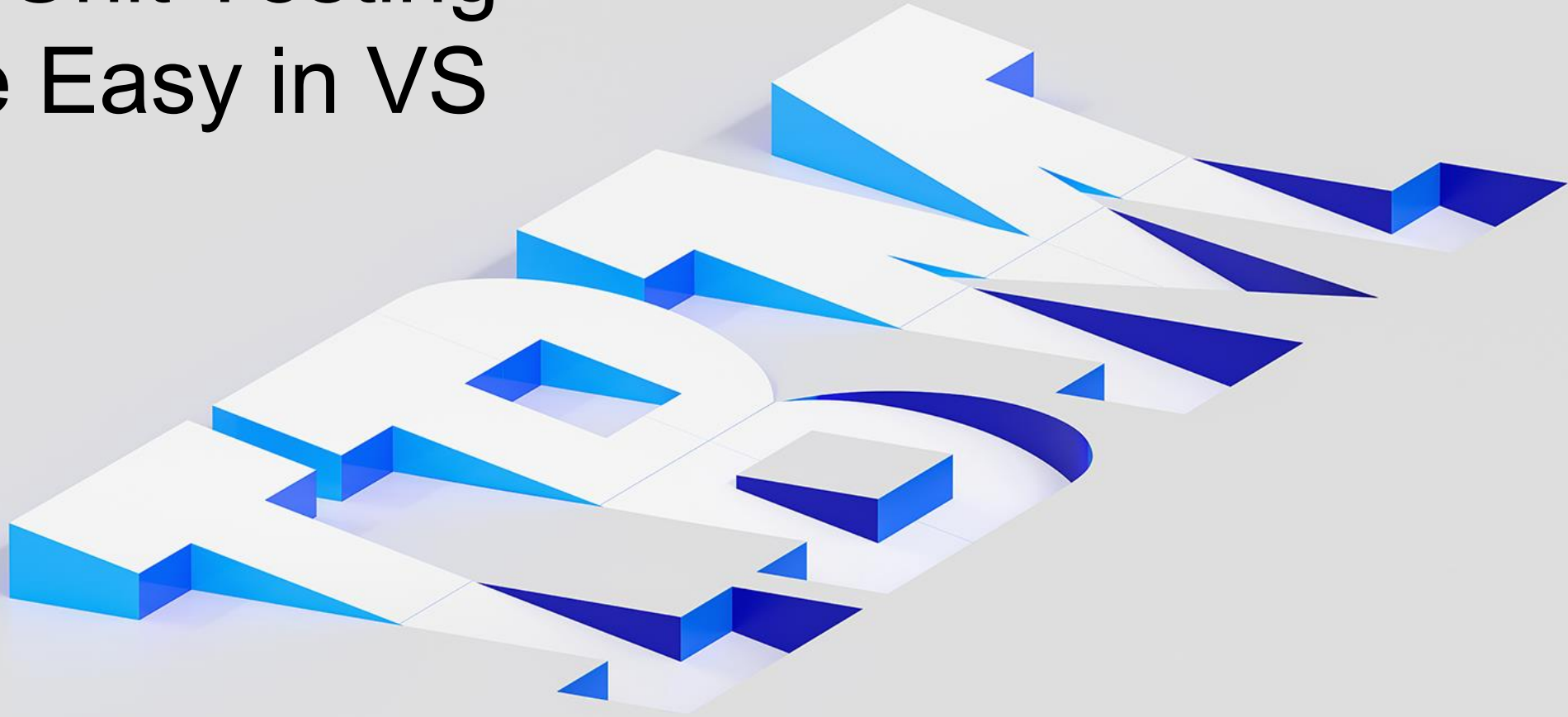


RPG Unit Testing Made Easy in VS Code

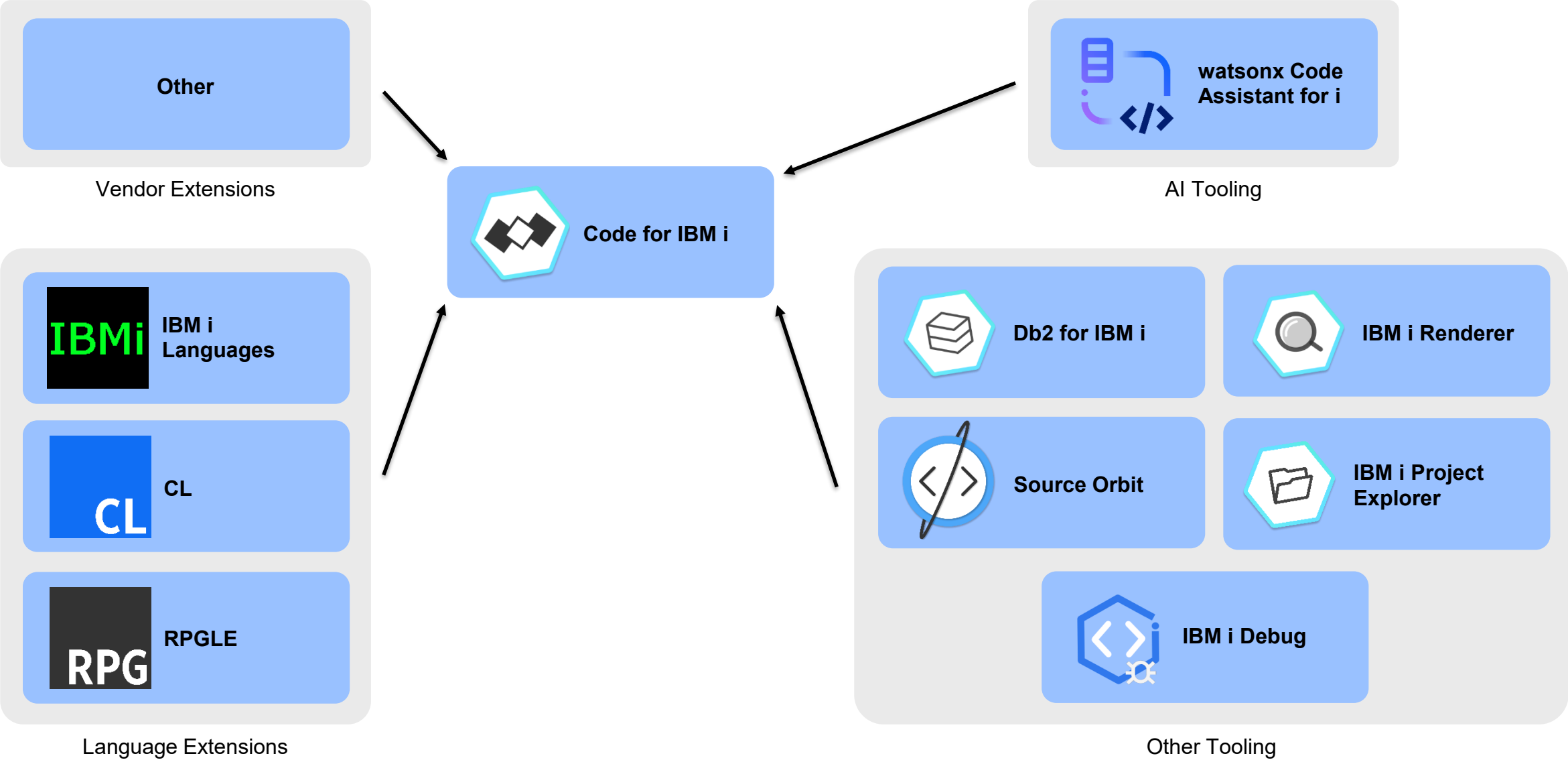


Agenda

- Why Build Tooling for Unit Testing?
- Basics of RPGUnit
- IBM i Testing Extension Demo
 - How to write tests and generate test stubs?
 - How to run tests with code coverage?
- IBM i Testing CLI Demo
 - How to run tests outside VS Code?
 - How to run tests in a CI/CD pipeline?

Why Build Tooling for Unit Testing?

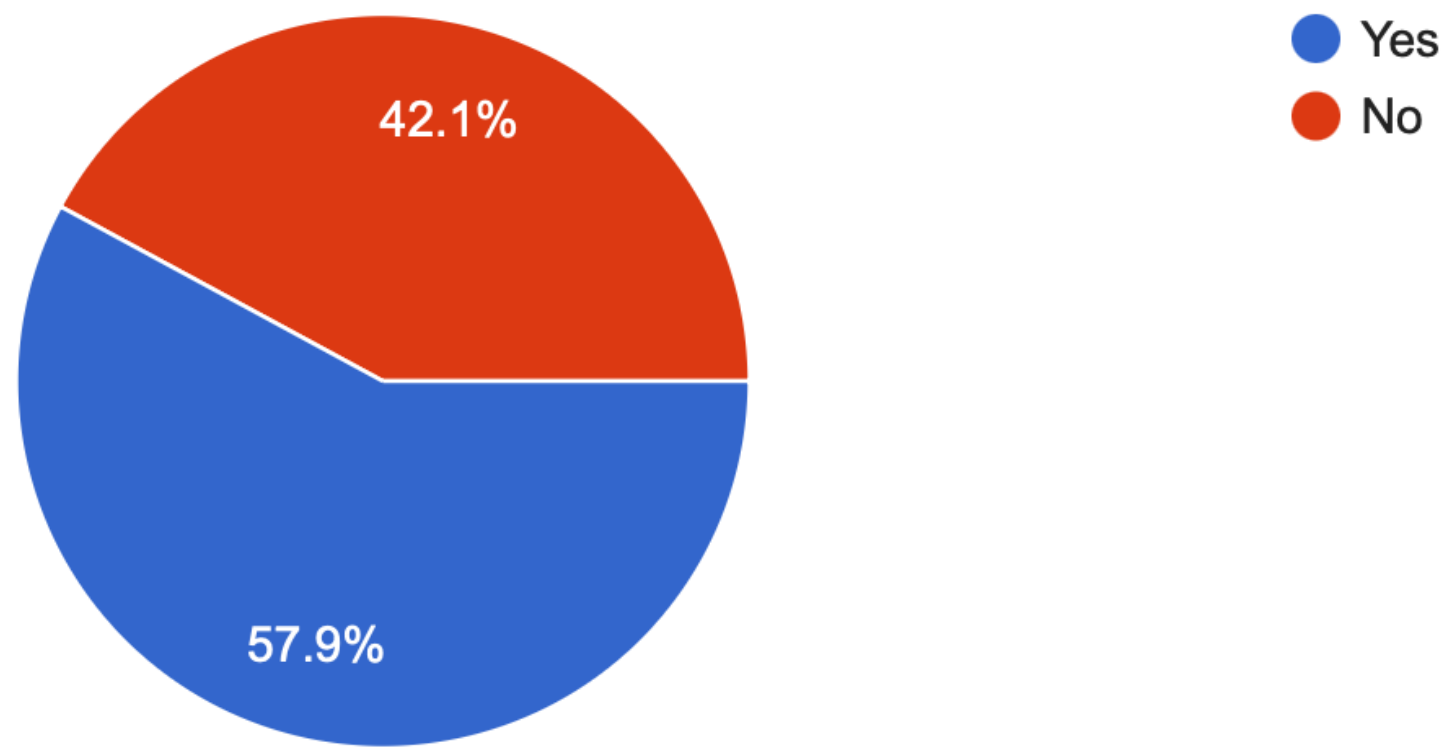
What is missing in the VS Code stack?



We did a survey...

Have you done unit testing with IBM i before?

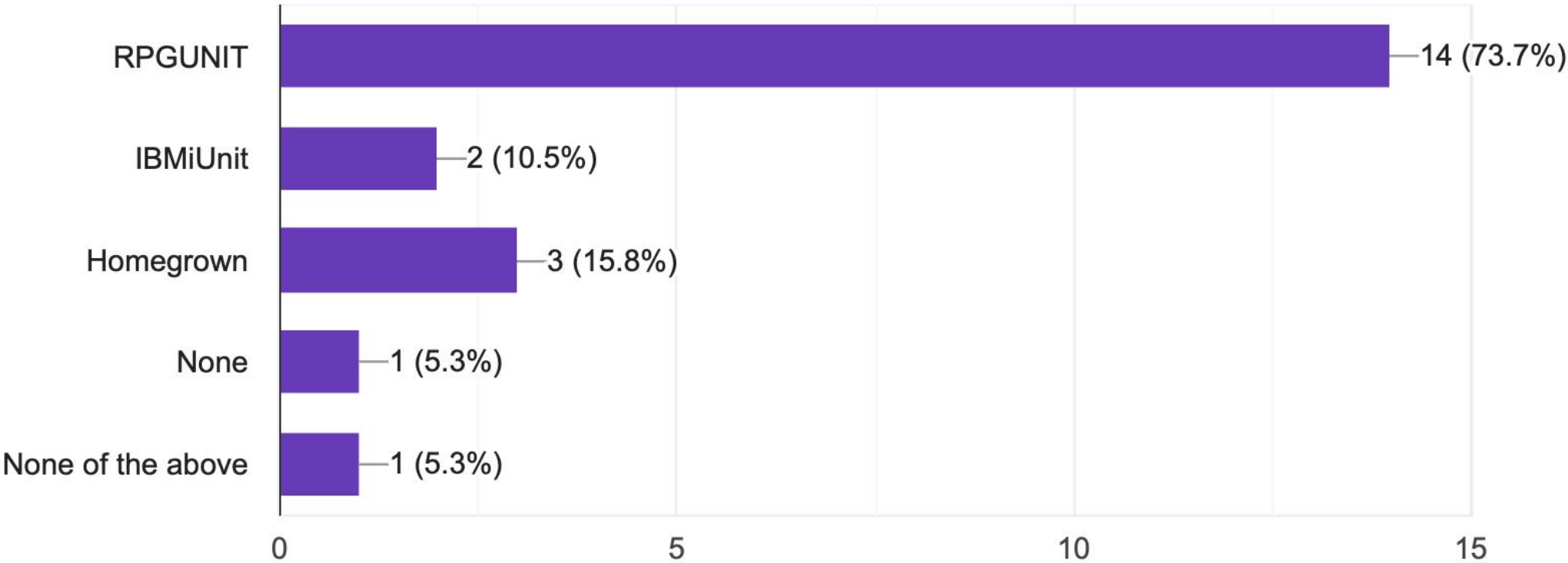
19 responses



We did a survey...

What framework did you use for IBM i testing?

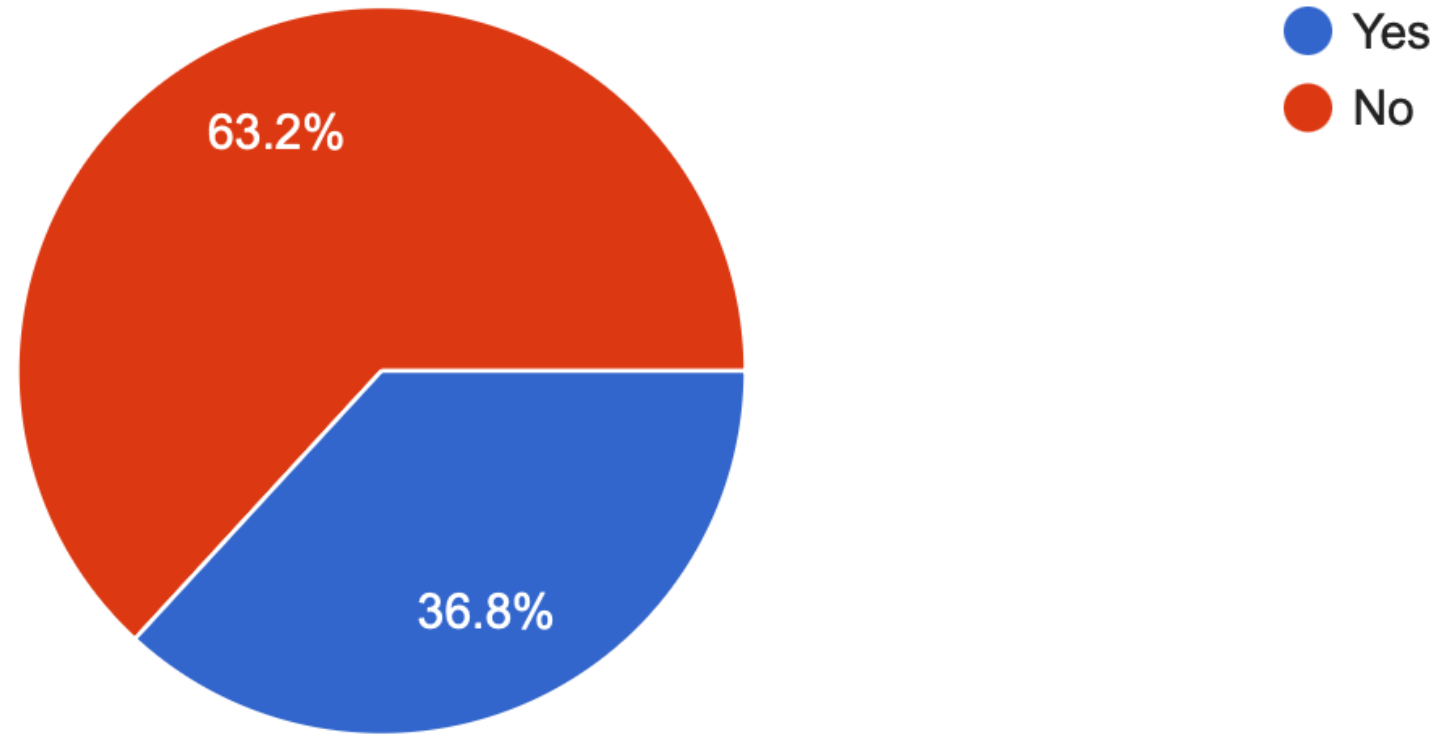
19 responses



We did a survey...

Have you used IBM i Code Coverage? (CODECOV)

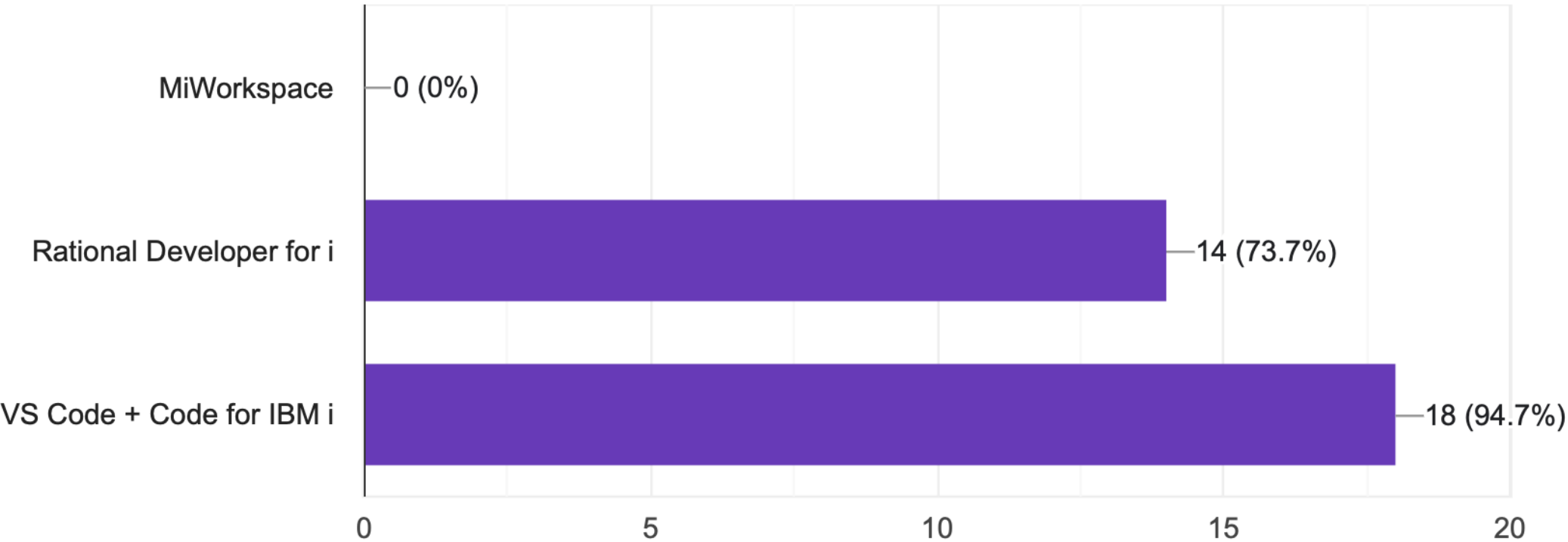
19 responses



We did a survey...

Which IDEs for IBM i have you used?

19 responses





Have you done unit testing with IBM i before?



What framework did you use for IBM i testing?



Have you used CODECOV?

Basics of RPGUnit

Naming requirements

A test case ...

- is an **exported procedure**
- starts with the name **test**
 - test_getEmployeeDetail
 - test_getDeptDetail
 - testIsPalindrome
 - testFactorial

A test suite ...

- has no naming requirements

but the extension requires that ...

- local files are suffixed with **.test.rpgle**, **.test.sqlrpgle**, **.test.cbll**, or **.test.sqlcbll**


and recommends that ...


- source members are prefixed or suffixed with **T**, **_T**, **T_**, etc.

Assertions & utilities

💡 Add the copybook: [/include qinclude,TESTCASE](#)


Example Assertions:

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


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

Example Utilities:

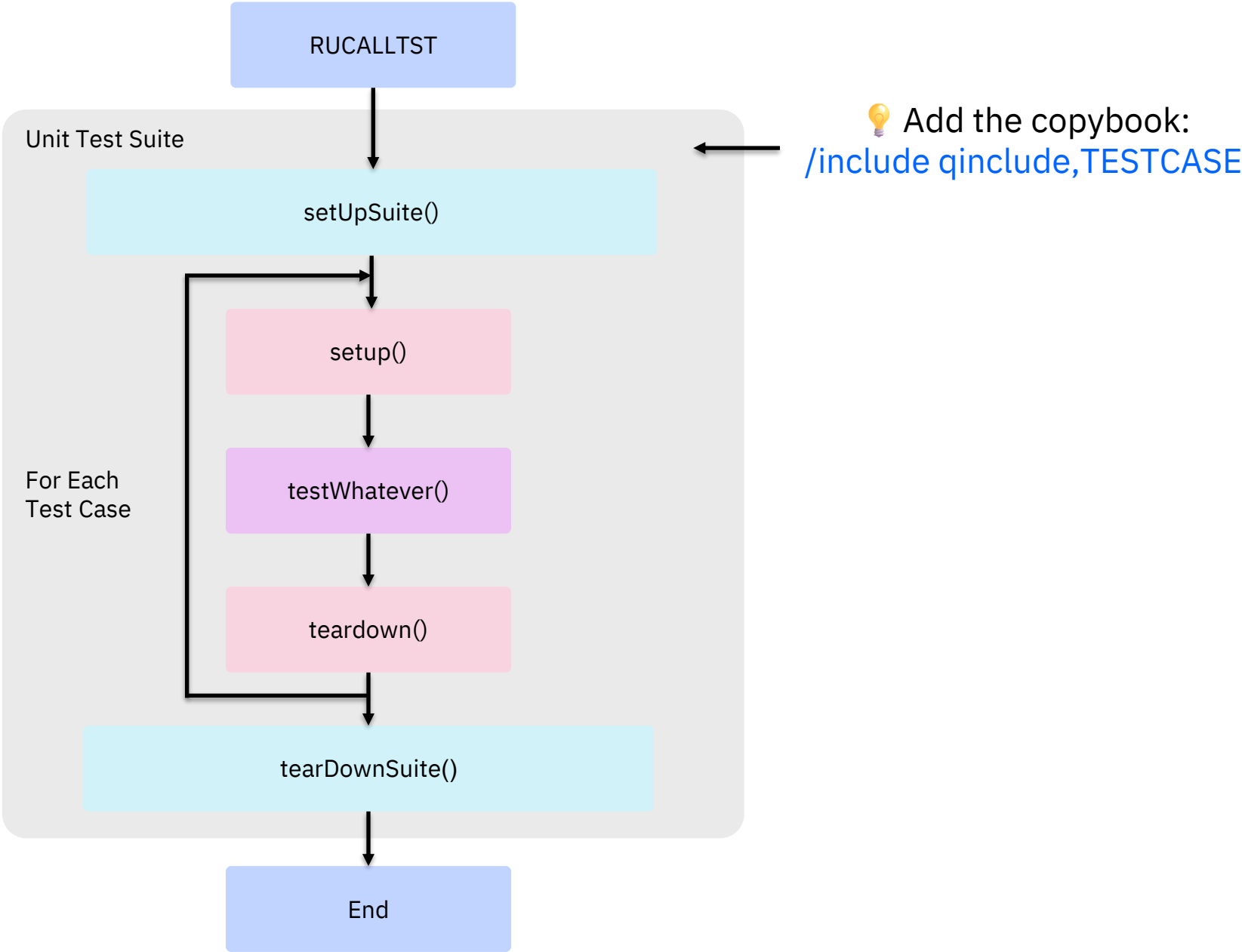
```
monitor;
  a = 10;
  b = 0;
  c = a / b;

//  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 );
```

Set up and tear down



Compile and execution commands

Compile Command:

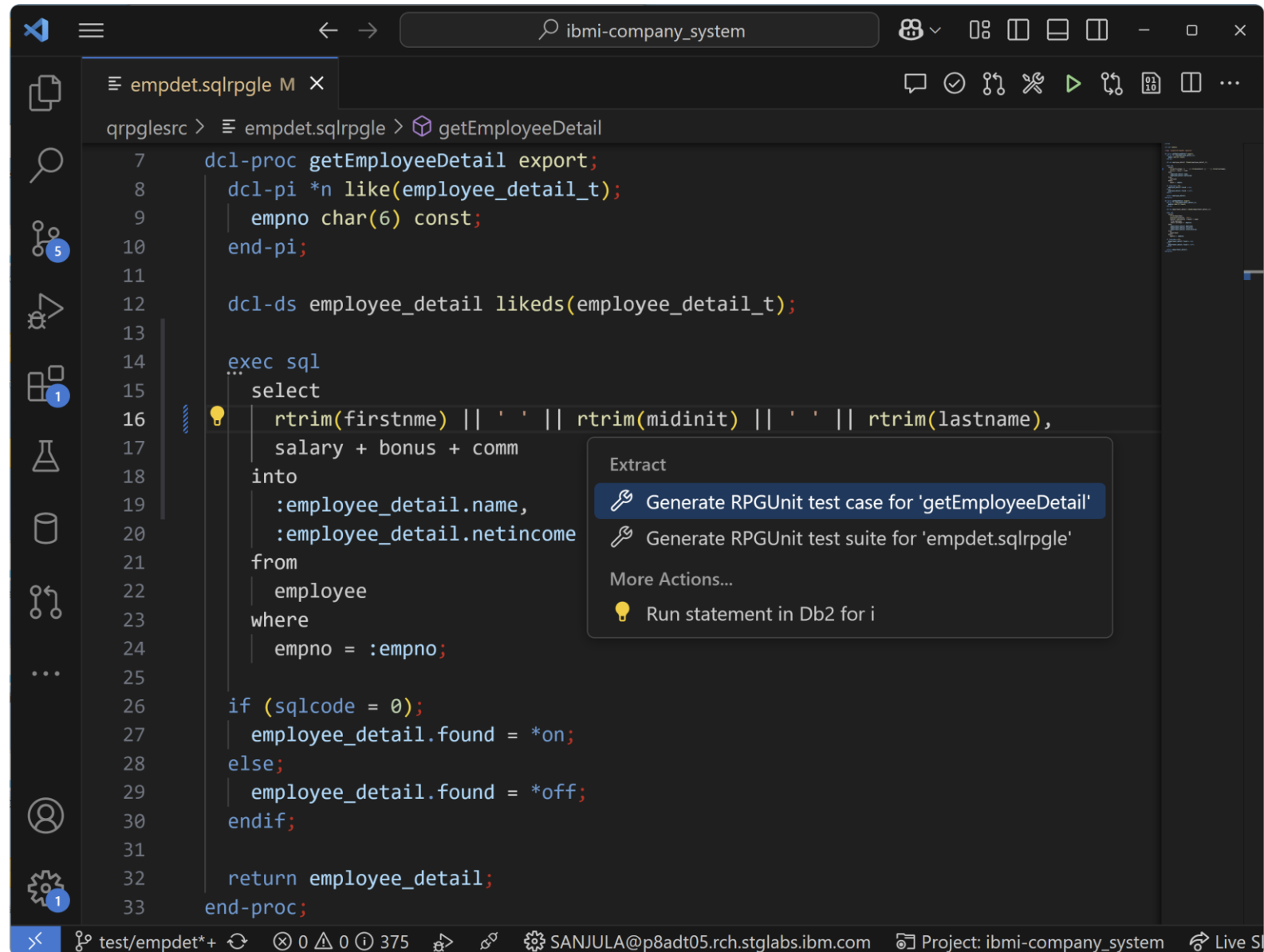
```
RPGUNIT/RUCRTRPG TSTPGM(MYUSER/TEMPDET) SRCSTMF('/home/MYUSER/builds/  
ibmi-company_system/qtestsrc/empdet.test.sqlrpgle') TGTCCSID(*JOB)  
DBGVIEW(*SOURCE) COPTION(*EVENTF) RPGPPOPT(*LVL2) INCDIR(''/home/  
MYUSER/builds/ibmi-company_system'')
```

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 Demo

Generating test stubs



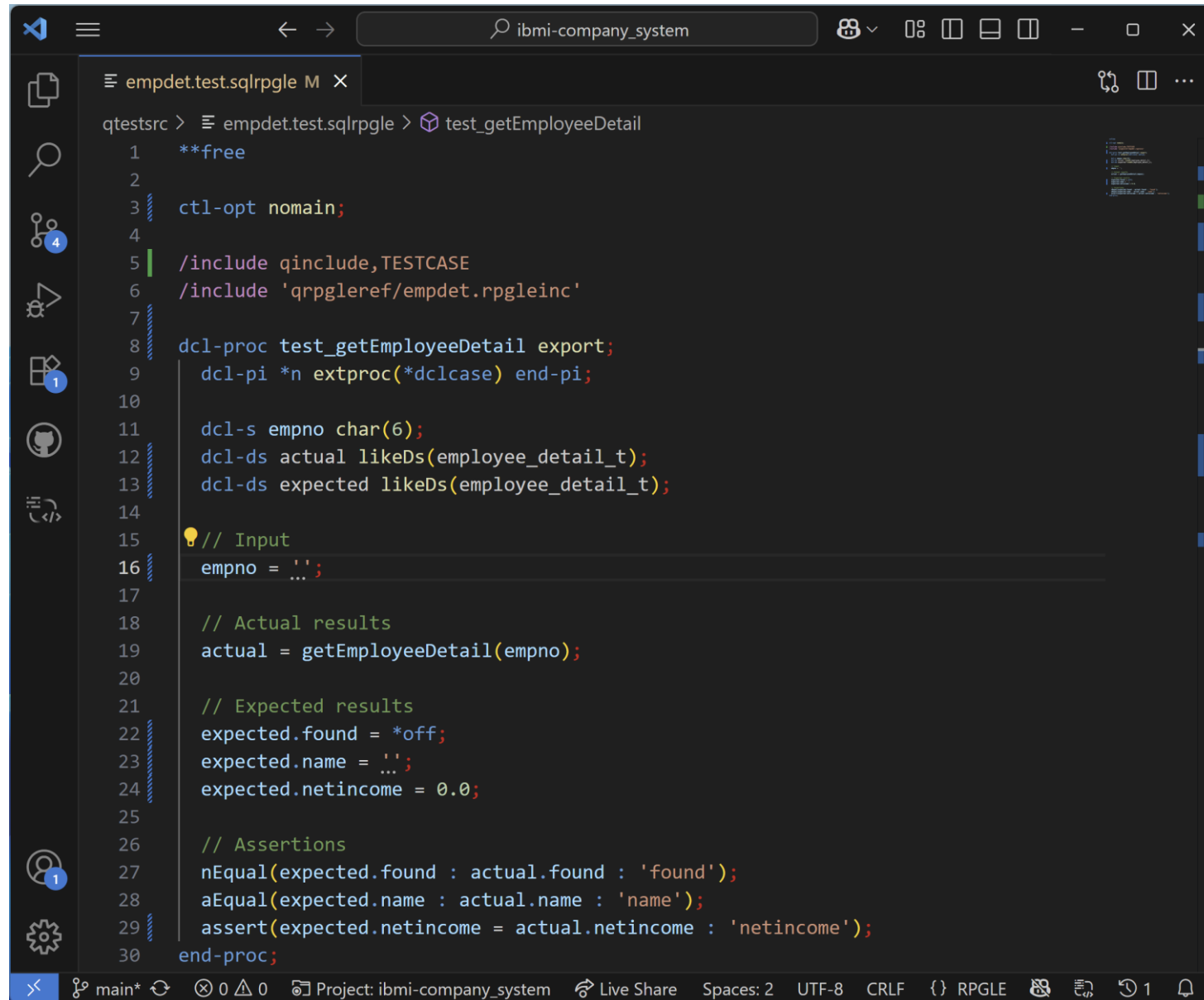
The screenshot shows the IBM Developer IDE with a file named `empdet.sqlrpgle` open. The code is in RPG format and defines a procedure `getEmployeeDetail`. A context menu is visible over the `select` statement, offering options to generate test stubs.

```
7 dcl-proc getEmployeeDetail export;  
8 dcl-pi *n like(employee_detail_t);  
9   empno char(6) const;  
10 end-pi;  
11  
12 dcl-ds employee_detail likeds(employee_detail_t);  
13  
14 exec sql  
15   select  
16     rtrim(firstname) || ' ' || rtrim(midinit) || ' ' || rtrim(lastname),  
17     salary + bonus + comm  
18   into  
19     :employee_detail.name,  
20     :employee_detail.netincome  
21   from  
22     employee  
23   where  
24     empno = :empno;  
25  
26 if (sqlcode = 0);  
27   employee_detail.found = *on;  
28 else;  
29   employee_detail.found = *off;  
30 endif;  
31  
32 return employee_detail;  
33 end-proc;
```

The context menu options are:

- Extract
 - Generate RPGUnit test case for 'getEmployeeDetail'
 - Generate RPGUnit test suite for 'empdet.sqlrpgle'
- More Actions...
 - Run statement in Db2 for i

Generating test stubs

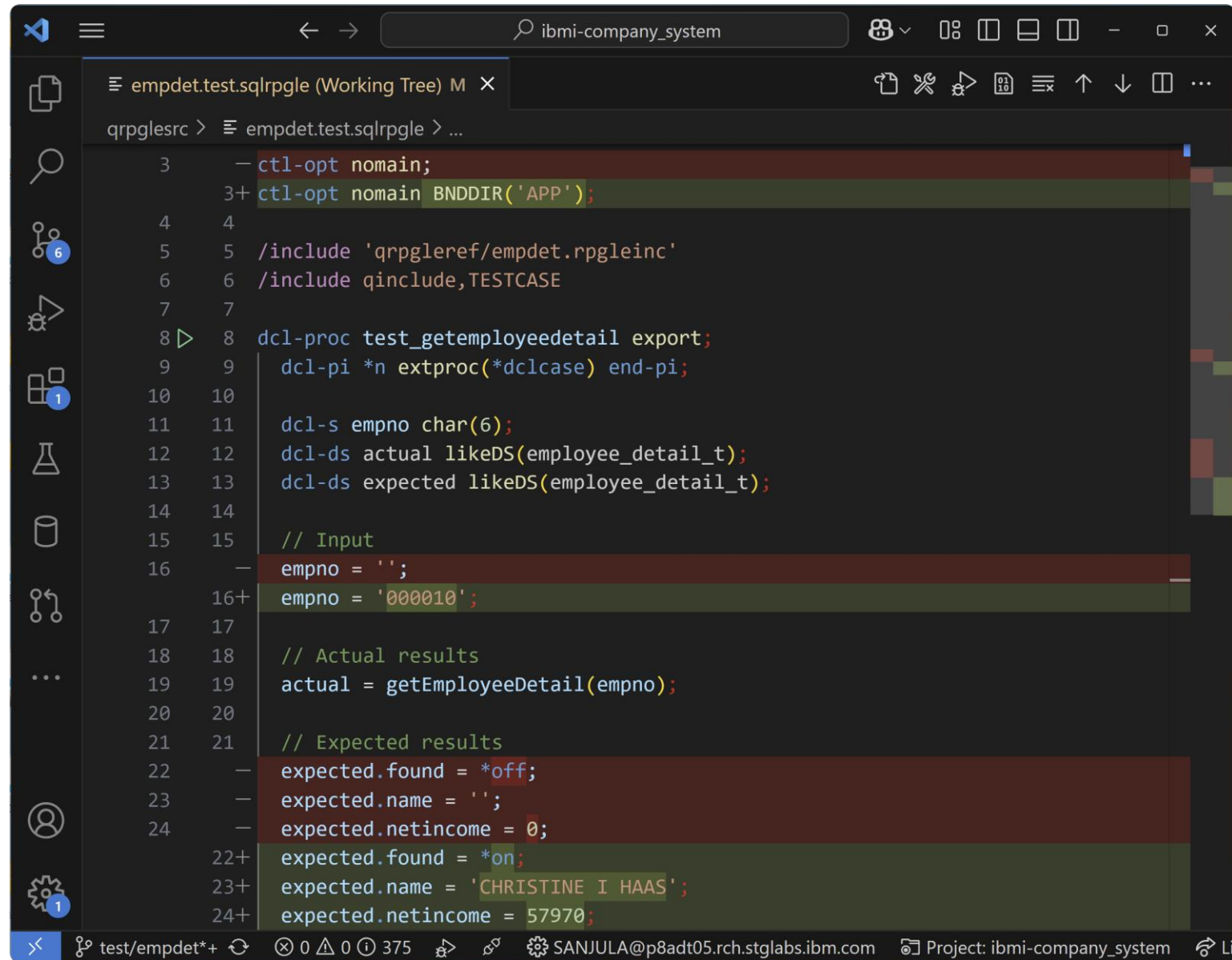


The screenshot shows a code editor window titled 'empdet.test.sqlrpgle M'. The editor displays an RPGLE test stub for the function 'test_getEmployeeDetail'. The code is as follows:

```
1  **free
2
3  ctl-opt nomain;
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);
13      dcl-ds expected likeDs(employee_detail_t);
14
15      // Input
16      empno = '..';
17
18      // Actual results
19      actual = getEmployeeDetail(empno);
20
21      // Expected results
22      expected.found = *off;
23      expected.name = '..';
24      expected.netincome = 0.0;
25
26      // Assertions
27      nEqual(expected.found : actual.found : 'found');
28      aEqual(expected.name : actual.name : 'name');
29      assert(expected.netincome = actual.netincome : 'netincome');
30  end-proc;
```

The IDE interface includes a sidebar with icons for Explorer, Search, Source Control, Run and Debug, Extensions, GitHub, and Testing. The bottom status bar shows 'main*', '0 0 0', 'Project: ibmi-company_system', 'Live Share', 'Spaces: 2', 'UTF-8', 'CRLF', 'RPGLE', and a notification bell.

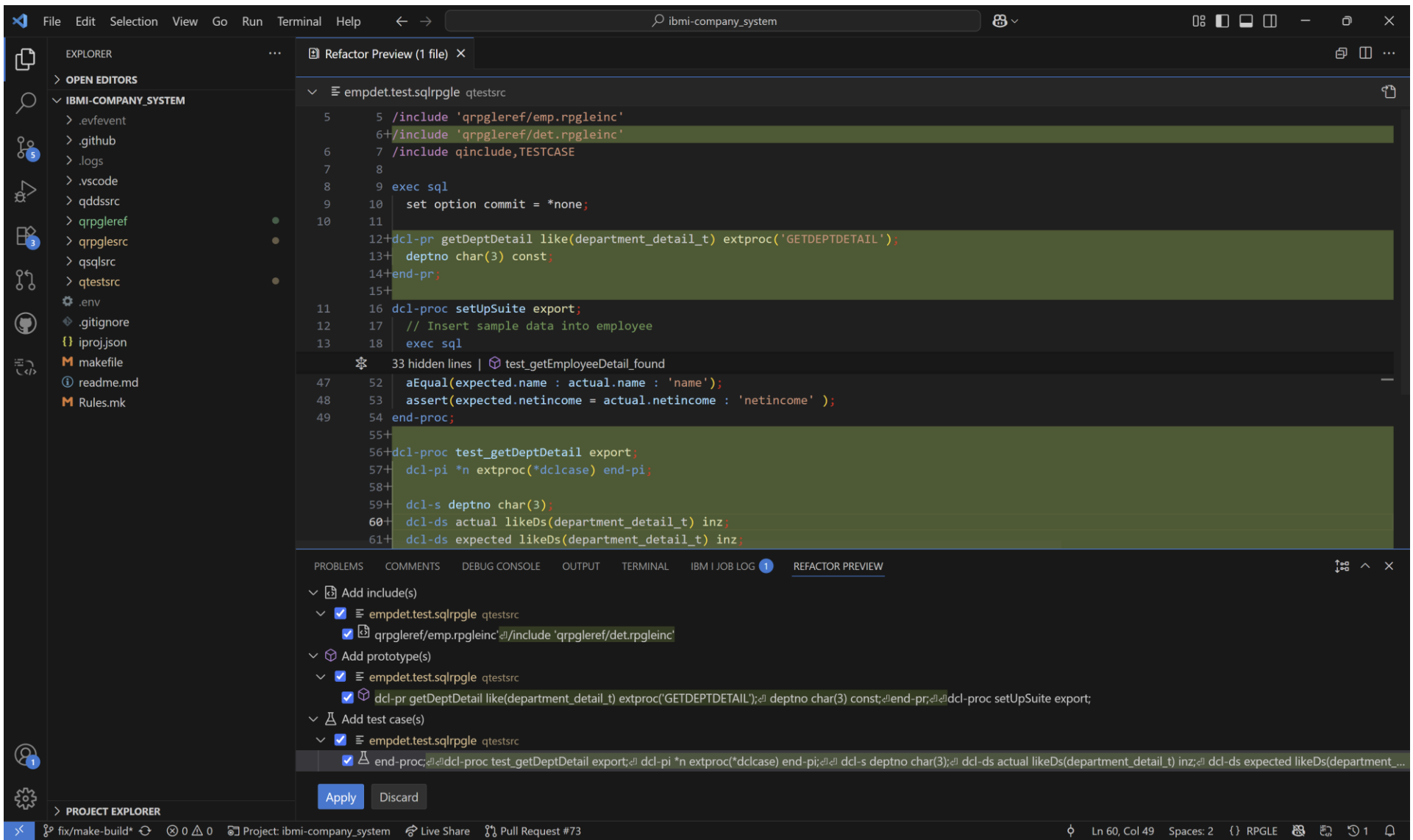
Writing tests



The screenshot shows a Visual Studio Code editor window with a file named `empdet.test.sqlrpgle` open. The editor is displaying COBOL code for a test program. The code includes comments, includes, and data declarations. The `test_getemployeedetail` procedure is defined with a `dc1-pi` block for input and `dc1-s` and `dc1-ds` blocks for data structures. The `actual` and `expected` data structures are defined, and the `actual` structure is populated with data from the `getEmployeeDetail` function. The `expected` structure is also populated with data. The code is written in a dark theme with syntax highlighting.

```
3      - ctl-opt nomain;  
3+    ctl-opt nomain BNDDIR('APP');  
4      4  
5      5 /include 'qrpqleref/empdet.rpgleinc'  
6      6 /include qinclude,TESTCASE  
7      7  
8      8 dc1-proc test_getemployeedetail export;  
9      9   dc1-pi *n extproc(*dc1case) end-pi;  
10     10  
11     11   dc1-s empno char(6);  
12     12   dc1-ds actual likeDS(employee_detail_t);  
13     13   dc1-ds expected likeDS(employee_detail_t);  
14     14  
15     15   // Input  
16     -   empno = '';  
16+    empno = '000010';  
17     17  
18     18   // Actual results  
19     19   actual = getEmployeeDetail(empno);  
20     20  
21     21   // Expected results  
22     -   expected.found = *off;  
23     -   expected.name = '';  
24     -   expected.netincome = 0;  
22+    expected.found = *on;  
23+    expected.name = 'CHRISTINE I HAAS';  
24+    expected.netincome = 57970;
```

Generating more test stubs (preview mode)



Running unit tests

TEST EXPLORER

Filter (e.g. text, !exclude, @tag)

1/1 9.3s

ibmi-company_system 68ms

qrpglesrc 68ms

empdet.test.sqlrpgle 68ms

test_getemployeedetail 68ms

RPGUNIT

empdet.test.sqlrpgle M X

qrpglesrc > empdet.test.sqlrpgle > test_getemployeedetail

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);

13 dcl-ds expected likeDS(employee_detail_t);

14

15 // Input

16 empno = '000010';

PROBLEMS 375 COMMENTS DB2 FOR I DEBUG CONSOLE OUTPUT TERMINAL TEST RESULTS

WORKSPACE ibmi-company_system (1) [Deployment Successful]

> empdet.test.sqlrpgle (1) [Compilation Successful]

✓ test_getemployeedetail 0.068s

EXECUTION

Deployments: 1 successful | 0 failed (1)

Compilations: 1 successful | 0 failed | 0 skipped (1)

RESULTS

Test Files: 1 passed | 0 failed | 0 errored (1)

Test Cases: 1 passed | 0 failed | 0 errored (1)

Assertions: 3

Duration: 0.068s

PASS

IBM i Testing

✓ test_getemployeede...

< test/empdet*+ 0 0 375 SANJULA@p8adt05.rch.stglabs.ibm.com Project: ibmi-company_system Live Share Deploy New job (1) Pul

Viewing failures and errors

TEST EXPLORER

Filter (e.g. text, !exclude, @tag)

6/16

17.3s

examples 35ms

- cbl.test.cbllle
 - TEST_TRUE
 - TEST_FALSE
 - TEST_FAIL0
 - TEST_FAIL1
- rpgsqlstmf.test.sqlrpgle 26ms
 - testWhatever_1 25ms
 - testWhatever_2 1.0ms
- rpgstmf.test.rpgle 3.0ms
 - testWhatever_1 3.0ms
 - testWhatever_2 0.0ms
- sub 6.0ms
- RPGUNIT

rpgstmf.test.rpgle

rpgstmf.test.rpgle > ...

```
222 // =====
223 // RPGUnit test case.
224 // =====
225 dcl-proc testWhatever_1 export;
226   dcl-pi *n extproc(*dclcase) end-pi;
227
228   print('Executing:      * testWhatever_1()');
229
230   // Run
231   assert(sds.pgmName = 'TEMPLATE': 'Name of the test suite should be 'TEMPLATE''');
232
233   // Place your assertions here.
234
235 end-proc;
```

PROBLEMS

DB2 FOR I

DEBUG CONSOLE

OUTPUT

TERMINAL

PORTS

IBM I JOB LOG

TEST RESULTS

WORKSPACE

examples (4) [Deployment Successful]

> rpgstmf.test.rpgle (2) [Compilation Skipped]

X testWhatever_1 0.003s

Failure (line 231): Name of the test suite should be 'TEMPLATE'

✓ testWhatever_2 0s

> rpgsqlstmf.test.sqlrpgle (2) [Compilation Skipped]

✓ testWhatever_1 0.025s

✓ testWhatever_2 0.001s

> cbl.test.cbllle (4) [Compilation Failed]

CPFA0A9: Object not found. Object is EN_US.UTF-8.

: CRTCBMOD MODULE(*PGMID) SRCSTMF('/home/SANJULA/builds/examples/cbl.test.cbllle') OPTION(*EVENTF) DBGVIEW(*SOURCE) TGTRLS(*CURRENT) INCDIR('/home/SANJULA/builds/examples') DEFINE(*NONE) TGTCCSID(37)

CPFA0A9: Object not found. Object is EN_US.UTF-8.

CPFA0A9: Object not found. Object is EN_US.UTF-8.

CPFA0A9: Object not found. Object is EN_US.UTF-8.

CPD4090: Printer device PRT01 not found. Output queue changed to QPRINT i

n library QGPL.

CPFA0A9: Object not found. Object is EN_US.UTF-8.

IBM i Testing

TEST_TRUE cbl.test.cbllle < examples

TEST_FALSE cbl.test.cbllle < examples

TEST_FAIL0 cbl.test.cbllle < examples

TEST_FAIL1 cbl.test.cbllle < examples

TEST_TRUE cbl.test.cbllle < sub < examples

TEST_FALSE cbl.test.cbllle < sub < examples

TEST_FAIL0 cbl.test.cbllle < sub < examples

TEST_FAIL1 cbl.test.cbllle < sub < examples

testWhatever_1 Name of the test suite should b...

testWhatever_1 Name of the test suite should b...

testWhatever_2 rpgstmf.test.rpgle < examples

testWhatever_1 rpgsqlstmf.test.sqlrpgle < examp...

testWhatever_2 rpgsqlstmf.test.sqlrpgle < examp...

testWhatever_2 rpgstmf.test.rpgle < sub < examp...

main*

0 0

SANJULA@p8adt05.rch.stglabs.ibm.com

Live Share

Deploy

New job (1)

Ln 239, Col 45

Spaces: 2

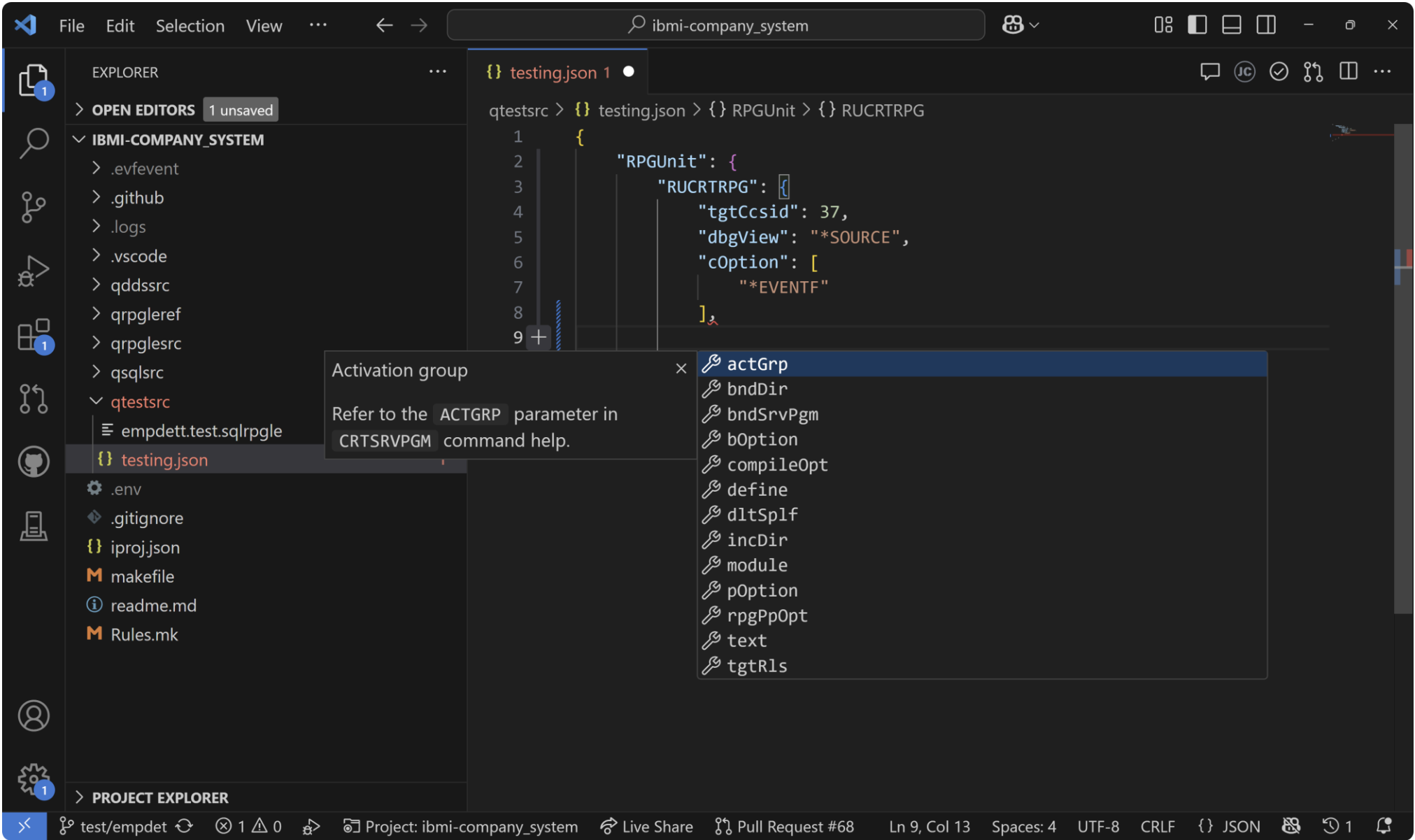
UTF-8

CRLF

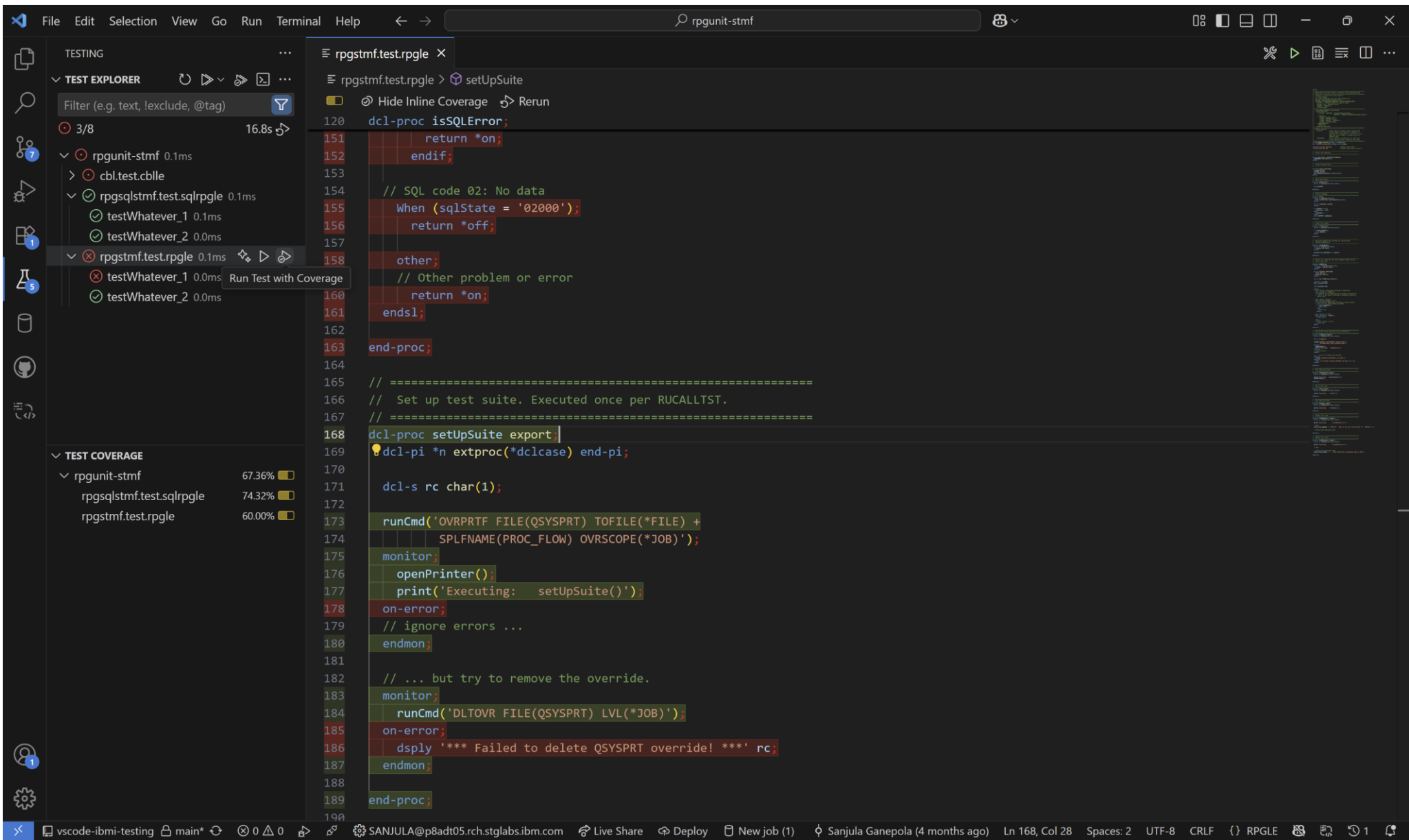
{ } RPGLE

1

Configuring tests



Generating code coverage



IBM i Testing CLI Demo

Running tests from a local terminal or in PASE on IBM i

```
-bash-5.1$ itest --help
Usage: itest [options]
```

The **IBM i Testing CLI** (**itest - v1.1.0**) can be used to run unit tests and generate code coverage results in PASE for RPG and COBOL programs on IBM i. Under the covers, this extension leverages the RPGUnit testing framework.

✦ Documentation: <https://codefori.github.io/docs/developing/testing/overview>

Options:

--v, --version	Display the version number
--ld, --localDirectory [path]	Local directory containing tests (preset: ".")
--id, --ifsDirectory [path]	IFS directory containing containing tests (preset: ".")
--l, --library <library>	Library containing tests.
--sf, --source-files <sourceFiles...>	Source files to search for tests.
--ll, --library-list <libraries...>	Libraries to add to the library list.
--cl, --current-library <library>	The current library to use for the test run.
--cc, --code-coverage [ccLv1]	Run with code coverage (choices: "*LINE", "*PROC", preset: "*LINE")
--ct, --coverage-thresholds <thresholds...>	Set the code coverage thresholds (yellow and green). (default: ["60", "90"])
--sr, --summary-report [path]	Save summary report (preset: "./.itest/summary-report.md")
--tr, --test-result [path]	Save test result logs (preset: "./.itest/test-result.log")
--to, --test-output [path]	Save test output logs (preset: "./.itest/test-output.log")
--co, --command-output [path]	Save command output logs (preset: "./.itest/command-output.log")
--h, --help	Display help for command

Examples:

```
itest --ld . --id /home/USER/builds/ibmi-company_system --ll RPGUNIT QDEVTOOLS --cl MYLIB
itest --id /home/USER/builds/ibmi-company_system --ll RPGUNIT QDEVTOOLS --cl MYLIB
itest --l MYLIB --ll RPGUNIT QDEVTOOLS --cl MYLIB
```

Running tests and code coverage in a pipeline

Summary

Jobs

ibmi

Run details

Usage

Workflow file

ibmi

succeeded 2 minutes ago in 41s

Search logs

Build and Test

221 =====

222 Executing step 9: cmd

223 =====

224

225 /home/***/builds/ics_test/example \$ itest --id . --ll VSB3C8B4D3 RPGUNIT QDEVT00LS --c1 VSB3C8B4D3 --cc --sr --tr --to --co

226 - Setting up environment

227 - Loading tests

228 IFS example (1)

229 > empdet.test.sqlrpgle → TEMPDET.SRVPGM (2) [Compilation Successful]

230 ✓ test_getEmployeeDetail_found 0.01s

231 ✓ test_getEmployeeDetail_notFound 0.004s

232

233

234 EXECUTION

235 Deployments: 0 successful | 0 failed (0)

236 Compilations: 1 successful | 0 failed | 0 skipped (1)

237

238 RESULTS

239 Test Files: 1 passed | 0 failed | 0 errored (1)

240 Test Cases: 2 passed | 0 failed | 0 errored (2)

241 Assertions: 4

242 Duration: 0.014s

243

244 PASS

245

246

Viewing test and code coverage results

Add employee test cases #2

Open

SanjulaGanepola wants to merge 2 commits into `main` from `test/example`

Conversation 1

Commits 2

Checks 1

Files changed 0

+0 -0

SanjulaGanepola commented 1 hour ago

No description provided.

SanjulaGanepola added 2 commits 1 hour ago

`restore tests`

SanjulaGanepola temporarily deployed to COMMON1 1 hour ago

with GitHub Actions

Test and Code Coverage Report

test passing coverage 76%

Test Result

Total Tests	Passed	Failed	Errored	Assertions	Duration
1	1	0	0	4	0.013s

Code Coverage

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

Reviewers

No reviews

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

Notifications

Unsubscribe

You're receiving notifications because you modified the open/close state.

1 participant

Lock conversation

© Copyright IBM Corporation 2024

29

Important 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>

IBM i