ago

Readme | Code Beta | 5 Dependencies | 0 Dependents | 2 Versions

IBM i Testing CLI

version v1.2.3 downloads 70/month

The **IBM i Testing CLI** (`itest`) is a companion to the **IBM i Testing** VS Code extension, which allows you to run unit tests and generate code coverage results for RPG and COBOL programs on IBM i. With this CLI, you can run tests in your terminal on your local PC or in PASE on IBM i. This enables you to even script the execution of tests in a CI/CD pipeline.

Install

```
> npm i @ibm/itest
```

Repository

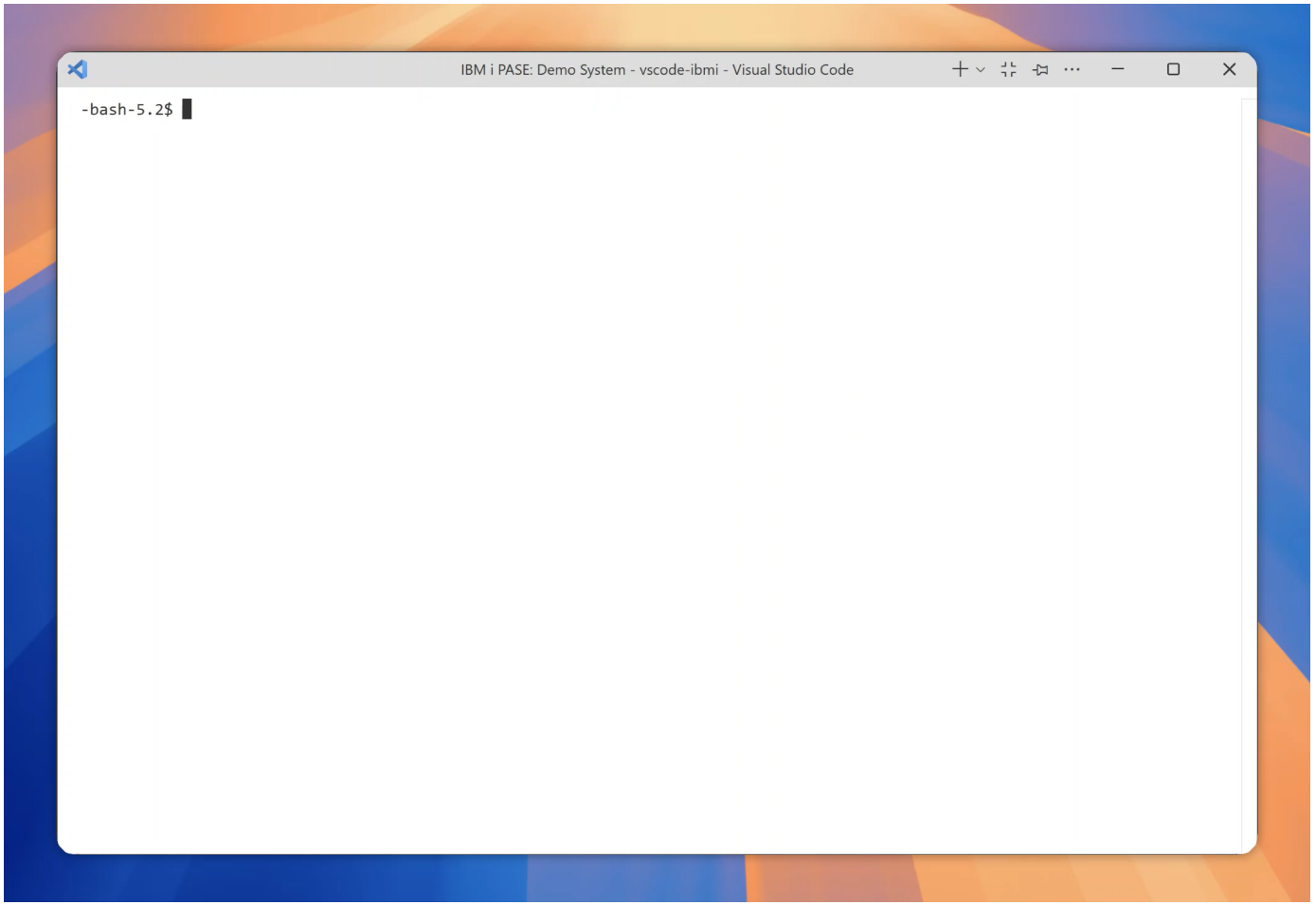
github.com/IBM/vscode-ibmi-testing/b...

Homepage

codefori.github.io/docs/developing/te...

https://www.npmjs.com/package/@ibm/itest

SEIDEN GROUP



SEIDEN GROUP

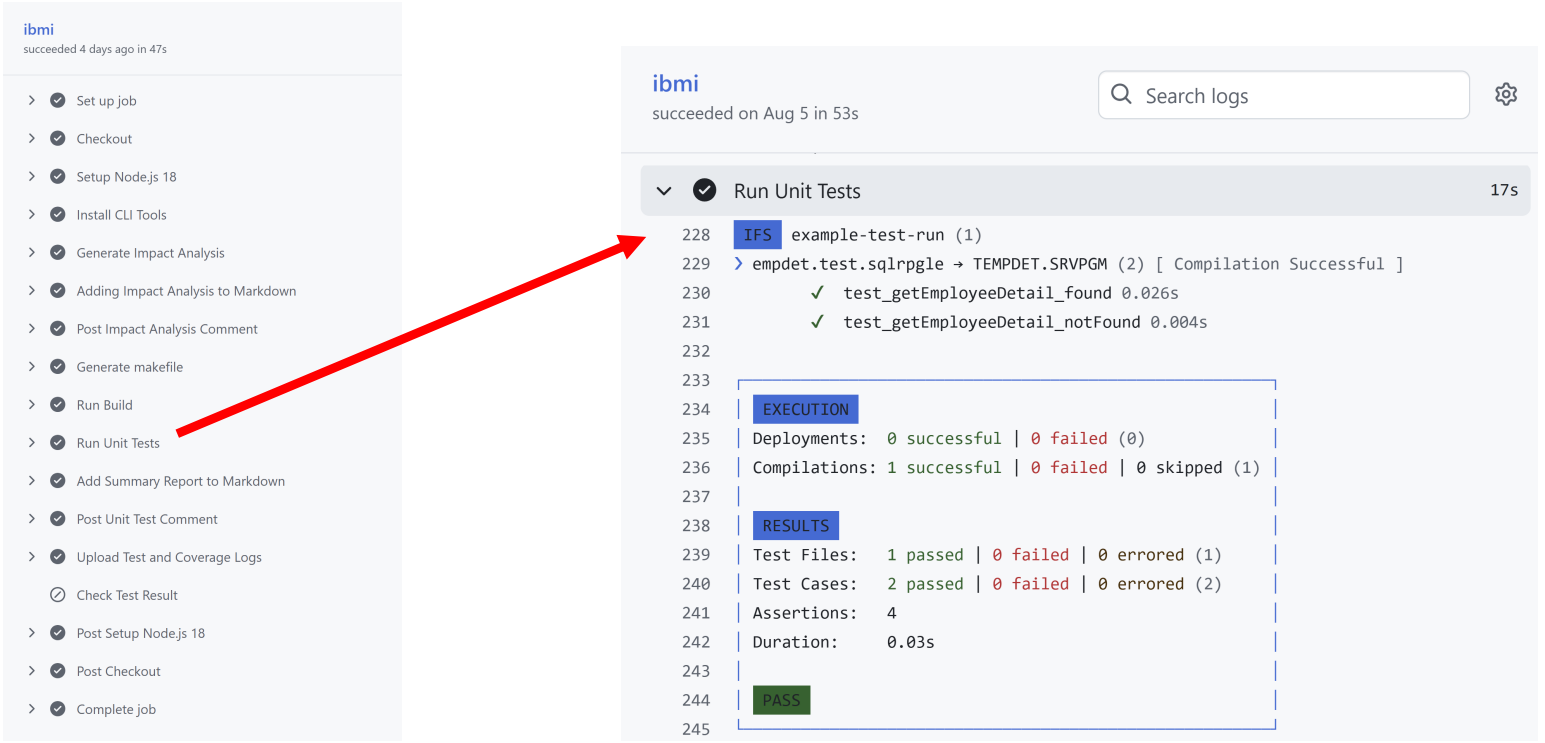
Sample GitHub Action workflow step

```yaml
- name: Run Unit Tests
  id: test
  continue-on-error: true
  run: |
    itest \
      --ld . \
      --id /home/$IBMI_USER/builds/ics_${GITHUB_HEAD_REF} \
      --ll $(so -bl ${GITHUB_HEAD_REF}) CMPSYS RPGUNIT QDEVTOOLS \
      --cl $(so -bl ${GITHUB_HEAD_REF}) \
      --cc --sr --tr --to --co
  env:
    IBMI_HOST: ${{ secrets.IBMI_HOST }}
    IBMI_USER: ${{ secrets.IBMI_USER }}
    IBMI_PASSWORD: ${{ secrets.IBMI_PASSWORD }}
    IBMI_SSH_PORT: ${{ secrets.IBMI_SSH_PORT }}
```

https://github.com/IBM/ibmi-company_system/blob/main/.github/workflows/pr.yaml

SEIDEN GROUP

Pull request triggers impact analysis, build, and tests



SEIDEN GROUP

Pull request reviews made easy

ibmi
succeeded 4 days ago in 47s

- Set up job
- Checkout
- Setup Node.js 18
- Install CLI Tools
- Generate Impact Analysis
- Adding Impact Analysis to Markdown
- Post Impact Analysis Comment
- Generate makefile
- Run Build
- Run Unit Tests
- Add Summary Report to Markdown
- Post Unit Test Comment
- Upload Test and Coverage Logs
- Check Test Result
- Post Setup Node.js 18
- Post Checkout
- Complete job

github-actions bot commented on Aug 5

Test and Code Coverage Report

test passing coverage 76%

Test Result

Total Tests	Passed	Failed	Errored	Assertions	Duration
1	1	0	0	4	0.03s

Code Coverage

File	Coverage	Uncovered Lines	Covered Lines	Executable Lines
empdet.sqlrpgle	53%	9	10	19
empdet.test.sqlrpgle	93%	2	25	27

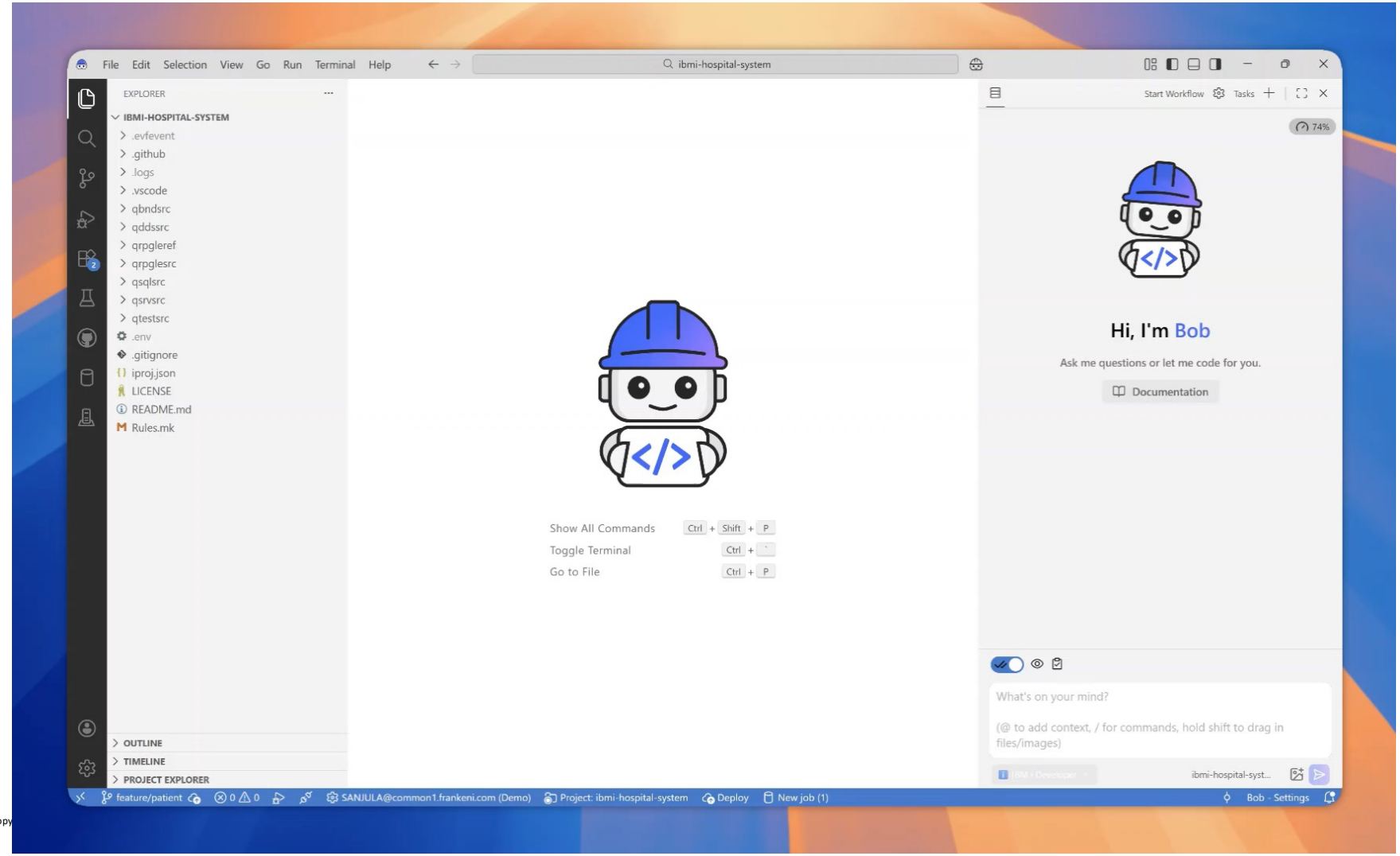
SEIDEN GROUP

Can Bob Help With Unit Testing?

SEIDEN GROUP

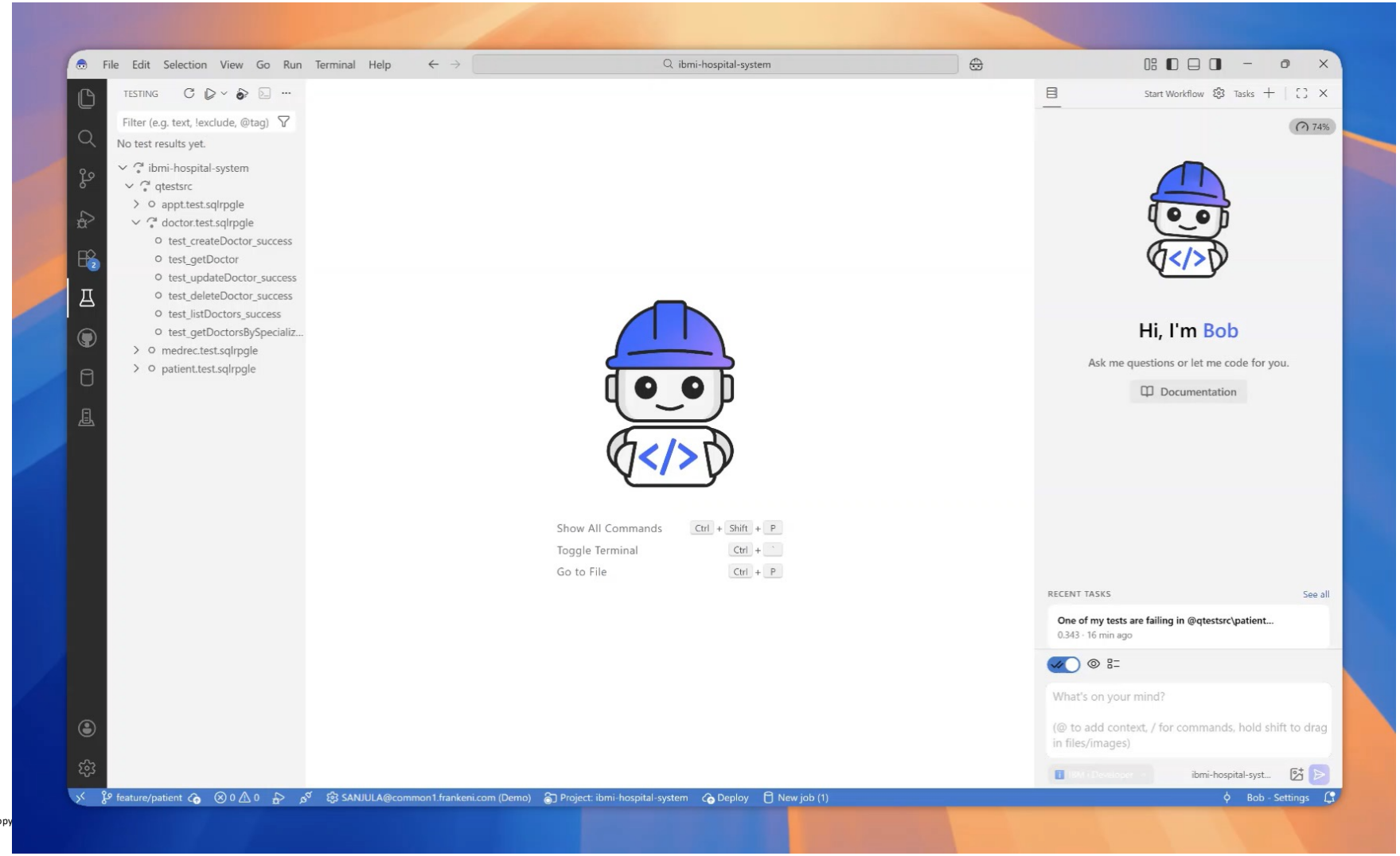
Run RPGUnit tests against some CRUD APIs and resolve error

SEIDEN GROUP


File Edit Selection View Go Run Terminal Help
ibmi-hospital-system
EXPLORER
IBMI-HOSPITAL-SYSTEM
.evfevent
.github
.logs
.vscode
qbndsrc
qddssrc
qrpgleref
qrpglesrc
qsqlsrc
qsrvsrc
qtestsrc
.env
.gitignore
iproj.json
LICENSE
README.md
Rules.mk
OUTLINE
TIMELINE
PROJECT EXPLORER
Show All Commands Ctrl + Shift + P
Toggle Terminal Ctrl + `
Go to File Ctrl + P
Start Workflow
Tasks
74%
Hi, I'm Bob
Ask me questions or let me code for you.
Documentation
What's on your mind?
(@ to add context, / for commands, hold shift to drag in files/images)
ibmi-hospital-syst...
feature/patient
SANJULA@common1.frankeni.com (Demo)
Project: ibmi-hospital-system
Deploy
New job (1)
Bob - Settings

Generate new RPGUnit tests for existing CRUD APIs to improve code coverage

SEIDEN GROUP


File Edit Selection View Go Run Terminal Help
ibmi-hospital-system
TESTING
Filter (e.g. text, !exclude, @tag)
No test results yet.
ibmi-hospital-system
qtestsrc
appt.test.sqlrpgle
doctor.test.sqlrpgle
test_createDoctor_success
test_getDoctor
test_updateDoctor_success
test_deleteDoctor_success
test_listDoctors_success
test_getDoctorsBySpecializ...
medrec.test.sqlrpgle
patient.test.sqlrpgle
Show All Commands Ctrl + Shift + P
Toggle Terminal Ctrl + `
Go to File Ctrl + P
Start Workflow
Tasks
74%
Hi, I'm Bob
Ask me questions or let me code for you.
Documentation
RECENT TASKS
See all
One of my tests are failing in @qtestsrc\patient...
0.343 · 16 min ago
What's on your mind?
(@ to add context, / for commands, hold shift to drag in files/images)
feature/patient
SANJULA@common1.frankeni.com (Demo)
Project: ibmi-hospital-system
Deploy
New job (1)
Bob - Settings

Possible future enhancements

- Improve test case generation support
  - Right now the extension now only supports generating stubs for RPG exported procedures, but Sanjula has the idea to add stubs for:
    - RPG programs
    - CL programs
    - SQL procedures / functions / table functions
- Add support for importing and viewing arbitrary CCZIP files (code coverage reports) in VS Code (similar feature exists in RDi)
- Generate PDF reports with test results and code coverage (in the IDE and via a pipeline) rather than just text/markdown file reports
- Add common test utilities as snippets that can be easily accessed by developers
  - Attendees can share any common utilities they would like to add to the extension and share with the community by posting on GitHub: https://github.com/IBM/vscode-ibmi-testing

RPGUnit and VS Code Links

IBM i Testing

VS Code Marketplace	https://marketplace.visualstudio.com/items?itemName=IBM.vscode-ibmi-testing
Documentation	https://codefori.github.io/docs/developing/testing/overview/
GitHub Repository	https://github.com/IBM/vscode-ibmi-testing

RPGUnit

Documentation	https://irpgunit.sourceforge.io/help/
GitHub Repository	https://github.com/tools-400/irpgunit

Sample Projects

Company System Demo	https://github.com/IBM/ibmi-company_system
RPGUTILS	https://github.com/IBM/vscode-ibmi-testing/tree/main/examples/rpg-utils

VS Code

Documentation	https://code.visualstudio.com/docs/editor/testing

SEIDEN GROUP

Any Questions?

SEIDEN GROUP

IBM i
continuous innovation
continuous integration

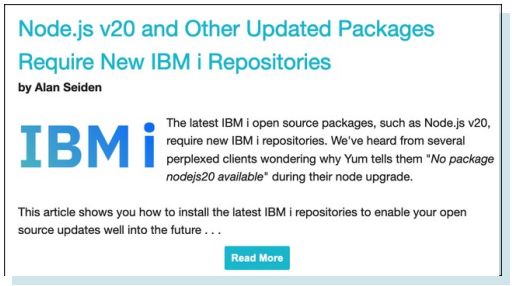
STAY CONNECTED

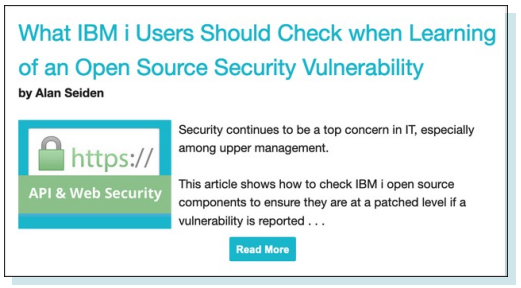
Free Resources from Seiden Group

Code for i Fridays
seidengroup.com/vs-code-for-i-fridays
- next session: 19 June at 20:30 CEST (recorded)

IBM i Strategy & Tips — monthly
seidengroup.com/tips

PHP Free Upgrade Assessment
seidengroup.com/free-php-upgrade-assessment





SEIDEN GROUP

STAY CONNECTED

Contact / Stay in Touch

Alan Seiden
Seiden Group
Ho-Ho-Kus, NJ
alan@seidengroup.com
201-447-2437

Monthly Tips:

seidengroup.com/tips

Open Source Support and Seiden PHP+ Server

https://www.seidengroup.com/seiden-php-for-ibm-i/

SEIDEN GROUP