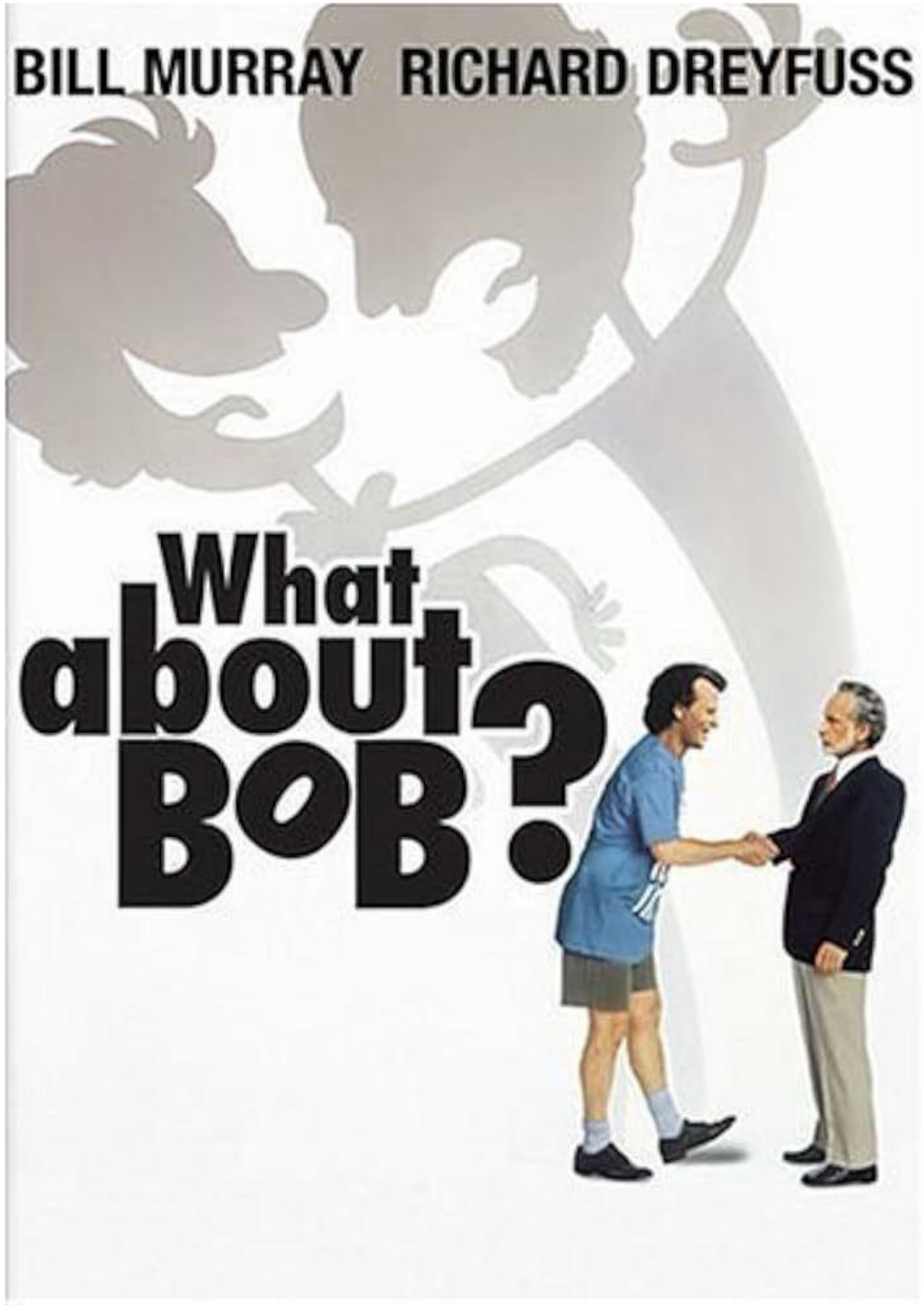
Modern, Buildable Projects
with IBM i Project Explorer and Bob

&

AI-Driven Code Assist for IBM i

Edmund Reinhardt
Product Architect - IBM i Application Development
edmund.reinhardt@ca.ibm.com

© Copyright IBM Corporation 2024



IBM i
continuous innovation
continuous integration

Better

Object

Builder

© Copyright IBM Corporation 2024

Agenda

IBM i
continuous innovation
continuous integration

- Challenges with Building on IBM i
- How do IBM i Projects and Bob overcome this?
- Ins and Outs of IBM i Project Explorer

- New AI-Driven Code Assistant for IBM i
- How you can help?

© Copyright IBM Corporation 2024

IBM i
continuous innovation
continuous integration

Challenges with Building on IBM i

© Copyright IBM Corporation 2024

Building on IBM i is limiting...

IBM i
continuous innovation
continuous integration

1. SRC-PF
   - 10 char names
   - Fixed record length
   - Not accessible to open ecosystem, including Git and Make
   - Source of the same type stored in QxxxSRC to avoid name conflicts (member type does not disambiguate)

2. Libraries
   - Only 2 level hierarchy to organize, with only short 10 char names

3. Source control
   - None (sequence number dates)
   - Home grown
   - Proprietary IBM i systems
     - Cost
     - Smaller market = less investment

4. Build system
   - Individual CRTXXXMOD + CRTPGM
   - CL Scripts
   - A couple of vendors have dependency-based build

© Copyright IBM Corporation 2024

How do IBM i Projects and Bob overcome this?

© Copyright IBM Corporation 2024

Let's use a different (but similar) file system

IBM i
continuous innovation
continuous integration

MYPROJECT
- QRPGLESRC
  - CUSTOMER.RPGLE
  - INVENTORY.RPGLE
  - STARTJOB.RPGLE
- QSQLSRC
  - CUSTOMER.SQL
  - INVENTORY.SQL
- QCLLESRC
  - START.CLLE
- QCMDSRC
  - STARTJOB.CMD

QSYS.LIB Library

No more character name restrictions

Now usable with Git and Make

/my-project
- /.git
- qrpglesrc
  - customer.rpgle
  - inventory.rpgle
  - startjob.rpgle
- qsqlsrc
  - customer.table
  - inventory.sql
- qcllesrc
  - start.clle
- qcmdsrc
  - startjob.cmd

IFS/Local File System

Flexible directory structure

/my-project
- /.git
- customers-file-maint
  - customer.rpgle
  - customer.table
  - programc.rpgle
- inventory-service
  - inventory.rpgle
  - inventory.table
- start-command
  - start.clle
  - startjob.pgm.rpgle
  - startjob.cmd

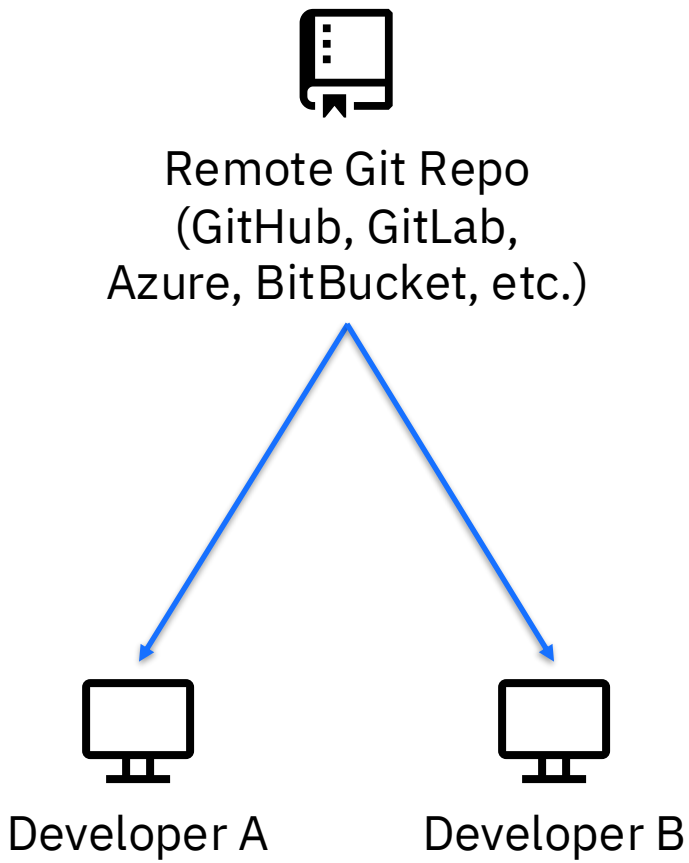
Logical Organization

© Copyright IBM Corporation 2024

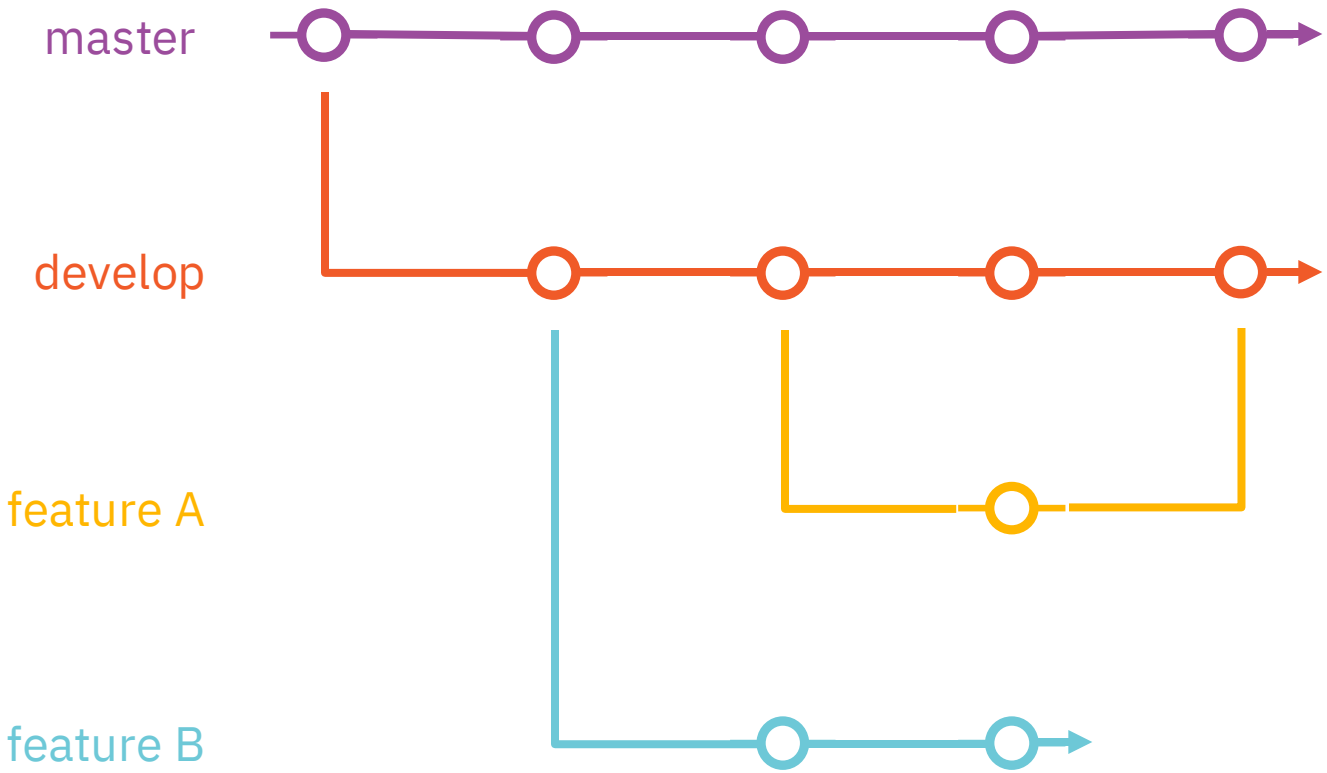
Unlocking source control with Git

IBM i
continuous innovation
continuous integration

Distributed Development



Version Control and Git Workflow



© Copyright IBM Corporation 2024

Projects that self-describe how to build themselves!?

IBM i
continuous innovation
continuous integration

- my-project
  - .evfevent
    - file.evfevent
    - source.evfevent
  - .logs
    - joblog.json
    - file.splf
    - source.splf
  - includes
    - constants.rpgleinc
  - QRPGLESRC
    - .ibmi.json
    - Rules.mk
    - source.rpgle
  - QDDSSRC
    - Rules.mk
    - file.pf
  - .env
  - .gitignore
  - .ibmi.json
  - iproj.json
  - Rules.mk

Project Information

Configure library list

Configure build/compile environment

iproj.json

```
{
    "version": "0.0.1",
    "description": "SAMPLE PROJECT",
    "repository": "https://github.com/edmundreinhardt/bob-recursive-example.git",
    "license": "Apache 2.0",
    "objlib": "&CURLIB",
    "curlib": "&CURLIB",
    "includePath": [
        "includes",
        "QPROTOSRC"
    ],
    "preUsrlibl": [
        "&lib1"
    ],
    "postUsrlibl": [
        "&lib2"
    ],
    "setIBMiEnvCmd": [],
    "compileCommand": "makei c -f {filename}",
    "buildCommand": "makei build"
}
```

Standardized metadata format with variables (&...)

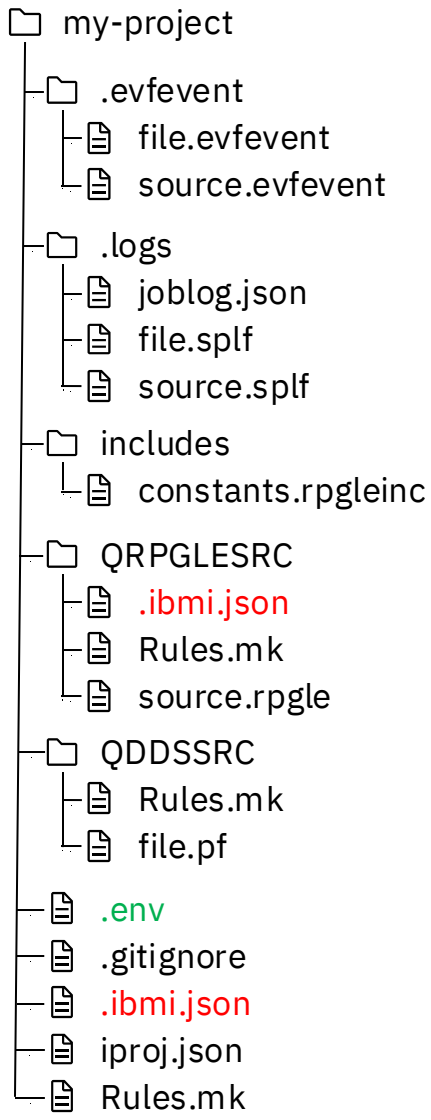
Set build/compile command

iproj.json in project root

© Copyright IBM Corporation 2024

Flexible subdirectories and build customization

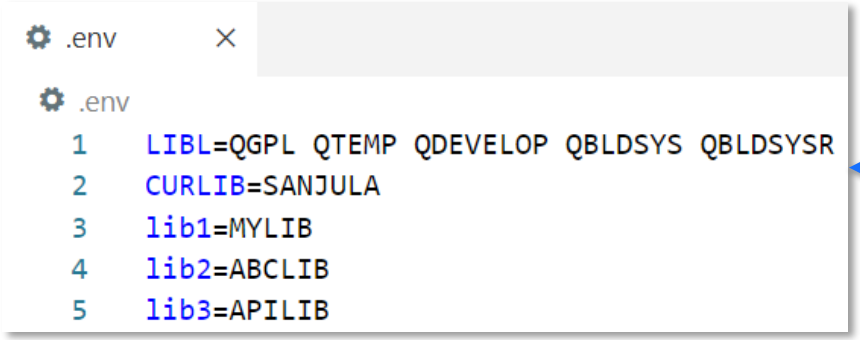
IBM i
continuous innovation
continuous integration



EBCDIC encoding for compiler

Target object library for directory

.ibmi.json in subdirectories



Custom variable values so that each developer can customize build

.env in project root

© Copyright IBM Corporation 2024

Control what objects to build and how to build them

IBM i
continuous innovation
continuous integration

- my-project
  - .evfevent
    - file.evfevent
    - source.evfevent
  - .logs
    - joblog.json
    - file.splf
    - source.splf
  - includes
    - constants.rpgleinc
  - QRPGLESRC
    - .ibmi.json
    - Rules.mk
    - source.rpgle
  - QDDSSRC
    - Rules.mk
    - file.pf
  - .env
  - .gitignore
  - .ibmi.json
  - iproj.json
  - Rules.mk

Rules.mk

```
1   SUBDIRS = qrpglesrc qddssrc
```

Rules.mk in project root

Declare subdirectories to be built

Rules.mk

```
1   FVAT.SRVPGM: fvat.bnd VAT300.MODULE
2   FVAT.SRVPGM: TEXT = Functions VAT
3   FVAT.SRVPGM: private TEXT = Functions VAT
4
5   VAT300.MODULE: vat300.rpgle QPROTOSRC/vat.rpgleinc VATDEF.FILE
6   VAT300.MODULE: private TEXT := bound into FVAT.SRVPGM
7   VAT300.MODULE: private DBGVIEW ::= *SOURCE
8
9   VATDEF.FILE: vatdef.pf SAMREF.FILE
```

Rules.mk in subdirectories

Make style list of objects to be built and from what source files

© Copyright IBM Corporation 2024

Build and Compile Process

IBM i
continuous innovation
continuous integration

Initialization and Migration

Command	Description
makei init	Create iproj.json
makei cvtsrcpf	Convert QSYS members to Unicode IFS stream files

Building

Command	Description
makei build	Build the entire project
makei b –t <object>	Build target object
makei b –d <directory>	Build all objects in the specified directory (based on Rules.mk)

Compiling

Command	Description
makei compile -f <stream file>	Compile target object of specified stream file
makei compile –files file1: file2: …	Compile target objects of all specified stream files

© Copyright IBM Corporation 2024

IBM i
continuous innovation
continuous integration

Ins and Outs of IBM i Project Explorer

© Copyright IBM Corporation 2024

Overview

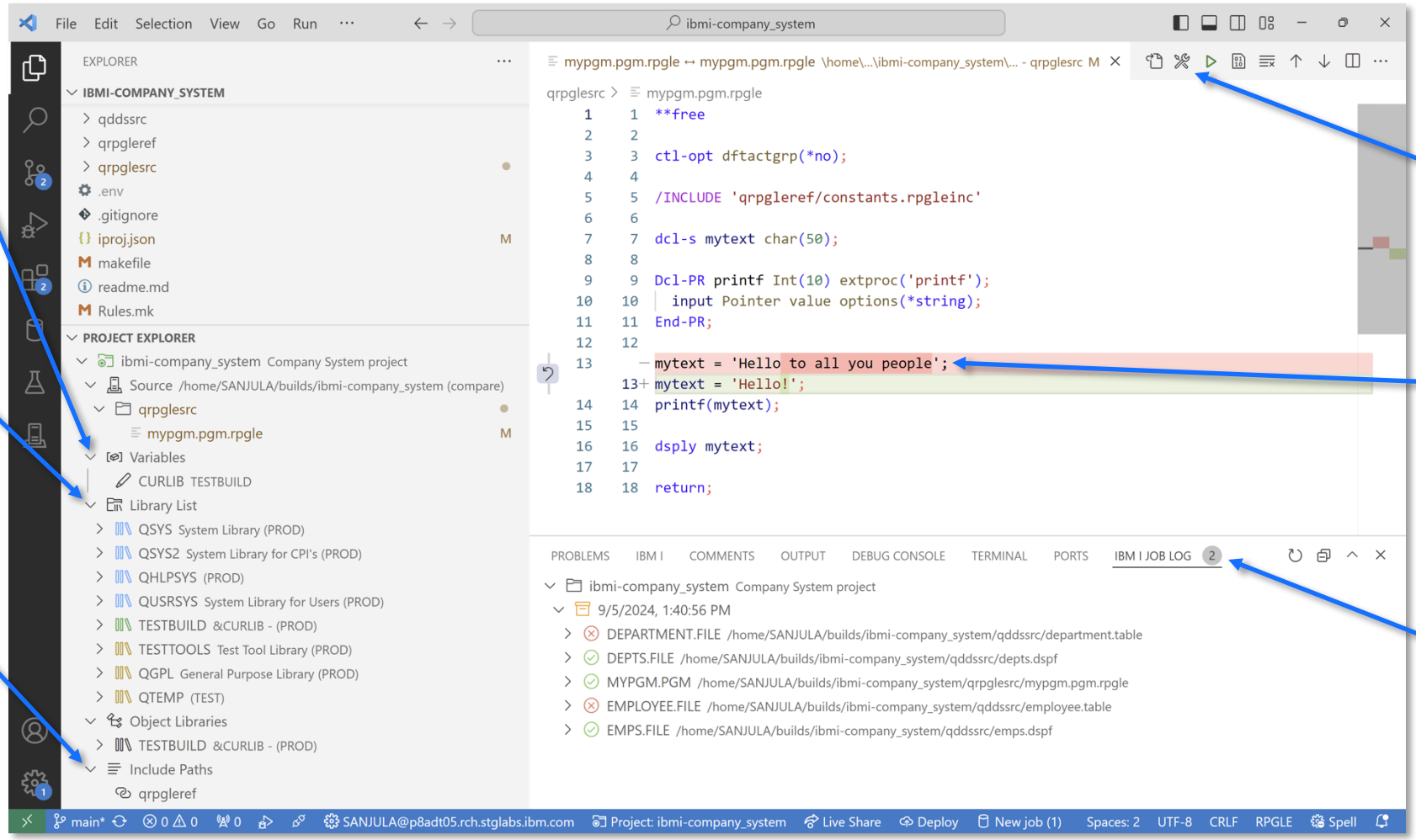
IBM i
continuous innovation
continuous integration

The ultimate tool for local development on IBM i!

Set variables

Manage library list

Modify include paths



Build and Compile

Local source vs. IFS source

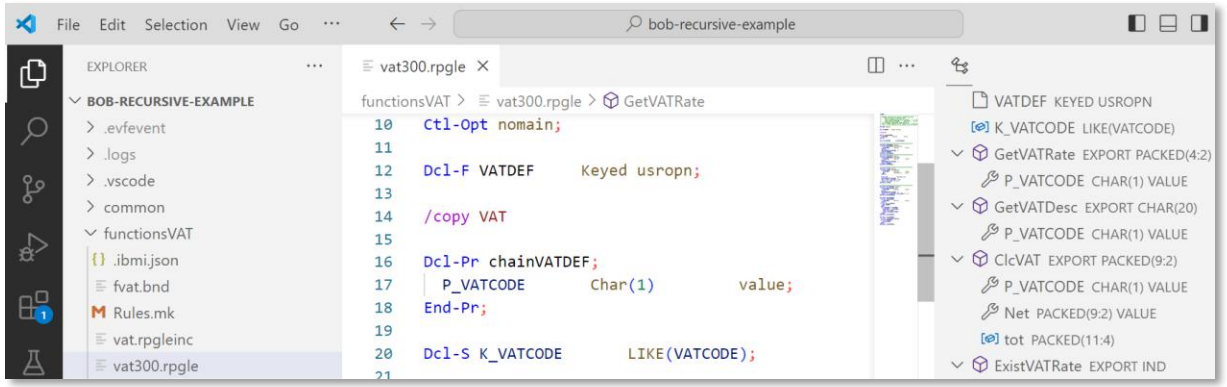
View job logs

© Copyright IBM Corporation 2024

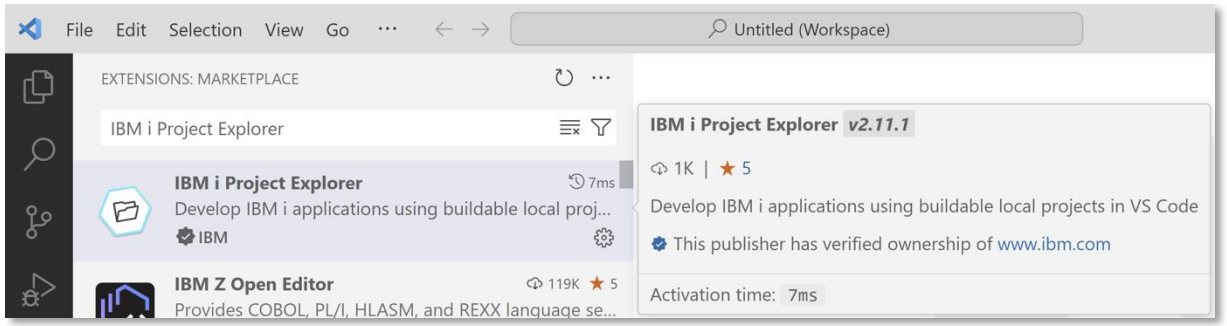
Installation

IBM i
continuous innovation
continuous integration

1. *Download*
   Visual Studio Code



2. *In VS Code install extensions*
   Code for IBM i
   And
   IBM i Project Explorer
   and
   Source Orbit



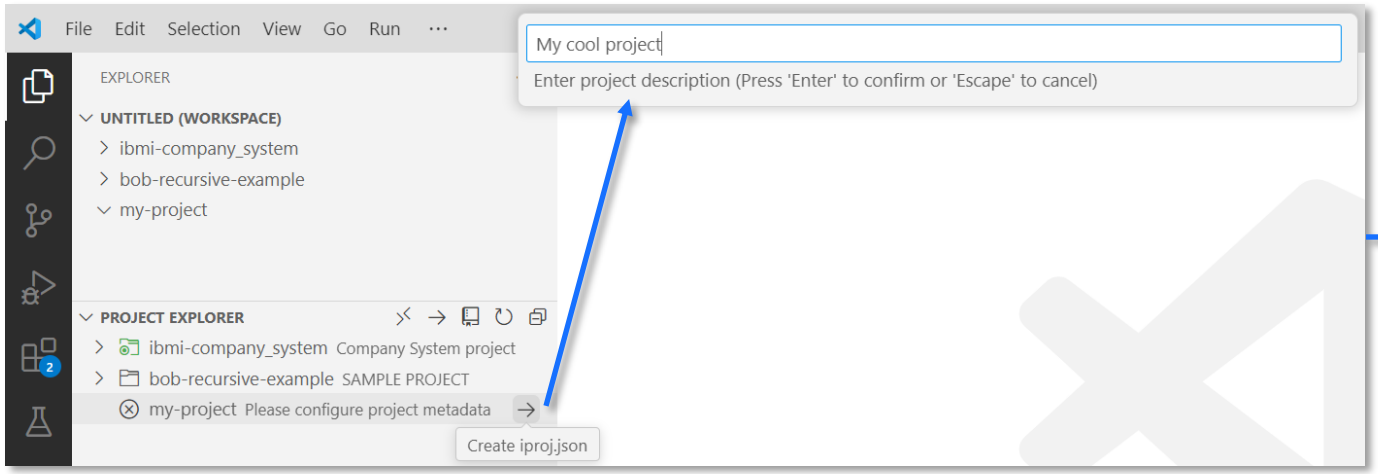
3. *Run*
   yum install bob
   on IBM i

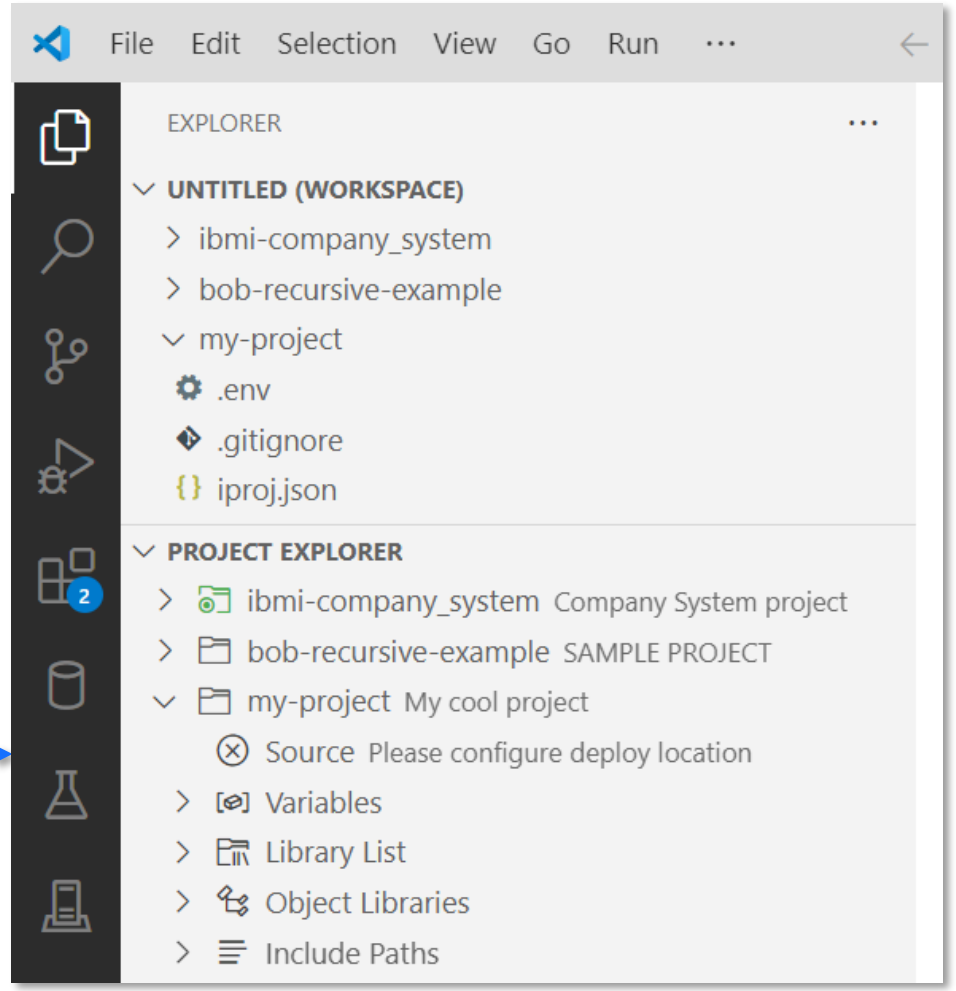
© Copyright IBM Corporation 2024

Create a New Project

IBM i
continuous innovation
continuous integration

- Create and open a folder for your project
- Create an iproj.json
- Set the project description
- Connect to an IBM i (using Code for IBM i)





© Copyright IBM Corporation 2024

Migrate Source from QSYS

IBM i
continuous innovation
continuous integration

CVTSRCPF from BOB

QSYS members in source physical files

↓

Properly encoded, terminated, and named source files in an IFS directory

↓

Download to local project

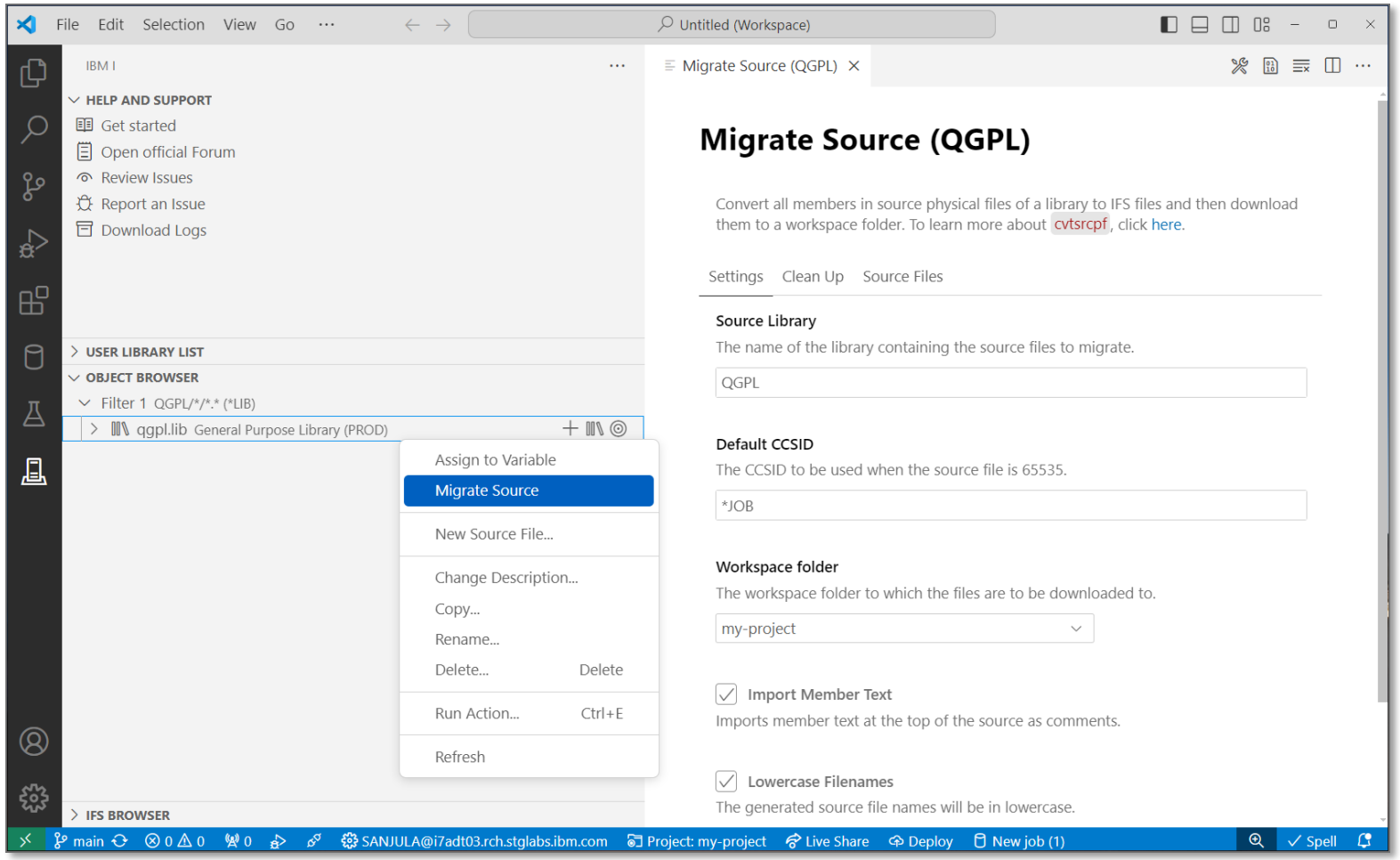
↓

Source Orbit

Rename extensions

↓

Convert includes/copy directives to Unix style paths

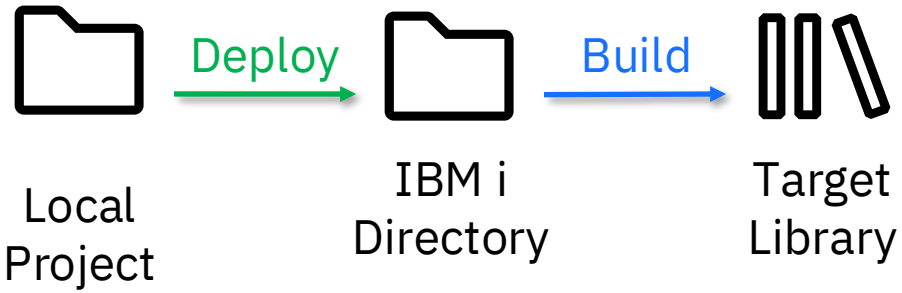


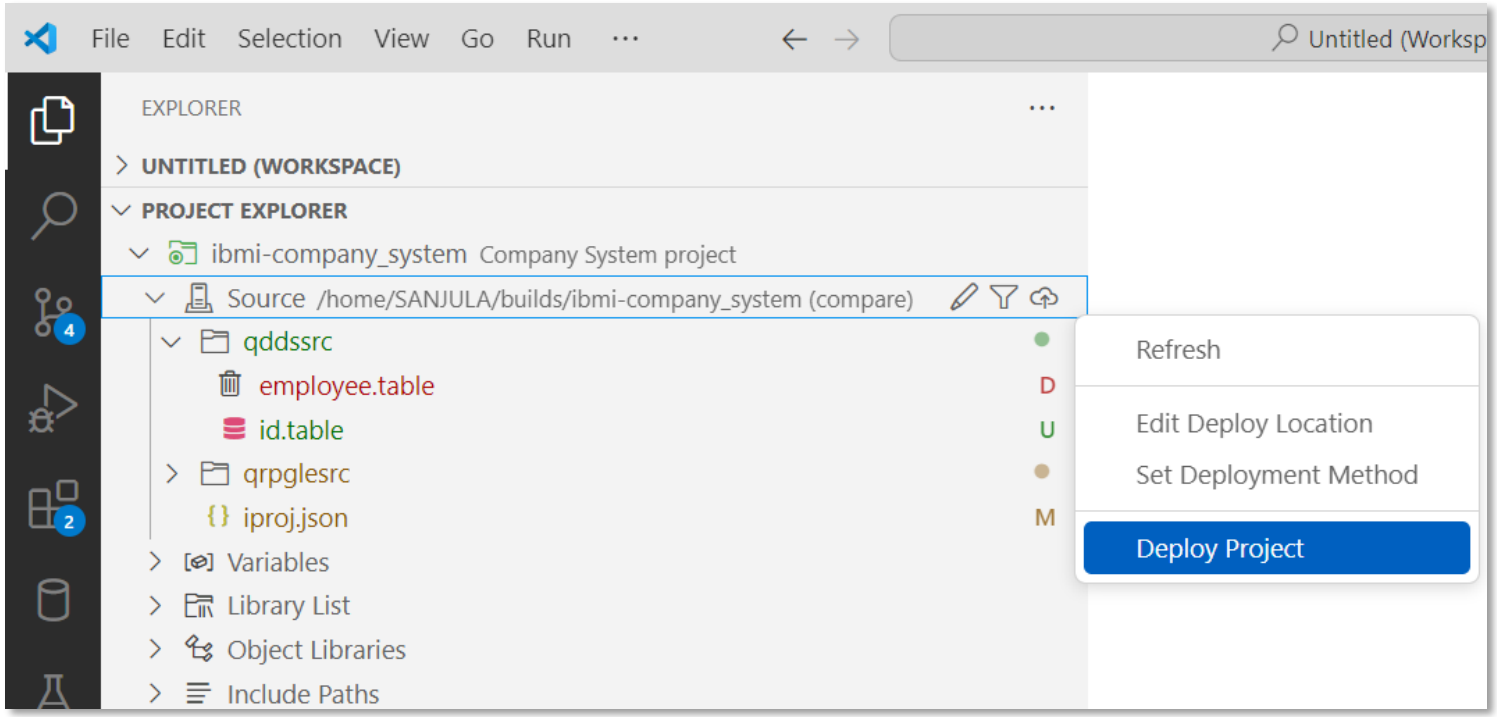
© Copyright IBM Corporation 2024

Source and Deployment

IBM i
continuous innovation
continuous integration

- Set deploy location
  - Where source gets uploaded to
  - Typically set one
  - Each developer gets a unique location
  - Each repository gets a unique location
- Set deployment method
  - Compare (typically the safest)
  - Changes (fastest)
  - Working Changes
  - Staged Changes
  - All
- Deploy project
  - Moves files to deploy location based on deployment method





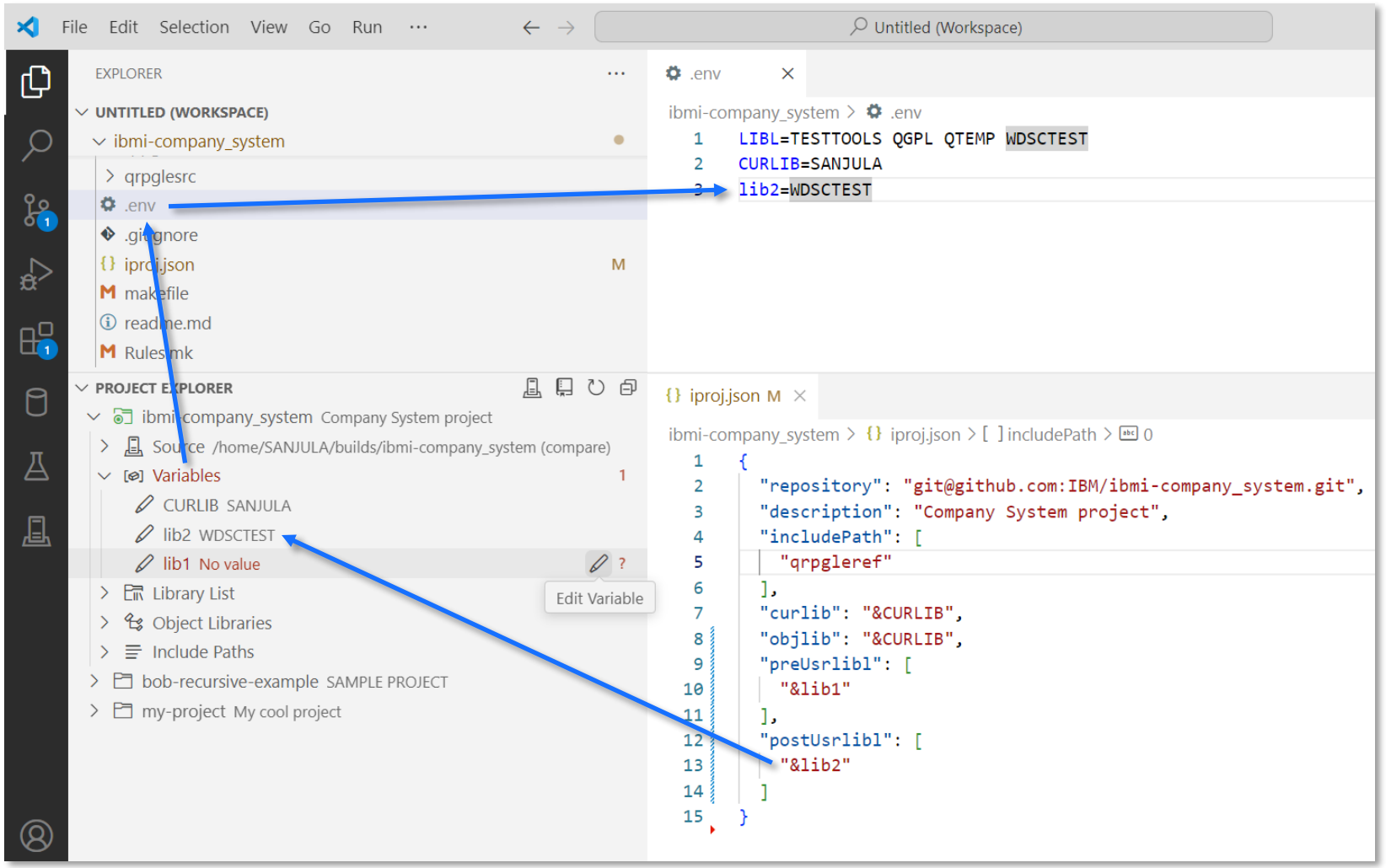
© Copyright IBM Corporation 2024

Work with Variables

IBM i
continuous innovation
continuous integration

- View and set variables (for libraries, include paths, or build/compile commands)
- Browse for libraries and assign values to variables
- Configure hardcoded values as variables

Do not push .env file to Git!

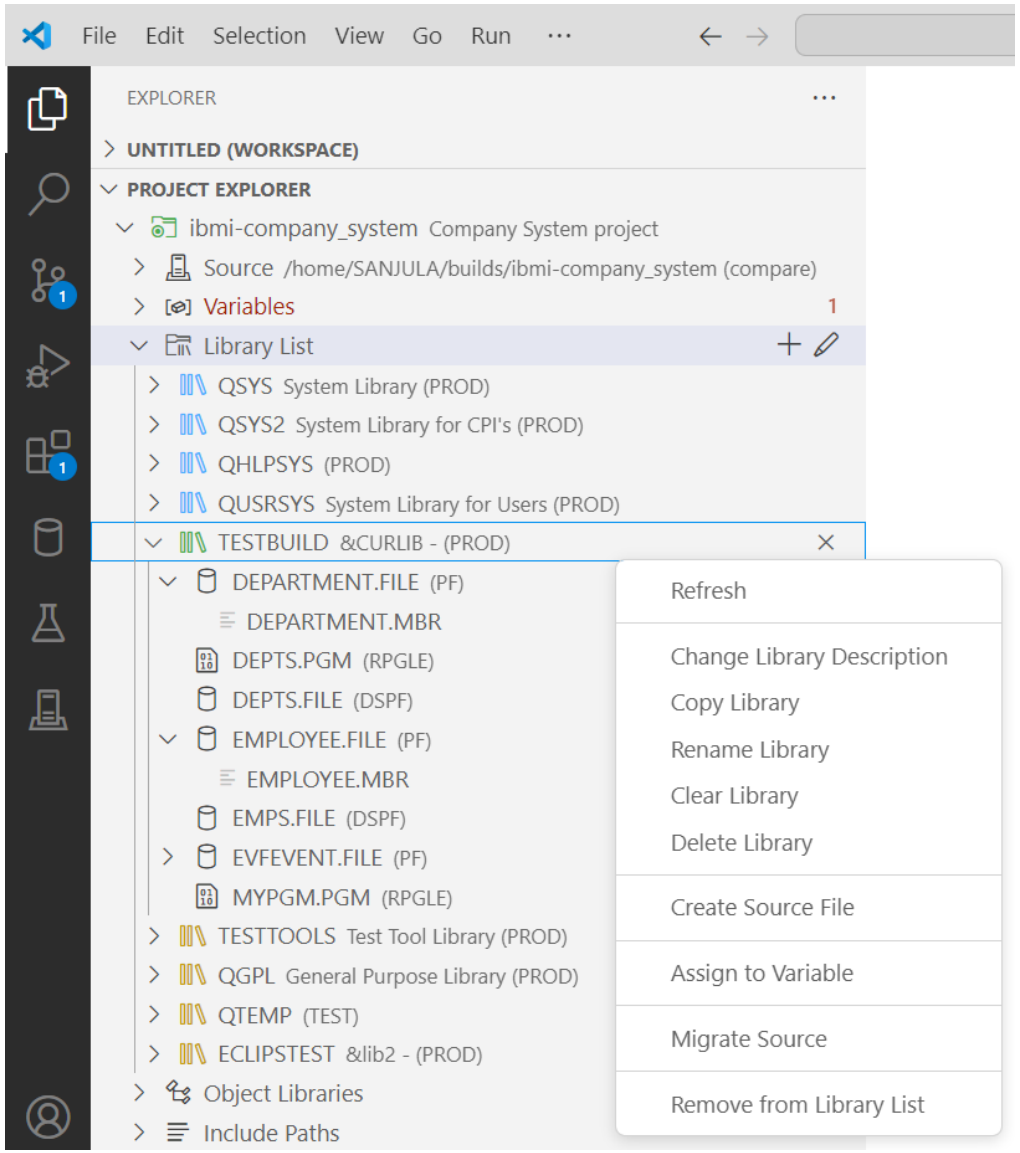


© Copyright IBM Corporation 2024

Manage the Library List

IBM i
continuous innovation
continuous integration

- Add to beginning/end of library list (preUsrlibl and postUsrlibl) and set current library (curlib in iproj.json)
- Reorder library list
- Browse objects and members
- Manage libraries, objects, and members

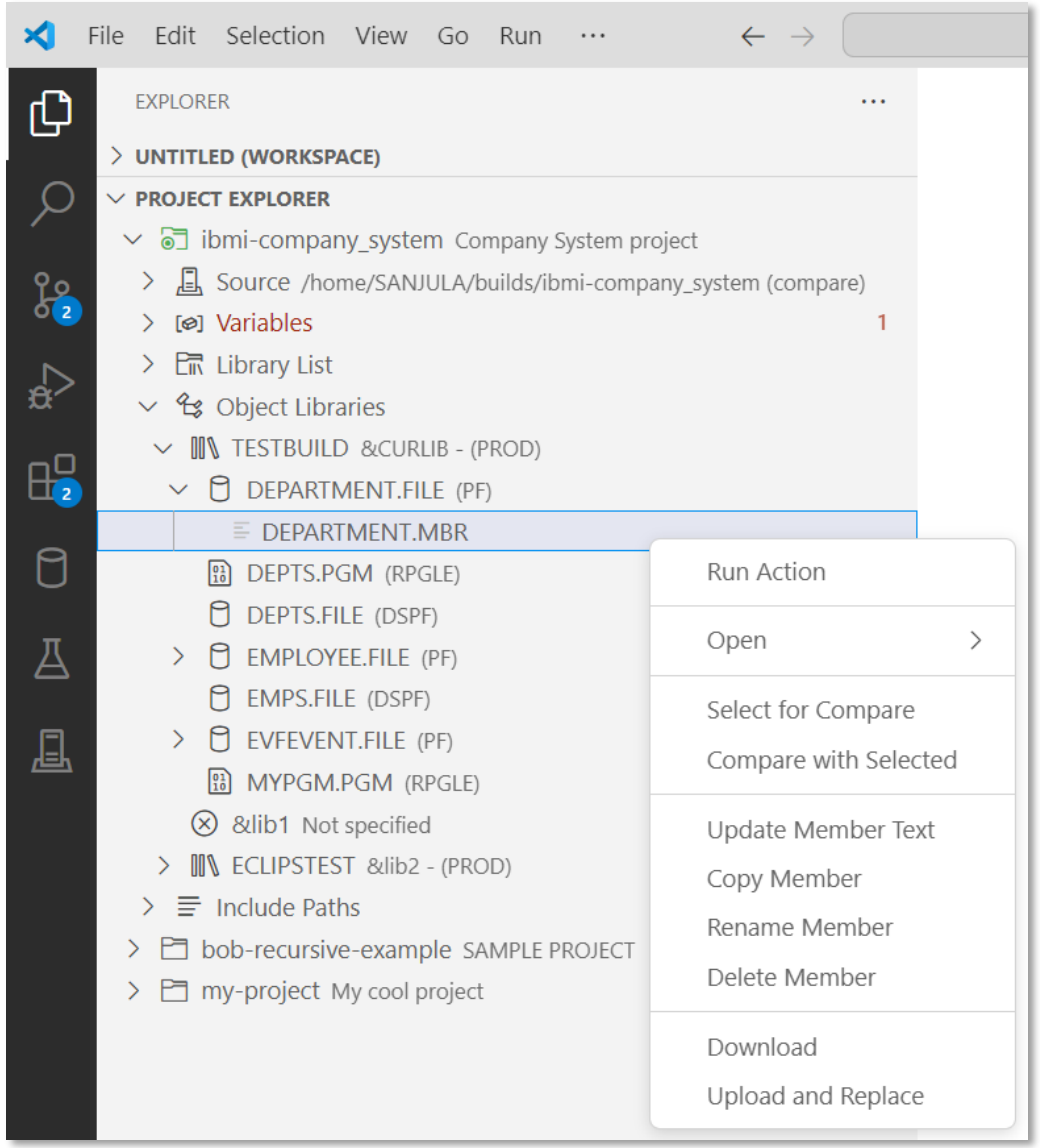


© Copyright IBM Corporation 2024

Browse Object Libraries

IBM i
continuous innovation
continuous integration

- Another place to manage libraries in iproj.json (curlib, objlib, preUsrlibl, postUsrLibl)
- Manage libraries, objects, and members

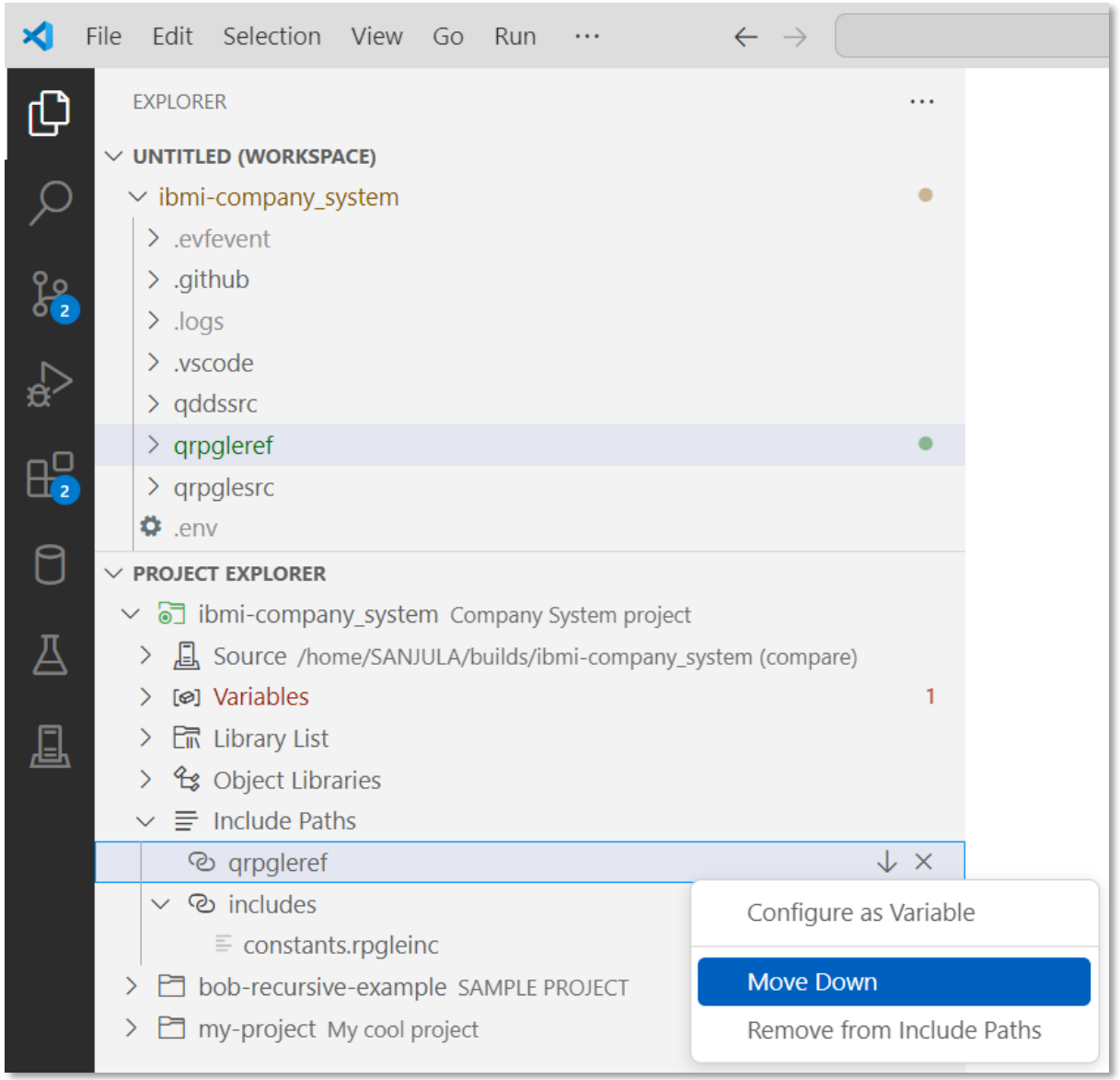


© Copyright IBM Corporation 2024

Manage Include Paths

IBM i
continuous innovation
continuous integration

- Add, remove, and reorder include paths
- Visualize if includes resolve locally or to remote IFS

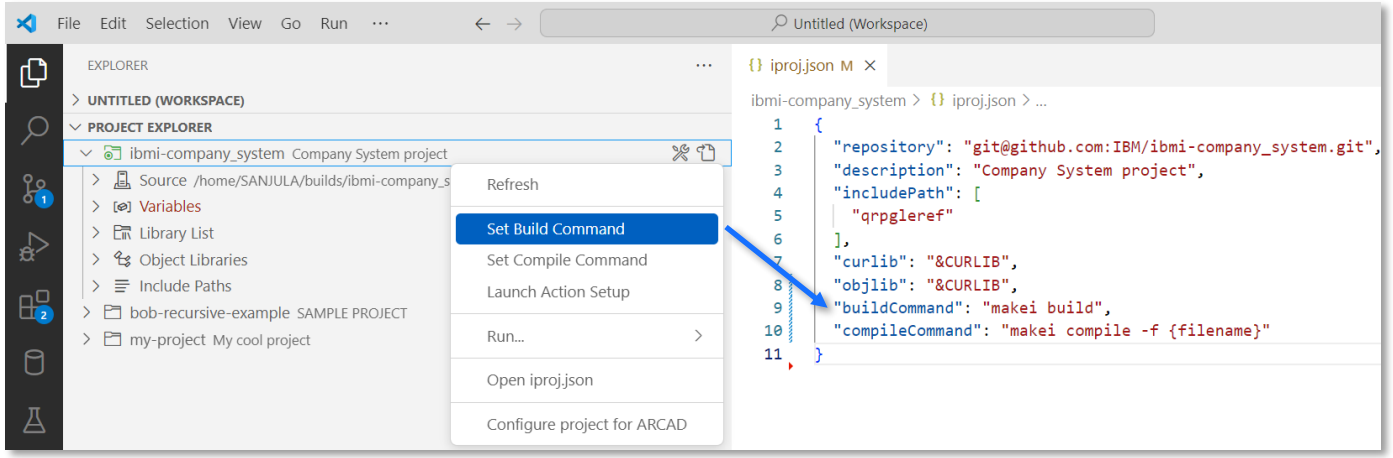


© Copyright IBM Corporation 2024

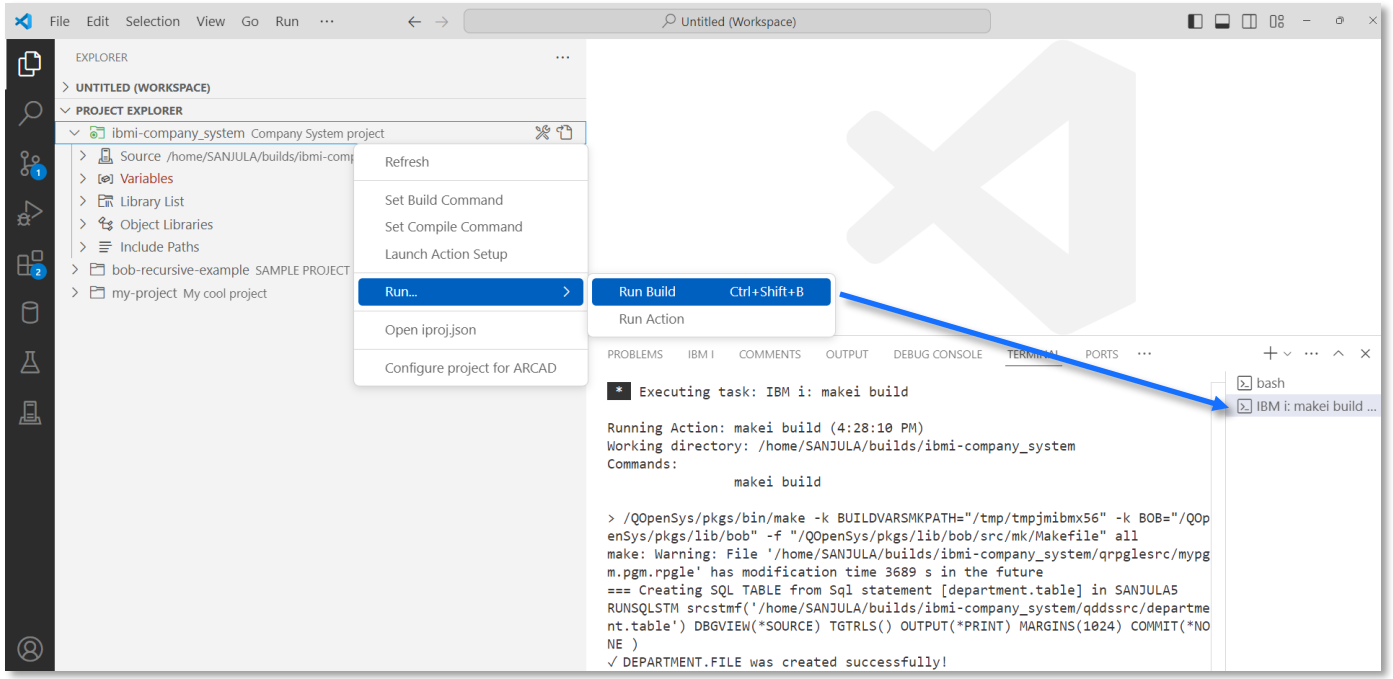
Build and Compile

IBM i
continuous innovation
continuous integration

1. Deploy
↓
2. Run build or compile command
↓
3. Download logs and evfevent files



- Building
  - Set build command
  - Run Build
- Compiling
  - Set compile command
  - Run compile
    - On active editor
    - On file or directory in File Explorer
    - On file or directory in Source

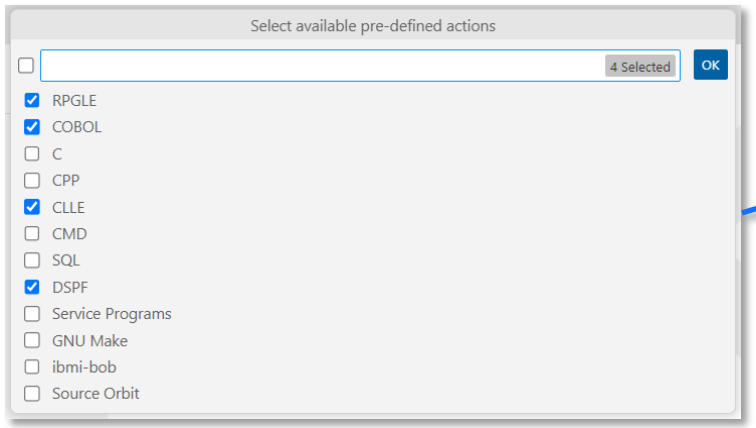


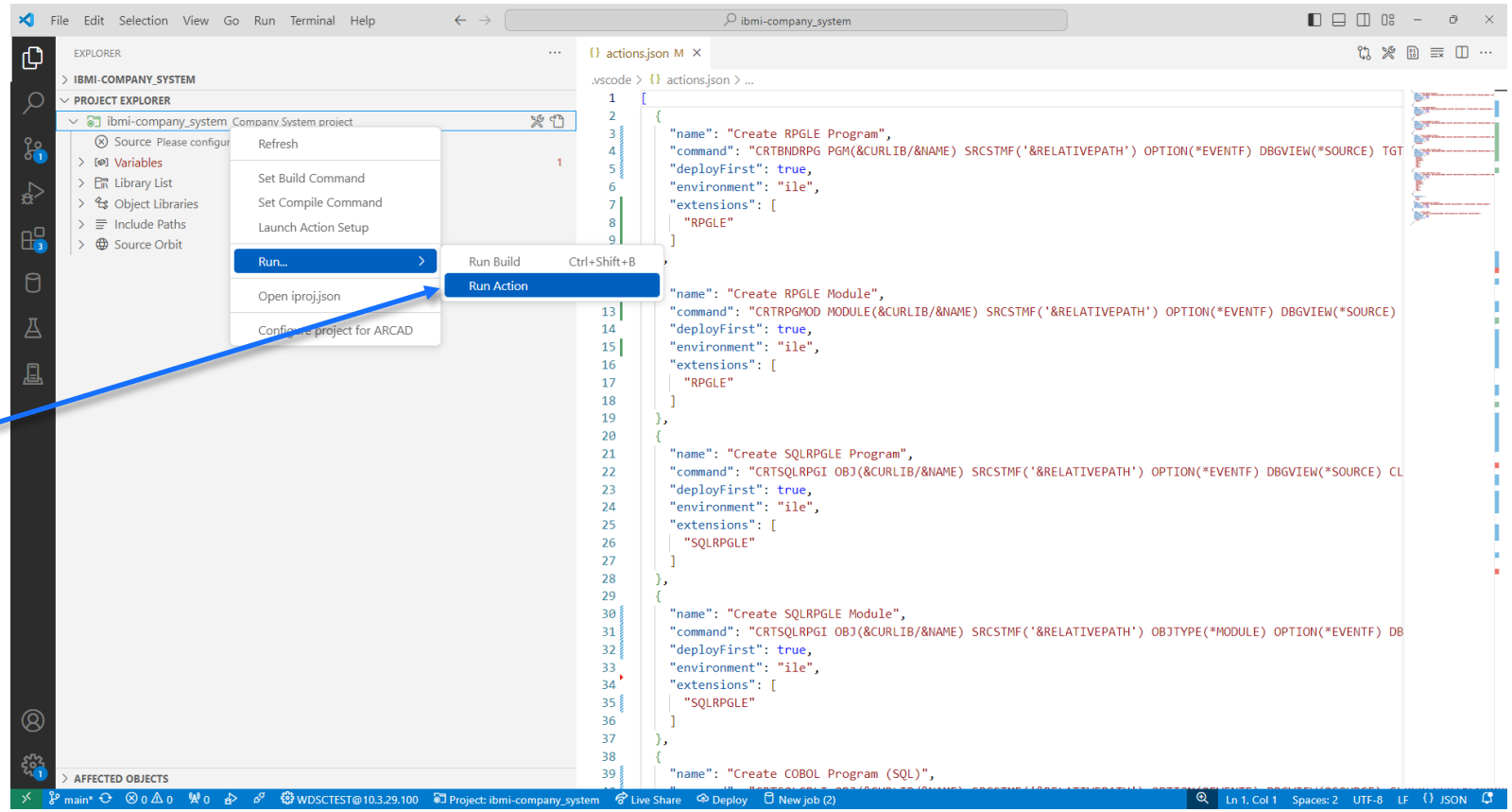
© Copyright IBM Corporation 2024

Run Actions

IBM i
continuous innovation
continuous integration

IBM i Project Explorer also still supports running Code for IBM i's custom workspace actions



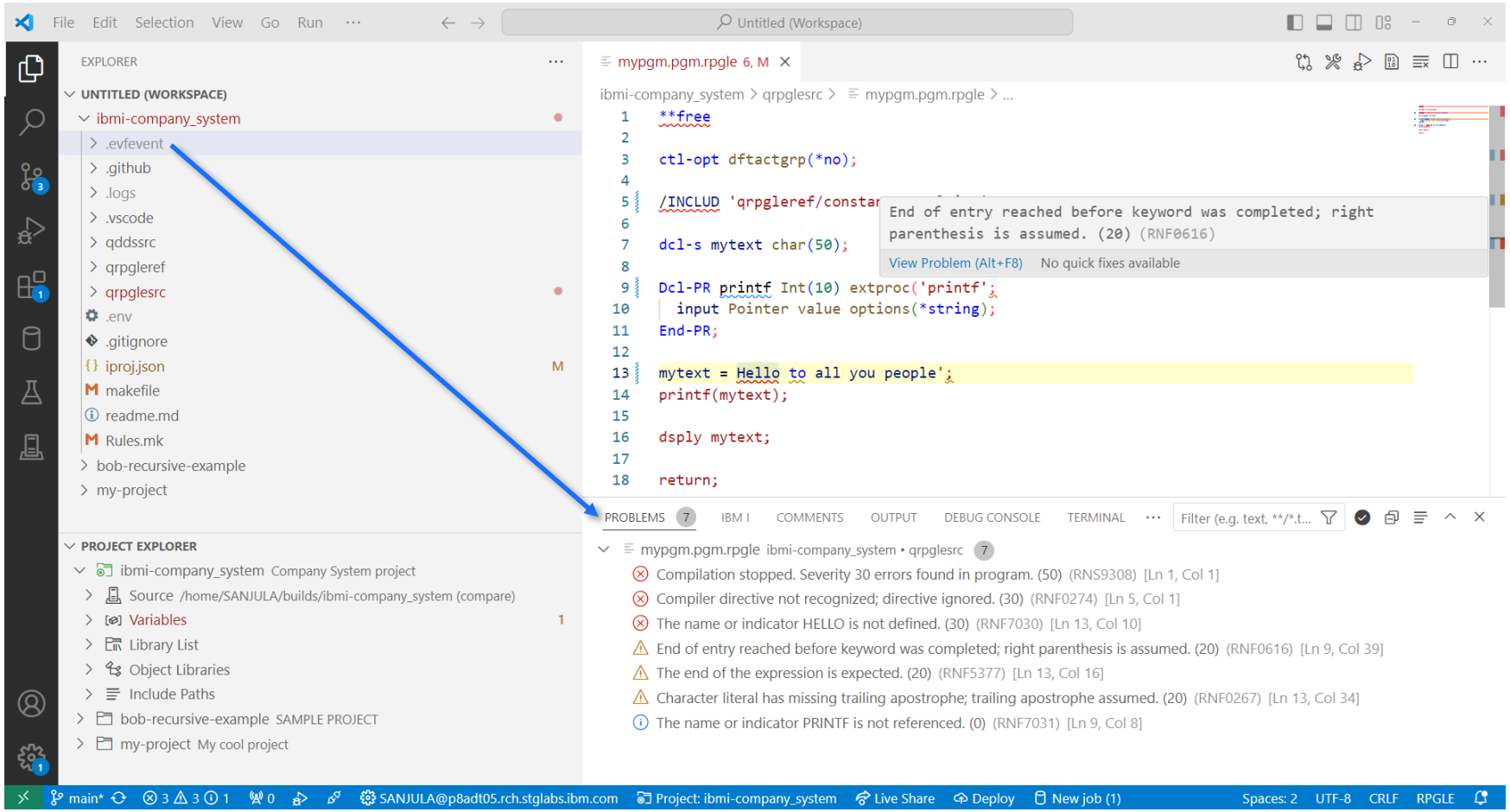


© Copyright IBM Corporation 2024

View Diagnostics

IBM i
continuous innovation
continuous integration

- Evfevent file diagnostics are dumped in .evfevent directory after a build or compile
- Visualize diagnostics in the Problems view
- Diagnostics are also rendered inline in the source file

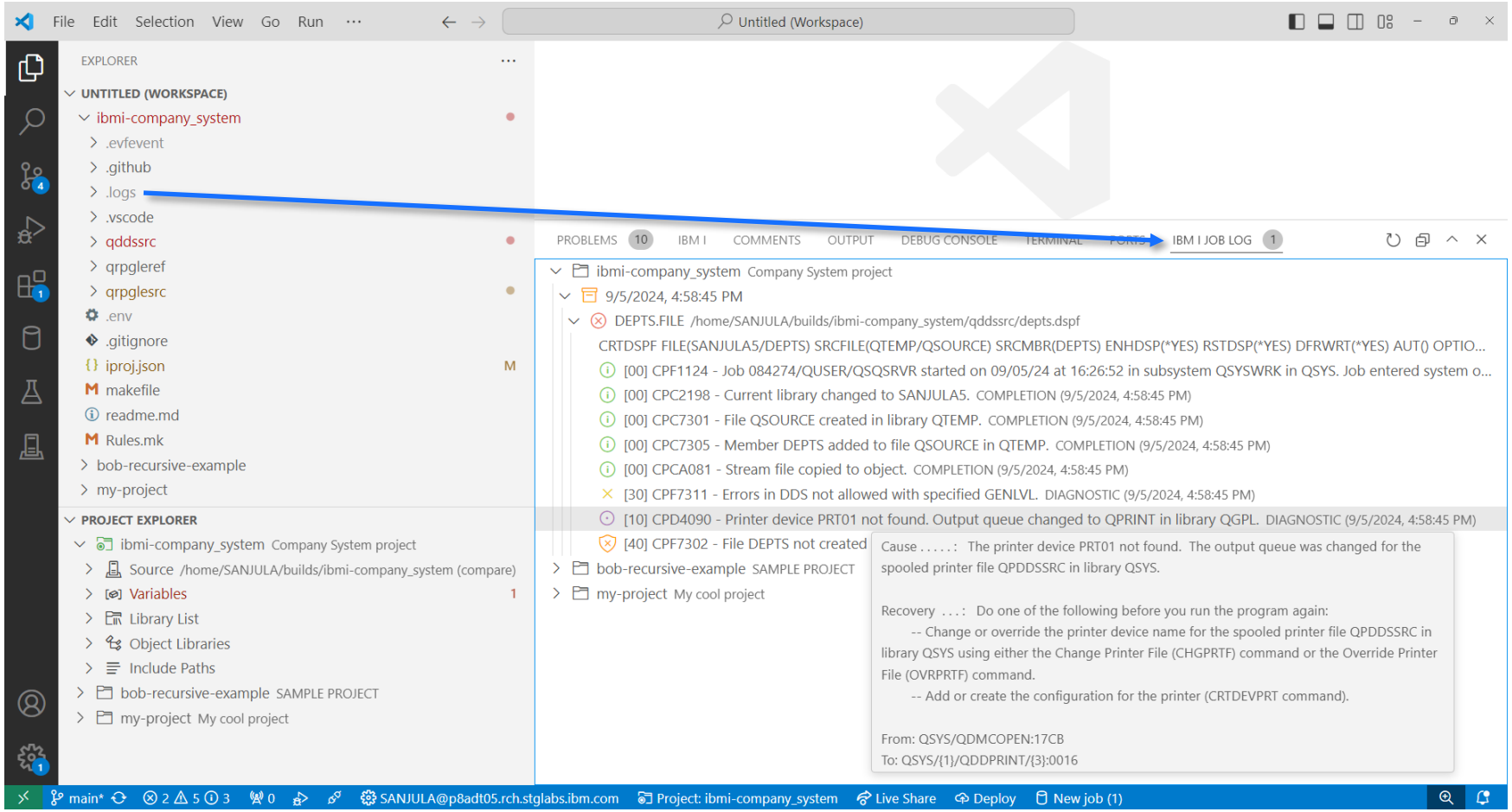


© Copyright IBM Corporation 2024

View Job Logs

IBM i
continuous innovation
continuous integration

- Job log and spool files are dumped in .logs directory after a build or compile
- Job log view is used to visualize and manage these logs
- Track up to 10 of the previous logs in memory
- Organized by the ILE objects being built
- Filter by failed objects or severity



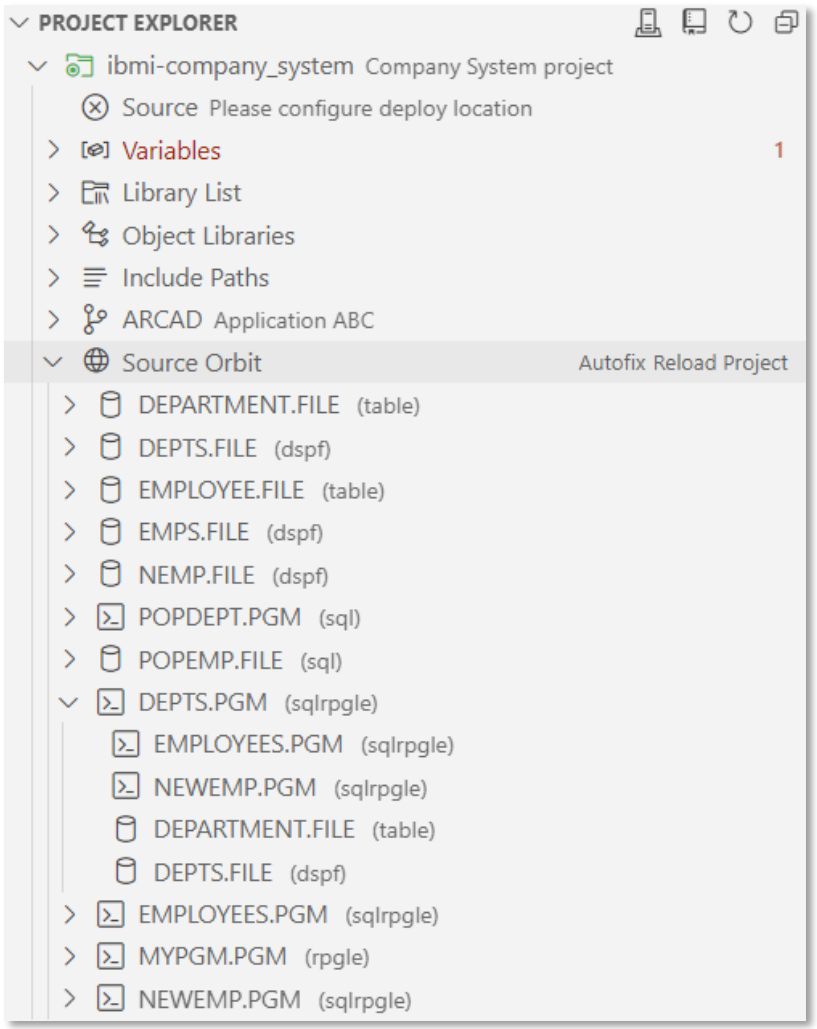
© Copyright IBM Corporation 2024

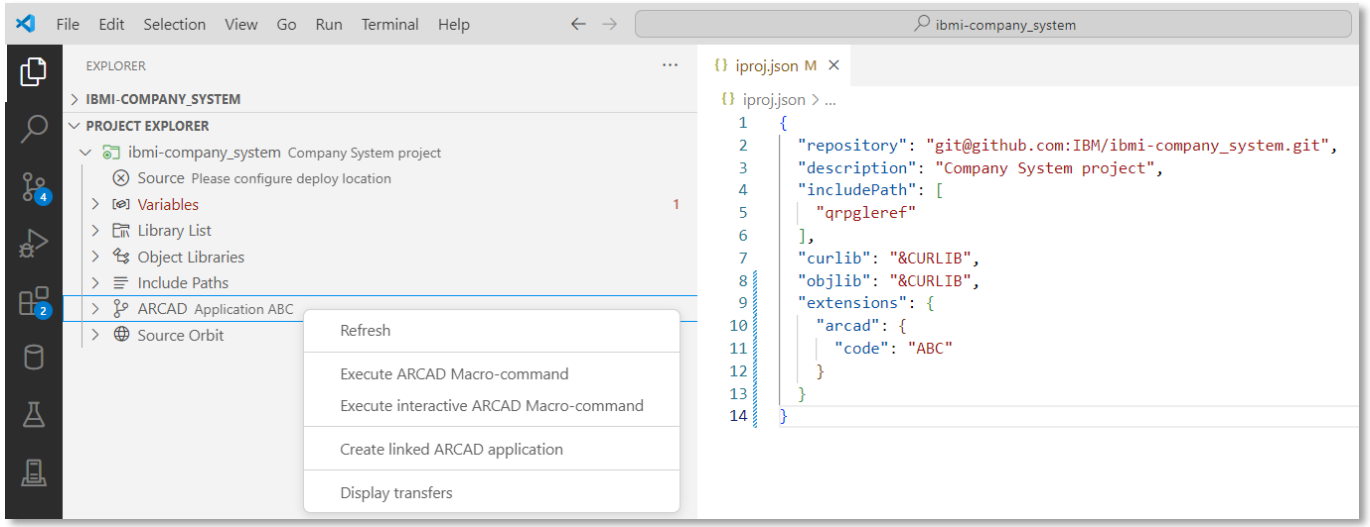
Integration

IBM i
continuous innovation
continuous integration

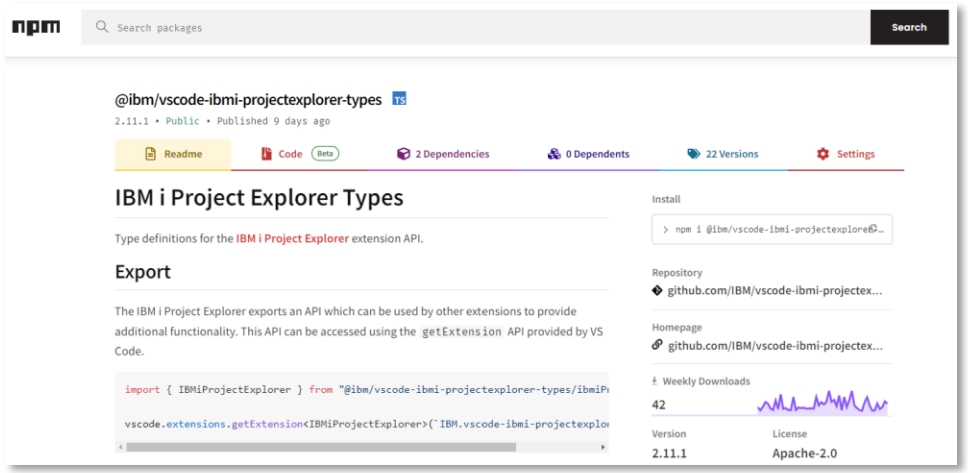
Source Orbit
https://github.com/IBM/sourceorbit

ARCAD-Elias





What can you integrate with IBM i Project Explorer's API?



© Copyright IBM Corporation 2024

Links

IBM i
continuous innovation
continuous integration

IBM i Project Explorer

- VS Code Marketplace https://marketplace.visualstudio.com/items?itemName=IBM.vscode-ibmi-projectexplorer
- Documentation https://ibm.github.io/vscode-ibmi-projectexplorer/#/
- GitHub Repository https://github.com/IBM/vscode-ibmi-projectexplorer
- API https://www.npmjs.com/package/@ibm/vscode-ibmi-projectexplorer-types

Bob

- Documentation https://ibm.github.io/ibmi-bob/#/
- GitHub Repository https://github.com/IBM/ibmi-bob
- Detailed Demo https://www.youtube.com/watch?v=6Xks0NmCUMM

Code for IBM i

- VS Code Marketplace https://marketplace.visualstudio.com/items?itemName=HalcyonTechLtd.code-for-ibmi
- Documentation https://codefori.github.io/docs/#/
- GitHub Repository https://github.com/codefori/vscode-ibmi
- API https://www.npmjs.com/package/@halcyontech/vscode-ibmi-types

© Copyright IBM Corporation 2024

Questions

© Copyright IBM Corporation 2024

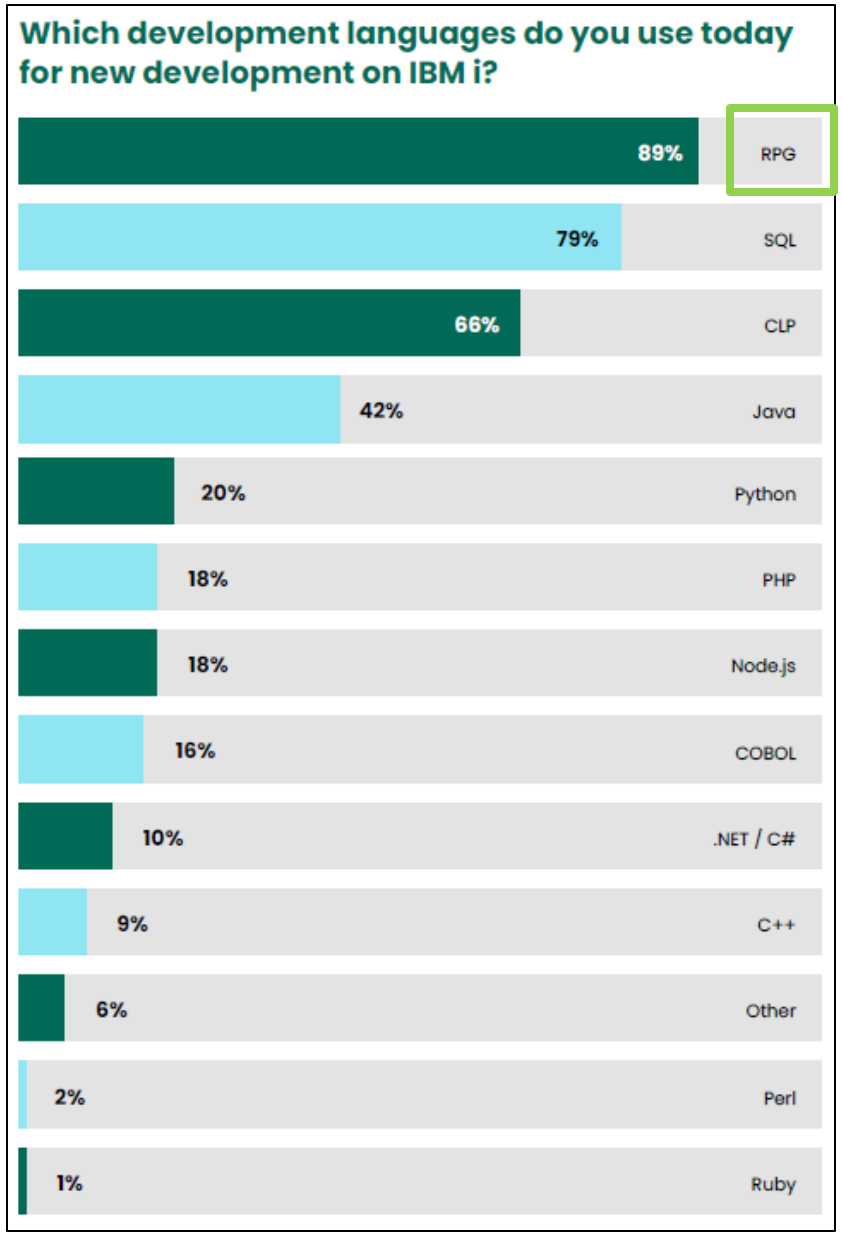
AI-Driven Code Assist for IBM i

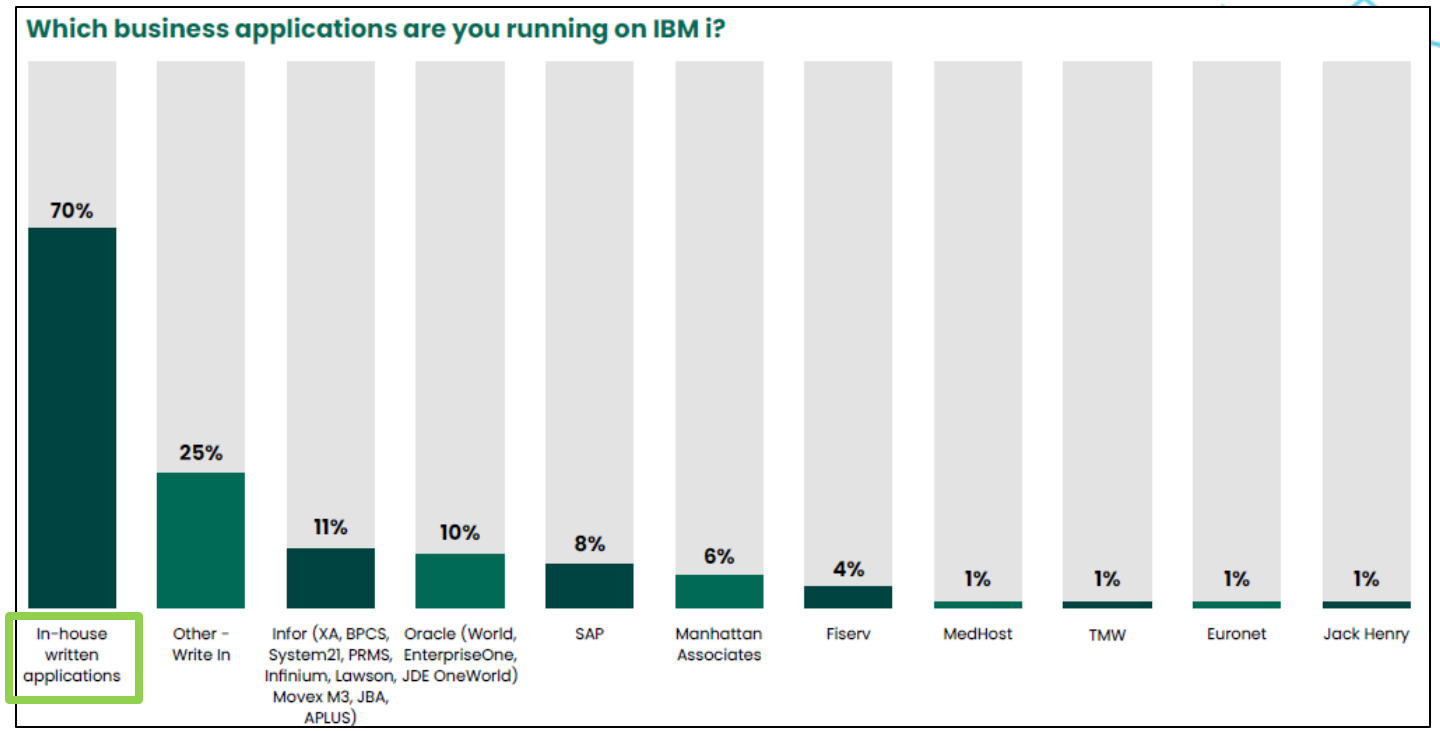
Edmund Reinhardt – IBM i App Dev Tools Architect

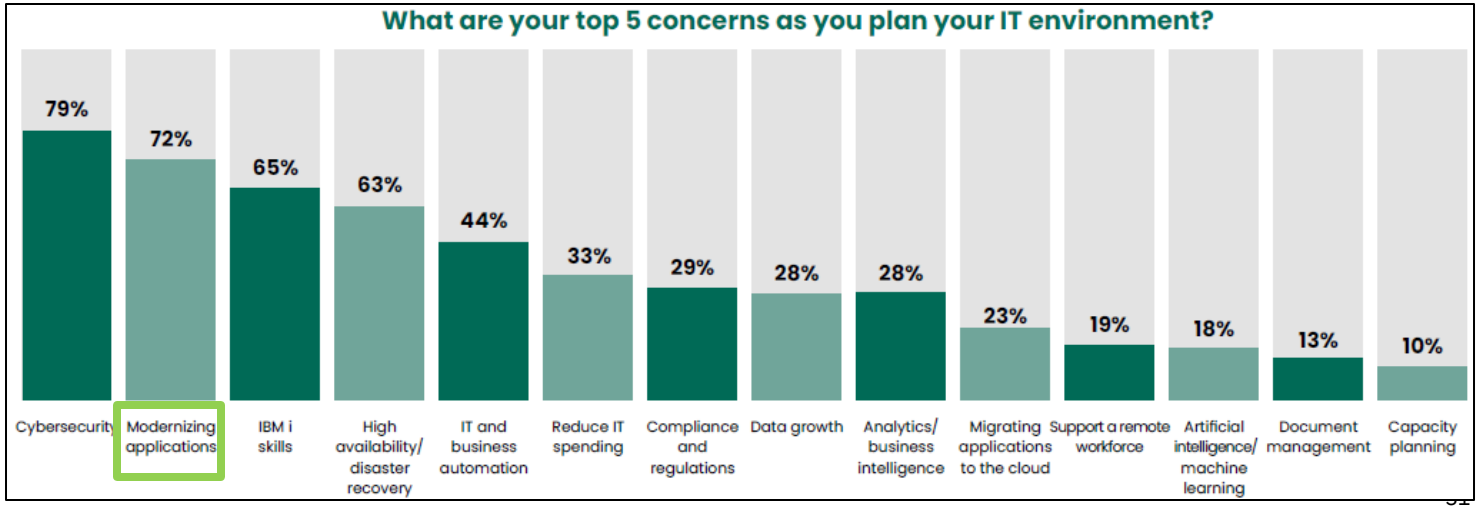
SEMIUG October 15 2024

© Copyright IBM Corporation 2024

2024 IBM i Survey



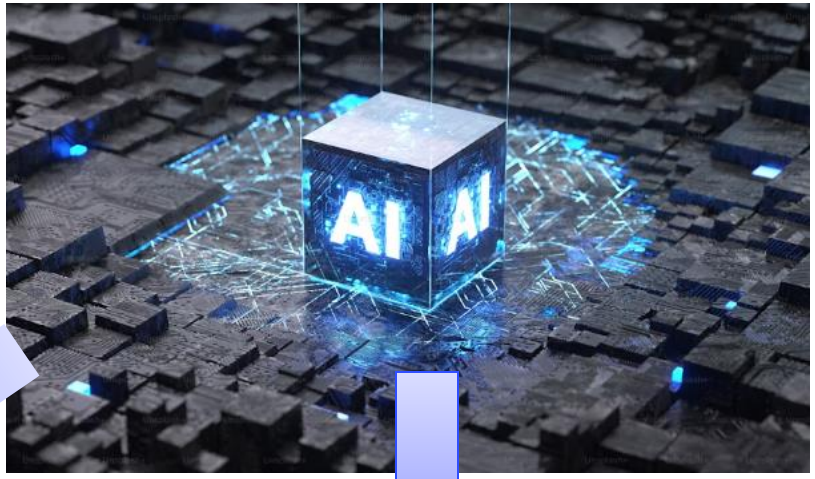




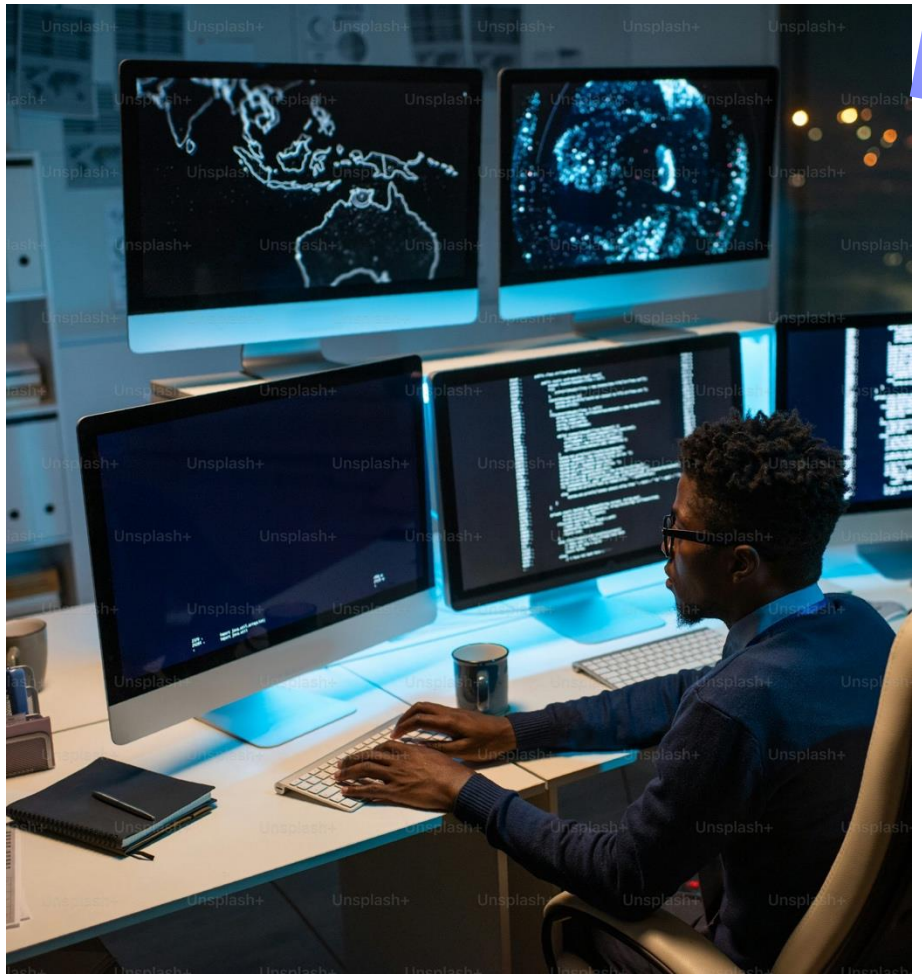
© Copyright IBM Corporation 2024

AI to the rescue!

IBM i
continuous innovation
continuous integration



Improve Application Developer productivity through AI-based coding assistant



Help test and transform older code into modern code

© Copyright IBM Corporation 2024

Customer Pain Points / Personas

"I am used to generating code with AI tools like ChatGPT and RPG is legacy language that I have not yet learned.

Jane
Young Developer who is unfamiliar with RPG

- **Problem statement:** Young Developers are focused on learning and using popular modern programming languages & runtimes like Python, node.js and .NET.
  - New Developers expect to be able to use AI Coding Assistants for RPG just as they do for more popular programming languages like Python, Java, C++.
  - They will not invest time to learn how to program in traditional RPG without the same level of tooling.

WCA4i can enable a young developer to understand what RPG code is doing, generate new RPG code and test cases and transform older RPG code.

- **User value**: Modernize existing RPG core business apps & develop new RPG apps and learn RPG quickly.
- **Client business value:** Modernize core business apps written in RPG faster by enabling new Developers with no RPG experience to understand & generate RPG code.

"I need to deliver RPG core business apps faster but am unfamiliar with the latest programming techniques

Sam
Seasoned RPG Developer unfamiliar with modern programming

- **Problem statement:** Experienced RPG Developers are handicapped by RPG complexity and coding productivity.
  - They have a backlog of new features or core app that need to be developed to support business needs.
  - RPG and IBM i are their strategic platform and they can't switch to other programming languages for their core business applications that their company depends on.

WCA4i can enable experienced RPG developers to increase productivity by auto-generating code, unit and system tests and auto-transforming older code.

- **User value:** Dramatically increase productivity with AI assisted code/test generation & transformation.
- **Client business value:** Drive growth by accelerating delivery of new features or core business apps that deliver customer value or competitive differentiation.

"My core applications are old and difficult to maintain. The technical debt seems like a mountain.

Nadia
Business Executive

- **Problem statement:** Business Executives are reluctant to invest in existing code base because technical debt makes the code difficult to change and validate.
  - Modernizing/developing new RPG apps take a long time
  - Finding and retaining RPG developers is very challenging
  - IBM i not integrating with the rest of the enterprise

WCA4i can enable a new generation of Developers to modernize and transform existing RPG core business applications.

- **User value:** Accelerate ROI for modernizing/developing new RPG apps and increase productivity of Developers.
- **Client business value:** Deliver business innovation driven by RPG core business apps faster and at lower costs.

© Copyright IBM Corporation 2024

Developer Productivity

IBM i
continuous innovation
continuous integration

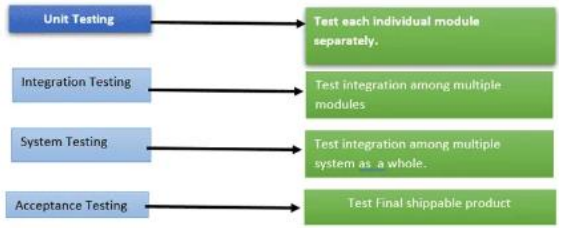
- Help programmers understand existing RPG
- Answer questions about RPG
- Generate modern free-format ILE RPG based on a description / comments
- Provide content assist to complete the current RPG statement or procedure
- Write test programs for RPG

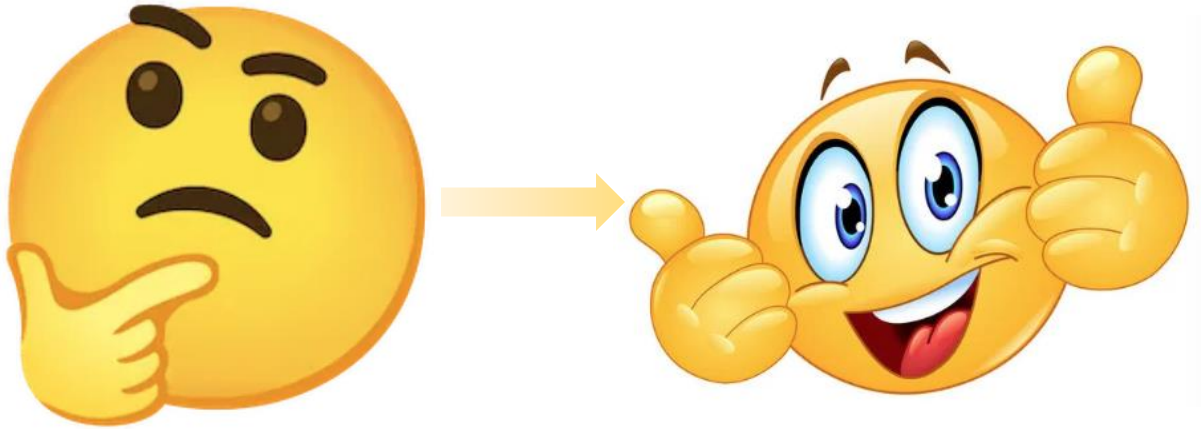
© Copyright IBM Corporation 2024

Vision: Let AI help you modernize your code

IBM i
continuous innovation
continuous integration

- Compartmentalize code so changes have limited impact (subprocedures & modules)
- Expose testable end points and generate comprehensive tests
- Automate build systems to that there is no human error in DevOps pipeline
- Improve readability and maintainable and learning experience for newcomers
- Gain performance





© Copyright IBM Corporation 2024

Explain existing RPG

IBM i
continuous innovation
continuous integration

Input

RPG Code:

```
**FREE
Ctl-Opt nomain;

Dcl-F VATDEF    Keyed usropn;

/copy VAT

Dcl-Pr chainVATDEF;
 P_VATCODE     Char(1)      value;
End-Pr;

Dcl-S K_VATCODE     LIKE(VATCODE);

//=========================================
Dcl-Proc GetVATRate export;
 Dcl-Pi GetVATRate Packed(4:2);
  P_VATCODE     Char(1)      value;
 End-Pi;
 chainVATDEF(P_VATCODE  );
 return VATRATE;
End-Proc GetVATRate;
//=========================================
Dcl-Proc GetVATDesc export;
 Dcl-Pi GetVATDesc Char(20);
  P_VATCODE     Char(1)      value;
 End-Pi;
 chainVATDEF(P_VATCODE  );
 return VATDESC;
End-Proc GetVATDesc;
```

Output

1. Business Purpose:
The RPG code provided is designed to handle Value Added Tax (VAT) calculations for a business. It allows for the retrieval of VAT rates, descriptions, and the calculation of VAT amounts based on a given net amount and VAT code. The program also maintains a file of VAT definitions and can handle opening and closing of the file as needed.

2. Inputs and Outputs:
The program has three main procedures that can be exported for use in other modules or programs:
- GetVATRate: This procedure takes a single character input representing a VAT code and returns the corresponding VAT rate as a packed decimal.
- GetVATDesc: This procedure takes a single character input representing a VAT code and returns the corresponding VAT description as a character string of up to 20 characters.
- ClcVAT: This procedure takes two parameters, a single character VAT code and a packed decimal net amount, and returns the calculated VAT amount as a packed decimal.

3. Internal Procedures:
- chainVATDEF: This procedure is used internally to open the VAT definitions file if it's not already open, set the key for the file based on the input VAT code, retrieve the corresponding VAT record, and close the file if necessary.
- closeVATDEF: This procedure is used to close the VAT definitions file if it's open.

4. Functional Summary:
The program starts by declaring the necessary file and data structures. It then defines the three main procedures that can be called from other modules. The chainVATDEF procedure is responsible for opening the VAT definitions file if it's not already open, setting the key for the file based on the input VAT code, retrieving the corresponding VAT record, and closing the file if necessary. The GetVATRate procedure uses chainVATDEF to retrieve the VAT rate for a given VAT code, the GetVATDesc procedure retrieves the VAT description, and the ClcVAT procedure calculates the VAT amount based on the input net amount and VAT code. Finally, the program includes error handling and file closure procedures to ensure proper file management.

© Copyright IBM Corporation 2024

1. Business Purpose:
The RPG code provided is designed to handle Value Added Tax (VAT) calculations for a business. It allows for the retrieval of VAT rates, descriptions, and the calculation of VAT amounts based on a given net amount and VAT code. The program also maintains a file of VAT definitions and can handle opening and closing of the file as needed.

2. Inputs and Outputs:
The program has three main procedures that can be exported for use in other modules or programs:
- GetVATRate: This procedure takes a single character input representing a VAT code and returns the corresponding VAT rate as a packed decimal.
- GetVATDesc: This procedure takes a single character input representing a VAT code and returns the corresponding VAT description as a character string of up to 20 characters.
- ClcVAT: This procedure takes two parameters, a single character VAT code and a packed decimal net amount, and returns the calculated VAT amount as a packed decimal.

3. Internal Procedures:
- chainVATDEF: This procedure is used internally to open the VAT definitions file if it's not already open, set the key for the file based on the input VAT code, retrieve the corresponding VAT record, and close the file if necessary.
- closeVATDEF: This procedure is used to close the VAT definitions file if it's open.

4. Functional Summary:
The program starts by declaring the necessary file and data structures. It then defines the three main procedures that can be called from other modules. The chainVATDEF procedure is responsible for opening the VAT definitions file if it's not already open, setting the key for the file based on the input VAT code, retrieving the corresponding VAT record, and closing the file if necessary. The GetVATRate procedure uses chainVATDEF to retrieve the VAT rate for a given VAT code, the GetVATDesc procedure retrieves the VAT description, and the ClcVAT procedure calculates the VAT amount based on the input net amount and VAT code. Finally, the program includes error handling and file closure procedures to ensure proper file management.

Based on the prompt can get good summary of uncomment code

In IDE support

IBM i
continuous innovation
continuous integration

Explorer (⇧⌘E)

@Codebase what does the ReasonMaintenance procedure in PAYROLL.pgm.RPGLE doe

watsonx - Mistral Large @ ⌘⏎ Use @codebase ⏎ Enter

28 context items

The `ReasonMaintenance` procedure in `PAYROLL.pgm.RPGLE` is responsible for handling the maintenance of reason codes within the time reporting system. This procedure allows users to add, change, or delete reason codes. Here's a detailed breakdown of what the `ReasonMaintenance` procedure does:

Overview

The `ReasonMaintenance` procedure is called when the user selects the reason code maintenance option from the main menu. It handles the display of the reason code selection and maintenance screens, validates user input, and performs the necessary database operations (add, change, delete) based on the user's actions.

Detailed Steps

1. **Initialization and Housekeeping**

```
DoU *INKC;
  EMESS = *Blanks;
  RSCDE = *Blanks;
  ACODE = *Blanks;
```

   - The procedure initializes error messages (`EMESS`), reason code (`RSCDE`), and action code (`ACODE`) to blank values.

2. **Error Loop**

MSTDSP.DSPF | PAYROLL.pgm.RPGLE

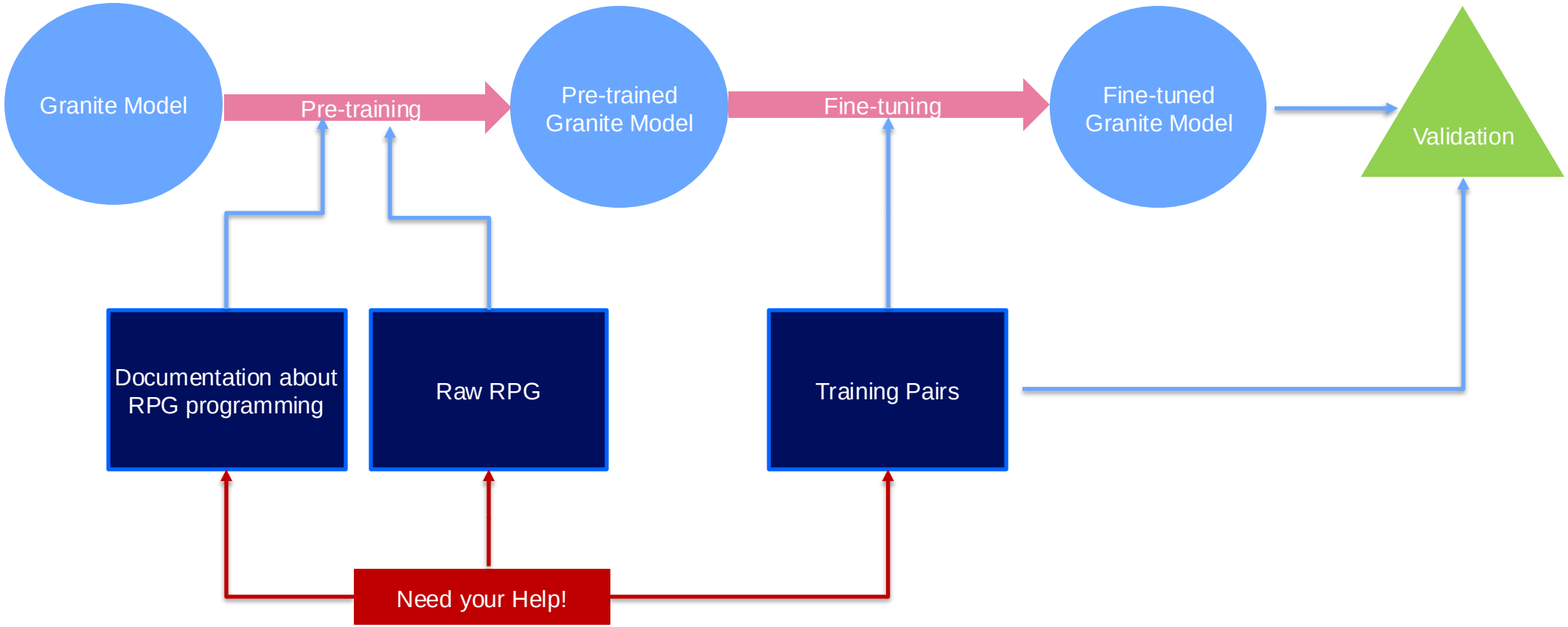
payroll-from-qsys > QRPGLESRC > PAYROLL.pgm.RPGLE

```
39   /copy 'QPROTOSRC/ERRORTABLE.rpgleinc'
49   //   determines if the program will add a new record to the file or
50   //   update an existing record in the file.
51   // ***************************************************
52   //   Housekeeping, clear display fields and reset indicators.
53   //
54   CALLP MAIN();
55
56   //  If MAIN is done program ends
57   *INLR = *on;
58   // ***********************************************************
59   //   End of job requested.  Control is passed to here when you press
60   //   F3 (*INKC).  The last record indicator *INLR is set on and the
61   //   program ends.
62   // ***********************************************************
63
64
65   //  Main screen - select file to maintain
66   //
67   Dcl-Proc MAIN;
68     DoU *INKC;
69       *IN60  = *OFF;
70       EMESS  = *BLANK;
71       EMPAPL = *BLANK;
72       PRJAPL = *BLANK;
73       RSNAPL = *BLANK;
74       //
75       //   Write the SELECT format to display until end of job requested,
76       //
77       DoU not *IN60;
78         //   IF the general error indicator *IN60 is on (equal to 1), the
79         //   program continues to loop
80         Exfmt SELECT;
```


© Copyright IBM Corporation 2024

What IBM does with to train RPG AI

IBM i
continuous innovation
continuous integration



© Copyright IBM Corporation 2024

IBM i Approach – Involve the Community

IBM i
continuous innovation
continuous integration

- IBM i is launching a project to train a “ready for prime time” AI language model-based code assistant
  - Using the RPG code IBM has developed
  - Using code donated by experts & Champions
    - Susan Gantner, Jon Paris, Scott Klement, Jim Buck, Paul Tuohy, Niels Liisberg
    - Yvonne Enselman, Mats Lidström, Koen DeCorte, Steve Bradshaw
    - … and we hope to have many more!
- Soliciting code for training from the entire IBM i community
  - That could mean YOU!
- The project for a community-trained Large Language Model has begun!

© Copyright IBM Corporation 2024

So How Do You Get Involved?

IBM i
continuous innovation
continuous integration

- Let us know you want to help
  - e-mail **AIforIBMi@ibm.com**

- Agree to the license
  - We'll send it to you – it says we can use your submitted material to train the model

- Submit code and/or training data
  - Information on how is found at: https://ibm.github.io/rpg-genai-data/#/
  - You decide if others (outside IBM) can see & use your code or not

- At some point in the future, we will likely ask for volunteers to actually use the submissions in training & evaluating

© Copyright IBM Corporation 2024

https://ibm.github.io/rpg-genai-data/#/

Welcome
Overview
How to contribute training data
Training Data Format
Task Description and Examples
Explain Existing RPG
Generate new RPG
Generate Unit Tests for Existing RPG
Ask questions about RPG
Fixed to Free Format
Generate a more modern version of RPG
How to contribute RPG source
How to manually contribute to IBM Repo
How to use UI to contribute training data

Training RPG Generative AI

IBM would like to train generative AI models to understand RPG and IBM i application development. We recognize that many customers have valuable assets and expertise and we want to collaborate on building an AI that will help us all.

How to contribute

1. Email AIforIBMi@ibm.com to sign up to contribute to training this AI
2. IBM will reach out and ask you to sign the appropriate legal agreements
3. Once you have signed the appropriate agreements, please provide your github id to the same email AIforIBMi@ibm.com. You can create one here
4. IBM will add you as a contributor to the git repository https://github.com/AIforIBMi/rpg-genai-data where we gathering all of the training data.
5. Now it is time to unleash your expertise. Follow the instructions below

- We would prefer contributing actual question and answer pairs to train the AI following these instructions. We can start fine tuning our model with this data right away.
- It will also be useful to contribute RPG source code following these instructions. If we get sufficient source we can do a general unlabeled training of our model.
- In either case, contributing to the IBM git repository for this data should follow this process

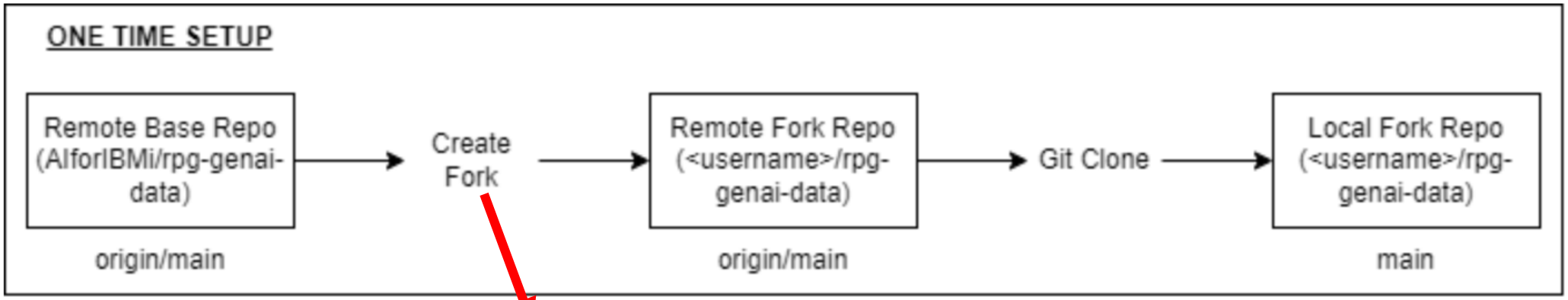
https://ibm.github.io/rpg-genai-data/#/

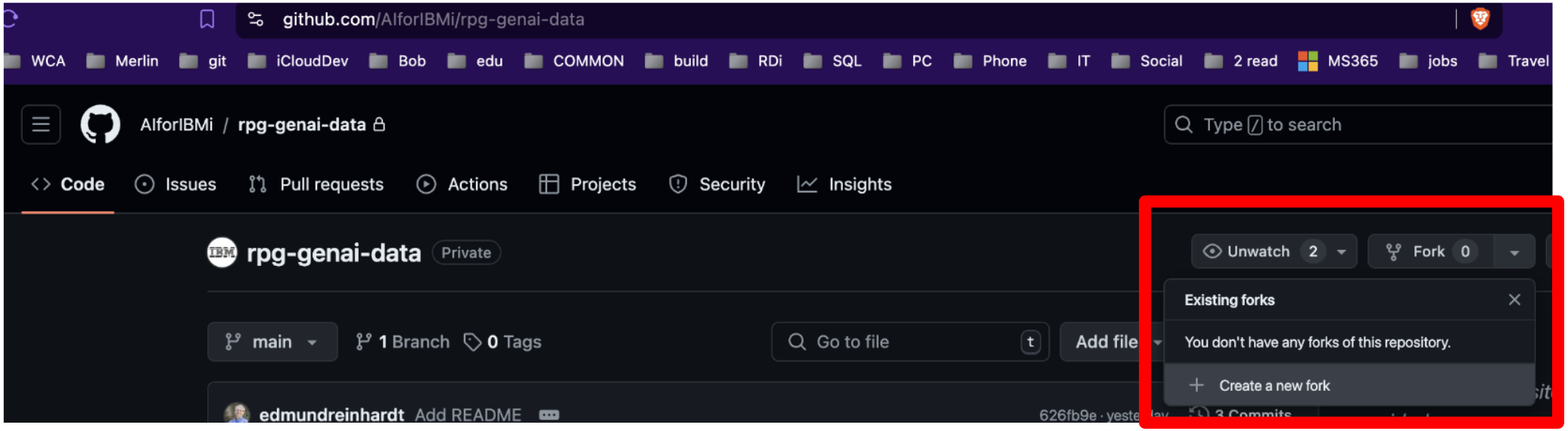
© Copyright IBM Corporation 2024

Github initial setup

IBM i
continuous innovation
continuous integration

The diagram below outlines the one time setup required to setup a forked repository on your local machine:

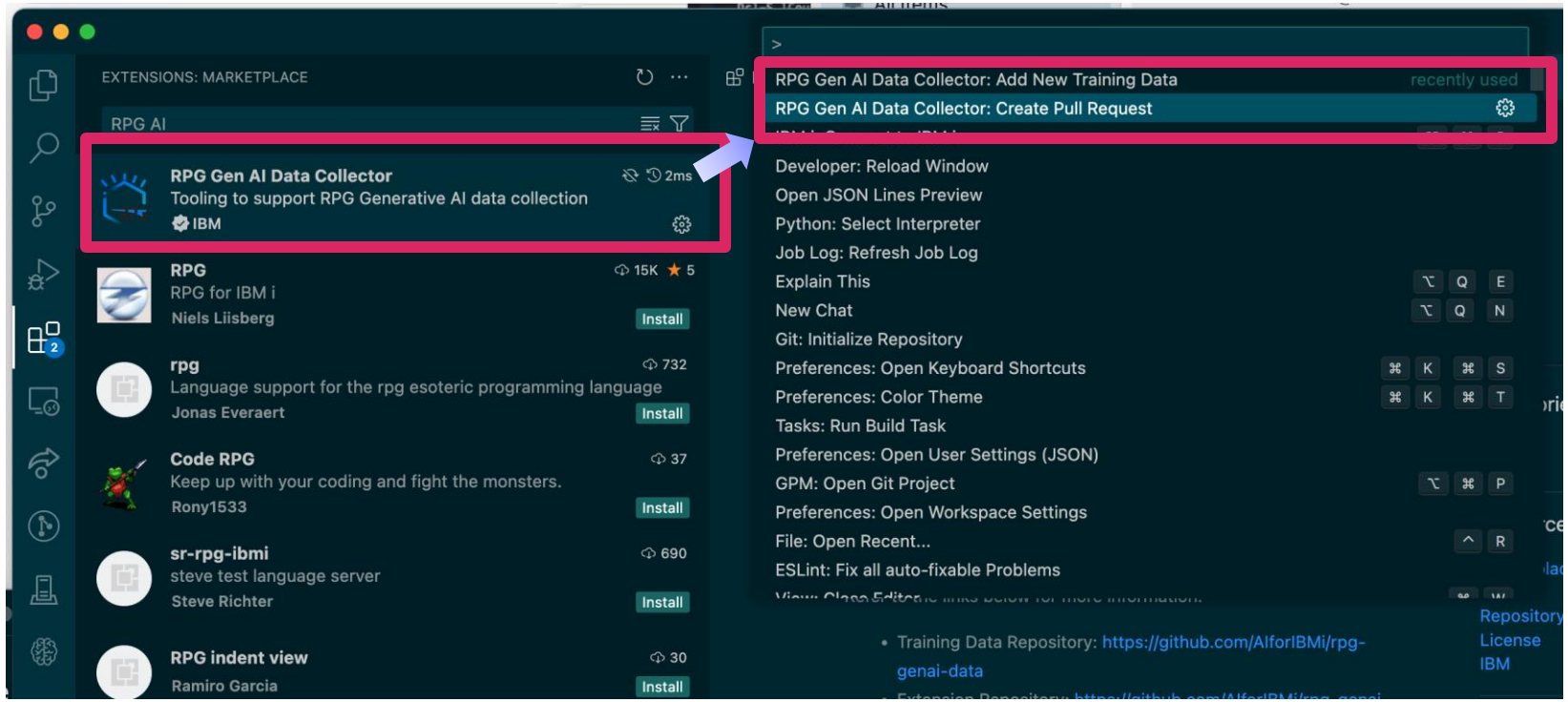




© Copyright IBM Corporation 2024

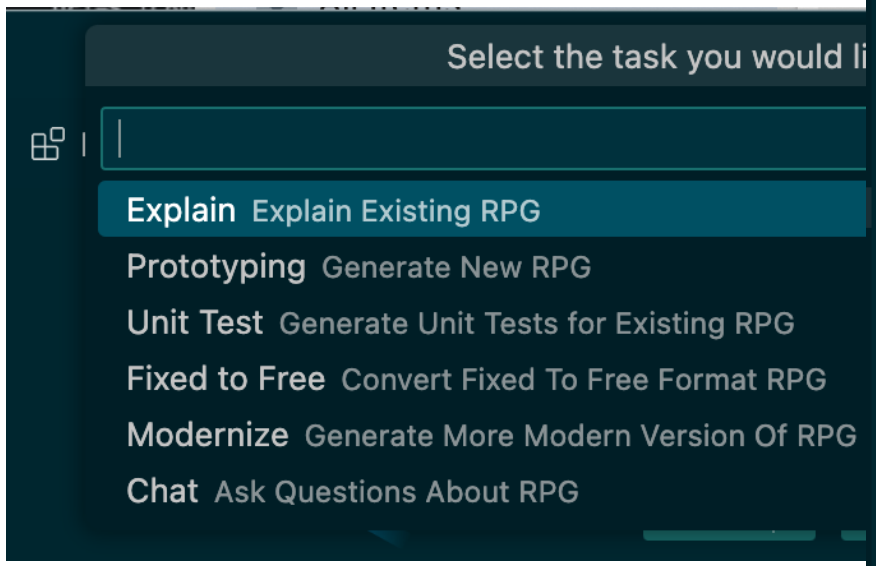
VS Code extension to automate test data contribution

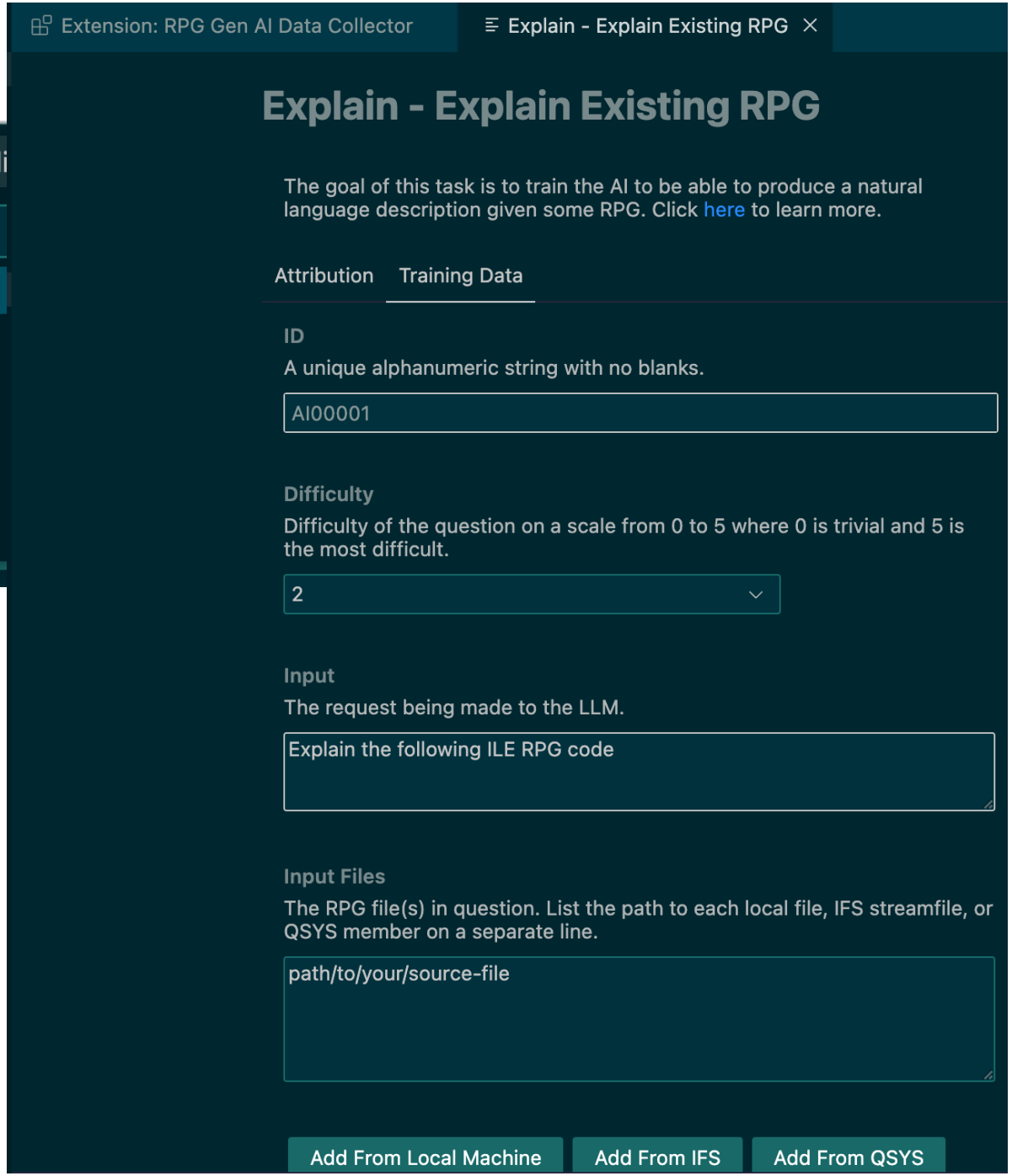
IBM i
continuous innovation
continuous integration



© Copyright IBM Corporation 2024

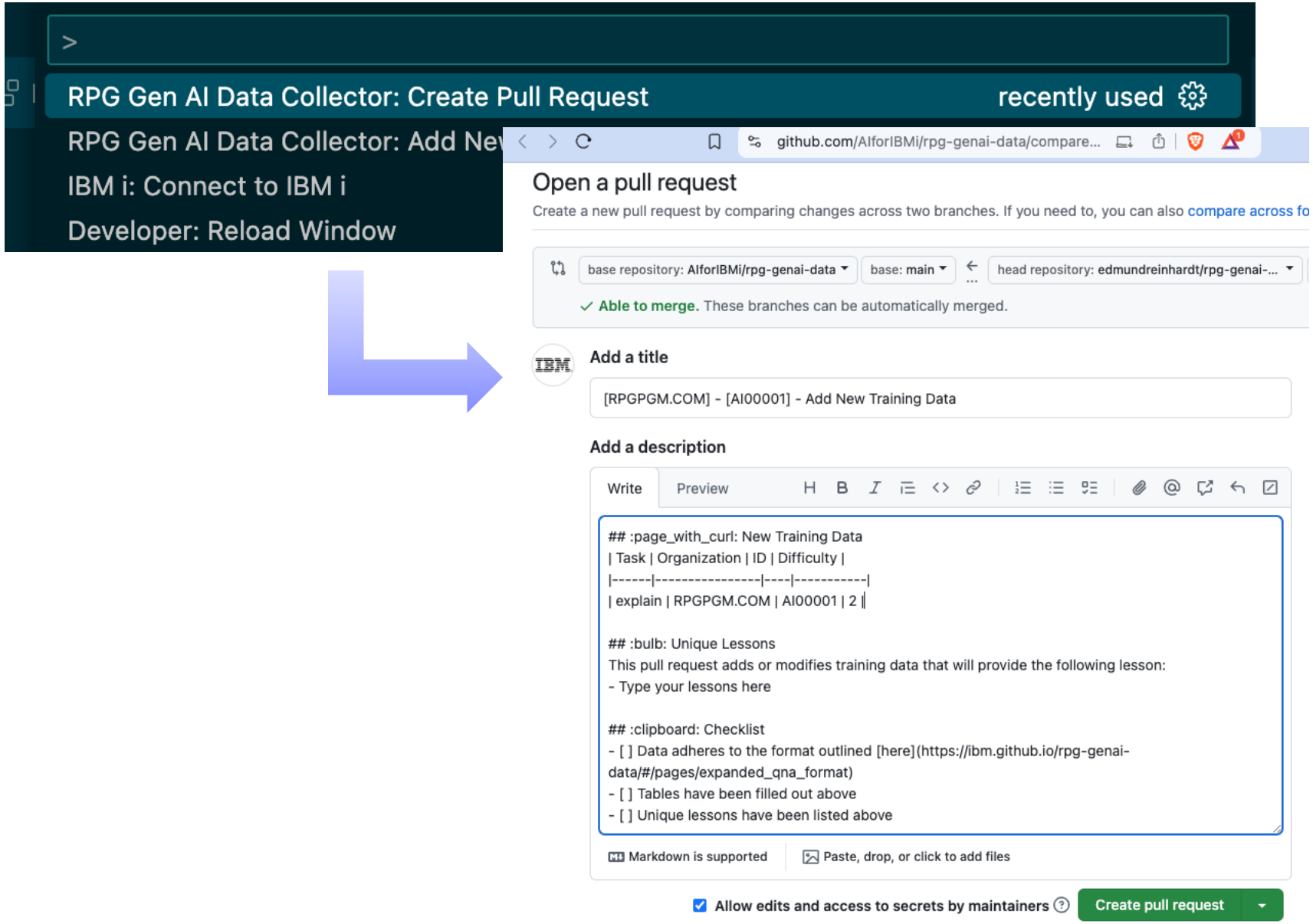
Fill out form to add data





© Copyright IBM Corporation 2024

One more command to send it to IBM repository



© Copyright IBM Corporation 2024

Questions

© Copyright IBM Corporation 2024

Links

IBM i
continuous innovation
continuous integration

- In depth positioning from Steve Will
- Instructions on how to contribute data

© Copyright IBM Corporation 2024

Would you like to get involved?

Join us in creating the new RPG code assistant for IBM i.

Engage with our team across various research activities, including but not limited to discussions and workshops to help define the offering, alpha & beta testing of the offering pre-GA, and submitting source code to help train and/or test the RPG code assistant model.

Your engagement helps shape the future of IBM Power.

URL:

https://ibm.biz/wca4i-interest-survey

QR code:

IBM i

continuous innovation
continuous integration