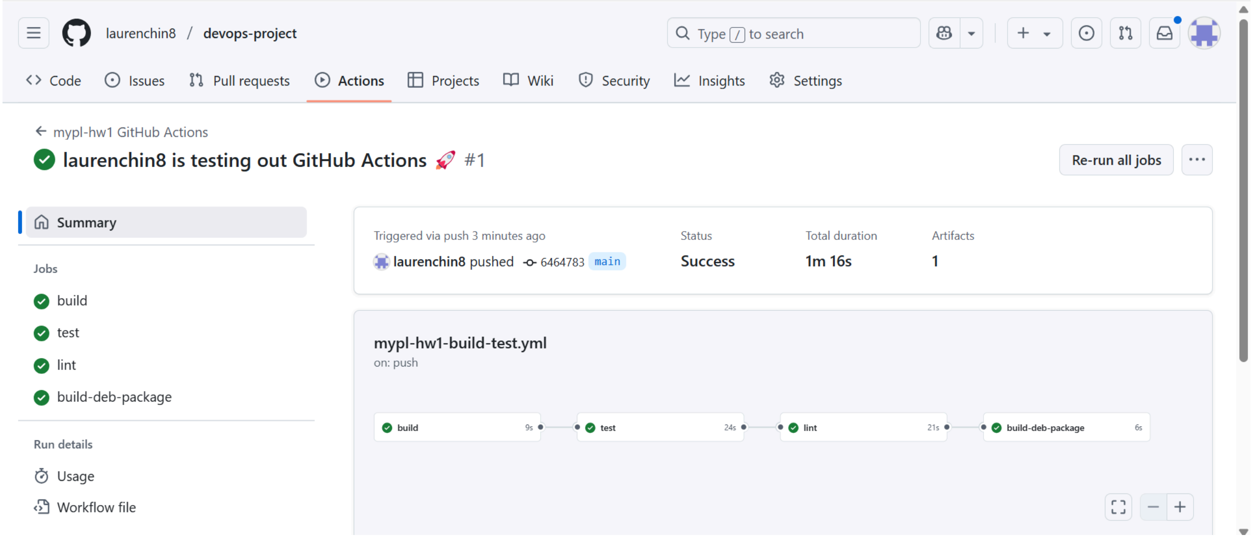
Lauren Chin
CPSC 334 Linux and DevOps

Final Project: Converting an Old Project to DevOps

Screenshots

CI/CD pipeline



laurenchin8 / devops-project

laurenchin8 is testing out GitHub Actions 🚀 #1

Triggered via push 3 minutes ago	Status	Total duration	Artifacts
laurenchin8 pushed 6464783 main	Success	1m 16s	1

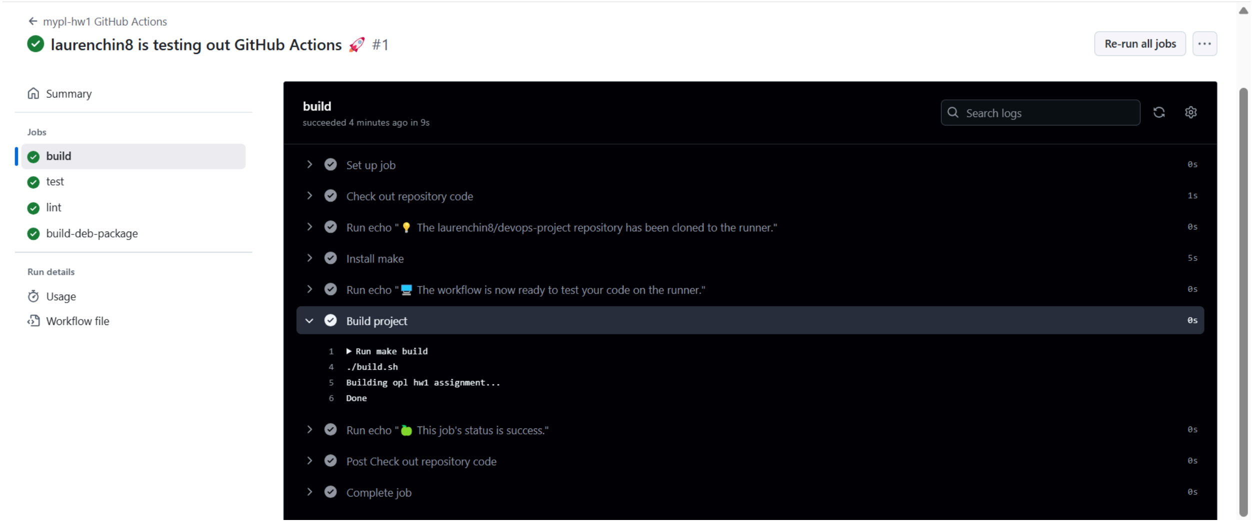
Jobs: ✓ build, ✓ test, ✓ lint, ✓ build-deb-package

mypl-hw1-build-test.yml
on: push

✓ build 9s → ✓ test 24s → ✓ lint 21s → ✓ build-deb-package 6s

Build, test, lint stages

Build stage

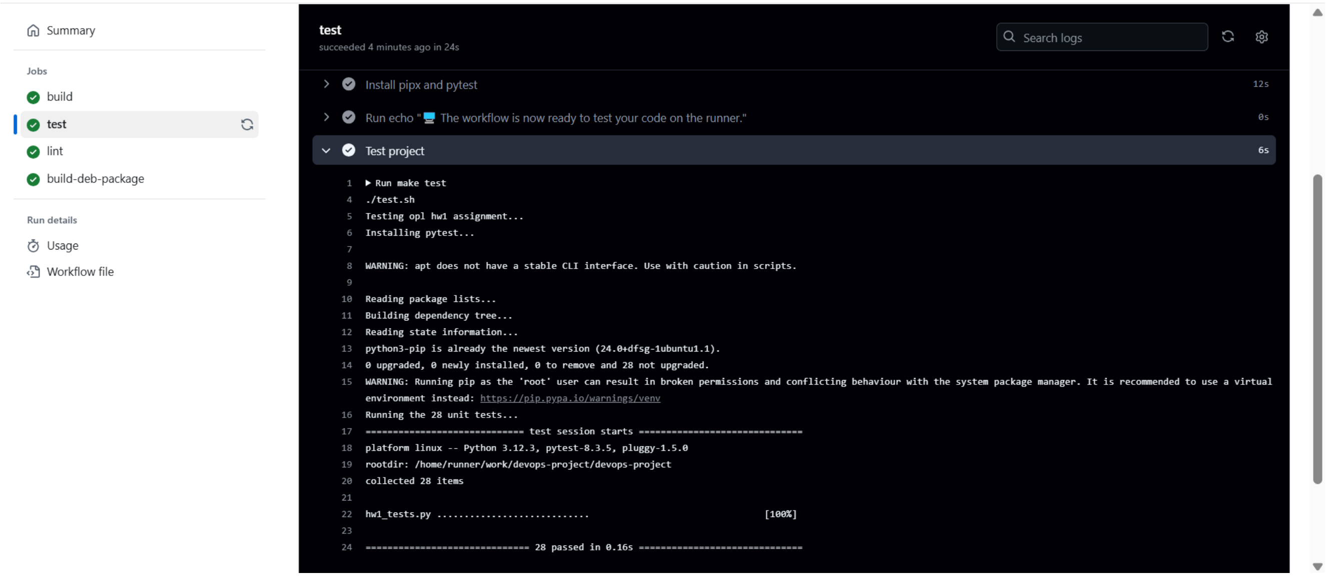


laurenchin8 is testing out GitHub Actions 🚀 #1

build
succeeded 4 minutes ago in 9s

```
> ✓ Set up job                                                                          0s
> ✓ Check out repository code                                                           1s
> ✓ Run echo "💡 The laurenchin8/devops-project repository has been cloned to the runner."  0s
> ✓ Install make                                                                        5s
> ✓ Run echo "🖥️ The workflow is now ready to test your code on the runner."            0s
v ✓ Build project                                                                       0s
      ▶ Run make build
      ./build.sh
      Building opl hw1 assignment...
      Done
> ✓ Run echo "🍏 This job's status is success."                                         0s
> ✓ Post Check out repository code                                                      0s
> ✓ Complete job                                                                        0s
```

Test stage



test

succeeded 4 minutes ago in 24s

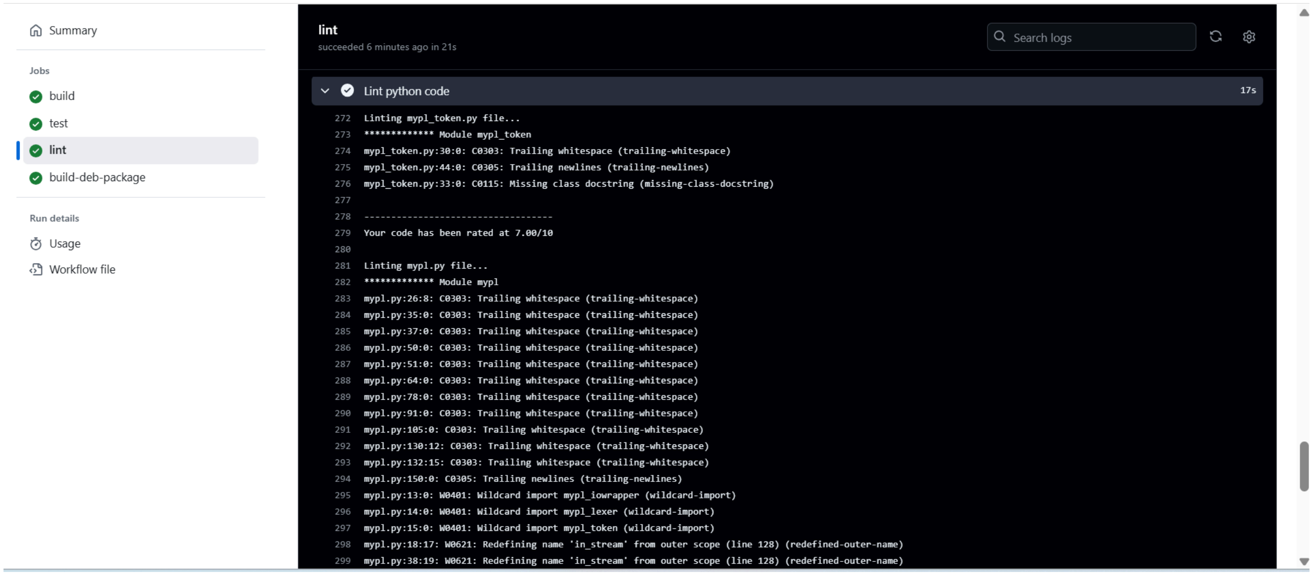
Install pipx and pytest 12s

Run echo "💻 The workflow is now ready to test your code on the runner." 0s

Test project 6s

```
▶ Run make test
./test.sh
Testing opl hw1 assignment...
Installing pytest...

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

Reading package lists...
Building dependency tree...
Reading state information...
python3-pip is already the newest version (24.0+dfsg-1ubuntu1.1).
0 upgraded, 0 newly installed, 0 to remove and 28 not upgraded.
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual
environment instead: https://pip.pypa.io/warnings/venv
Running the 28 unit tests...
============================= test session starts ==============================
platform linux -- Python 3.12.3, pytest-8.3.5, pluggy-1.5.0
rootdir: /home/runner/work/devops-project/devops-project
collected 28 items

hw1_tests.py ............................                                [100%]

============================== 28 passed in 0.16s ==============================
```

Lint stage



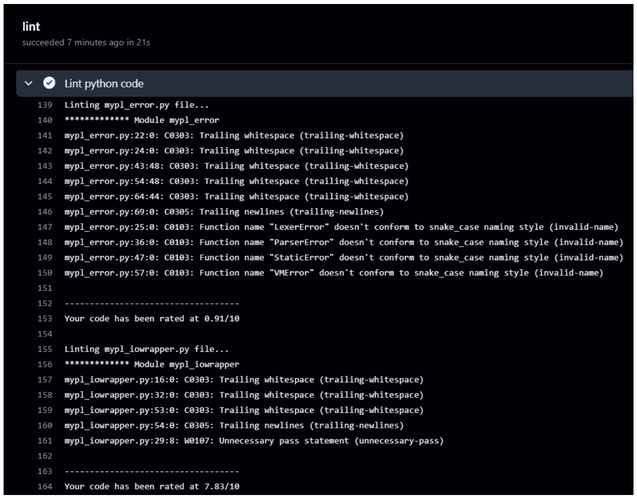
lint

succeeded 6 minutes ago in 21s

Lint python code 17s

```
Linting mypl_token.py file...
************* Module mypl_token
mypl_token.py:30:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_token.py:44:0: C0305: Trailing newlines (trailing-newlines)
mypl_token.py:33:0: C0115: Missing class docstring (missing-class-docstring)

-----------------------------------
Your code has been rated at 7.00/10

Linting mypl.py file...
************* Module mypl
mypl.py:26:8: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:35:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:37:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:50:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:51:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:64:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:78:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:91:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:105:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:130:12: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:132:15: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:150:0: C0305: Trailing newlines (trailing-newlines)
mypl.py:13:0: W0401: Wildcard import mypl_iowrapper (wildcard-import)
mypl.py:14:0: W0401: Wildcard import mypl_lexer (wildcard-import)
mypl.py:15:0: W0401: Wildcard import mypl_token (wildcard-import)
mypl.py:18:17: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
mypl.py:38:19: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
```

Linter run (lint output of linting all the mypl hw1 python files)

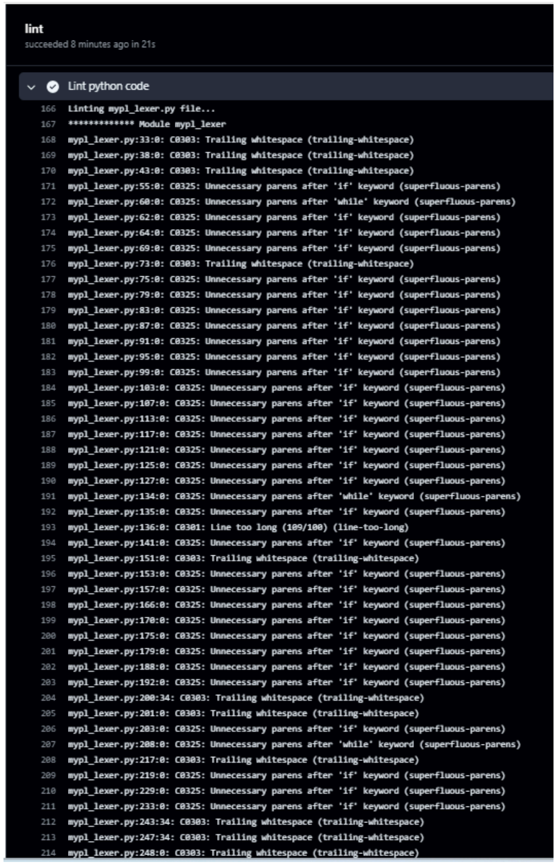


lint

succeeded 7 minutes ago in 21s

Lint python code

```
Linting mypl_error.py file...
************* Module mypl_error
mypl_error.py:22:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_error.py:24:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_error.py:43:48: C0303: Trailing whitespace (trailing-whitespace)
mypl_error.py:54:48: C0303: Trailing whitespace (trailing-whitespace)
mypl_error.py:64:44: C0303: Trailing whitespace (trailing-whitespace)
mypl_error.py:69:0: C0305: Trailing newlines (trailing-newlines)
mypl_error.py:25:0: C0103: Function name "LexerError" doesn't conform to snake_case naming style (invalid-name)
mypl_error.py:36:0: C0103: Function name "ParserError" doesn't conform to snake_case naming style (invalid-name)
mypl_error.py:47:0: C0103: Function name "StaticError" doesn't conform to snake_case naming style (invalid-name)
mypl_error.py:57:0: C0103: Function name "VMError" doesn't conform to snake_case naming style (invalid-name)

-----------------------------------
Your code has been rated at 0.91/10

Linting mypl_iowrapper.py file...
************* Module mypl_iowrapper
mypl_iowrapper.py:16:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_iowrapper.py:32:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_iowrapper.py:53:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_iowrapper.py:54:0: C0305: Trailing newlines (trailing-newlines)
mypl_iowrapper.py:29:8: W0107: Unnecessary pass statement (unnecessary-pass)

-----------------------------------
Your code has been rated at 7.83/10
```

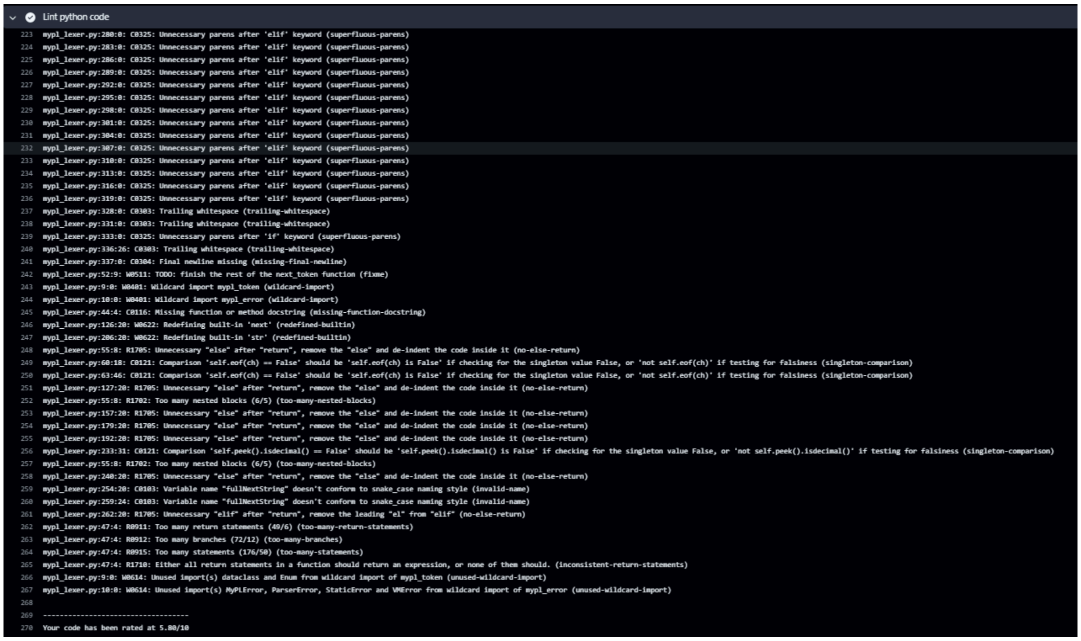


lint

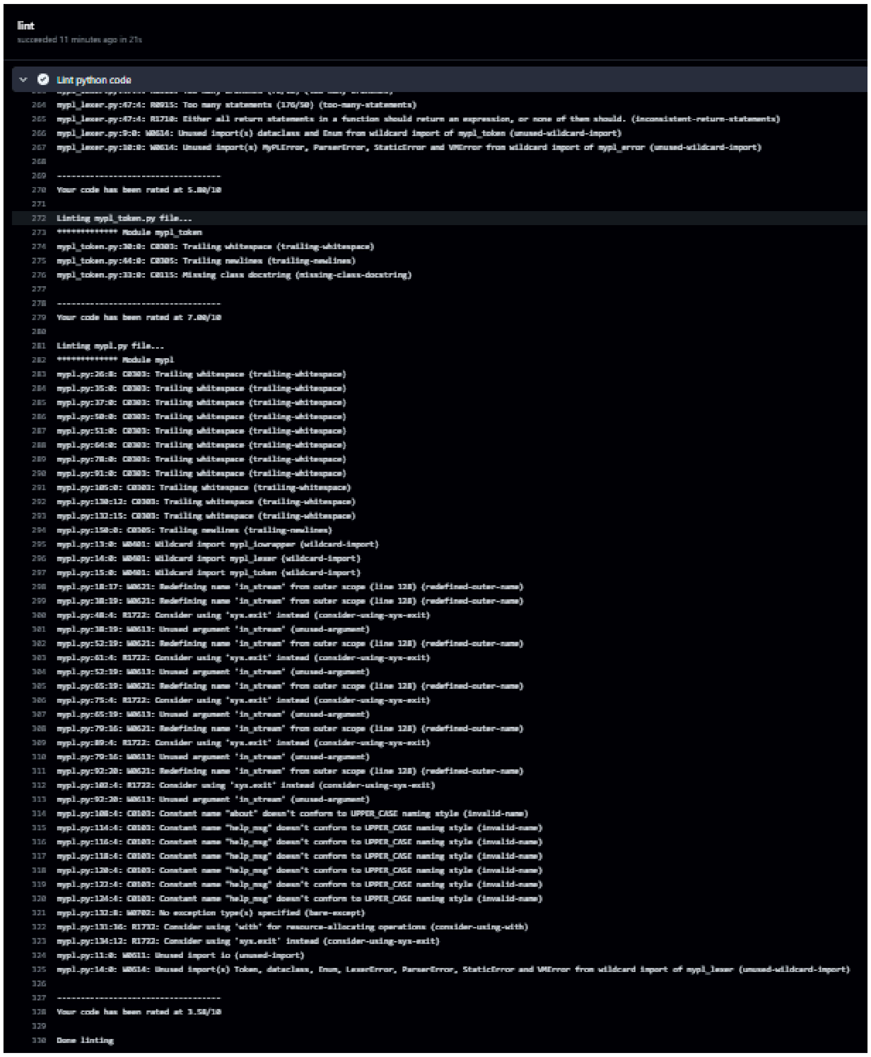
succeeded 8 minutes ago in 21s

Lint python code

```
Linting mypl_lexer.py file...
************* Module mypl_lexer
mypl_lexer.py:33:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:38:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:43:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:55:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:60:0: C0325: Unnecessary parens after 'while' keyword (superfluous-parens)
mypl_lexer.py:62:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:64:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:69:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:73:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:75:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:79:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:83:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:87:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:91:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:95:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:99:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:103:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:107:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:113:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:117:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:121:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:125:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:127:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:134:0: C0325: Unnecessary parens after 'while' keyword (superfluous-parens)
mypl_lexer.py:135:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:136:0: C0301: Line too long (109/100) (line-too-long)
mypl_lexer.py:141:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:151:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:153:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:157:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:166:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:170:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:175:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:179:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:188:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:192:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:200:34: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:201:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:203:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:208:0: C0325: Unnecessary parens after 'while' keyword (superfluous-parens)
mypl_lexer.py:217:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:219:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:229:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:233:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:243:34: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:247:34: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:248:0: C0303: Trailing whitespace (trailing-whitespace)
```

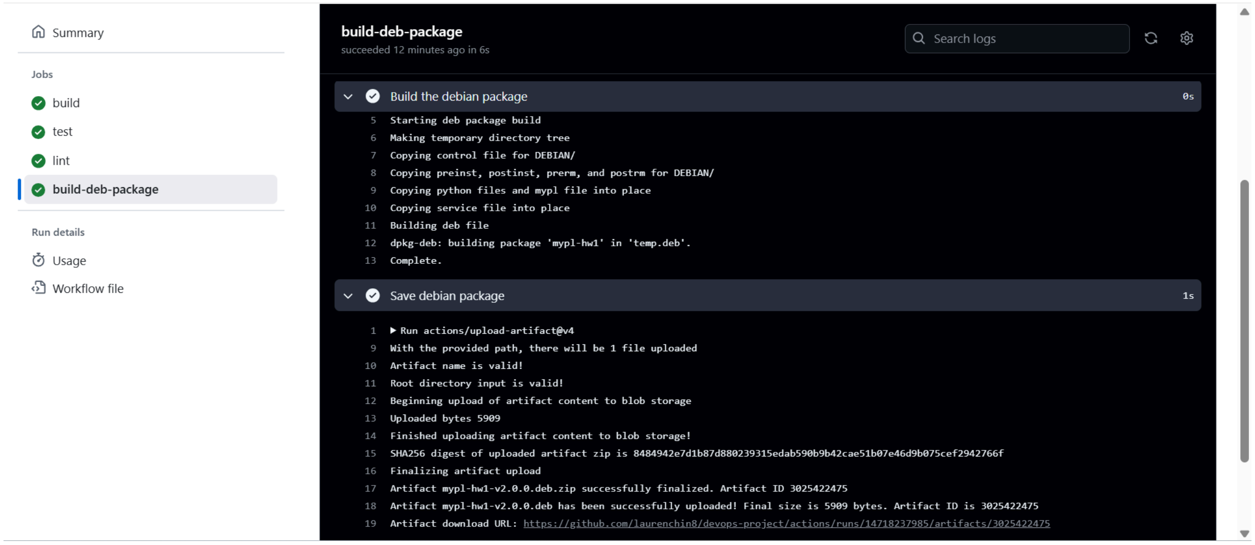


```
mypl_lexer.py:280:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:283:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:286:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:289:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:292:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:295:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:298:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:301:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:304:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:307:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:310:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:313:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:316:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:319:0: C0325: Unnecessary parens after 'elif' keyword (superfluous-parens)
mypl_lexer.py:328:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:331:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:333:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
mypl_lexer.py:336:26: C0303: Trailing whitespace (trailing-whitespace)
mypl_lexer.py:337:0: C0304: Final newline missing (missing-final-newline)
mypl_lexer.py:52:9: W0511: TODO: finish the rest of the next_token function (fixme)
mypl_lexer.py:9:0: W0401: Wildcard import mypl_token (wildcard-import)
mypl_lexer.py:10:0: W0401: Wildcard import mypl_error (wildcard-import)
mypl_lexer.py:44:4: C0116: Missing function or method docstring (missing-function-docstring)
mypl_lexer.py:126:20: W0622: Redefining built-in 'next' (redefined-builtin)
mypl_lexer.py:206:20: W0622: Redefining built-in 'str' (redefined-builtin)
mypl_lexer.py:55:8: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)
mypl_lexer.py:60:18: C0121: Comparison 'self.eof(ch) == False' should be 'self.eof(ch) is False' if checking for the singleton value False, or 'not self.eof(ch)' if testing for falsiness (singleton-comparison)
mypl_lexer.py:63:46: C0121: Comparison 'self.eof(ch) == False' should be 'self.eof(ch) is False' if checking for the singleton value False, or 'not self.eof(ch)' if testing for falsiness (singleton-comparison)
mypl_lexer.py:127:20: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)
mypl_lexer.py:55:8: R1702: Too many nested blocks (6/5) (too-many-nested-blocks)
mypl_lexer.py:157:20: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)
mypl_lexer.py:179:20: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)
mypl_lexer.py:192:20: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)
mypl_lexer.py:233:31: C0121: Comparison 'self.peek().isdecimal() == False' should be 'self.peek().isdecimal() is False' if checking for the singleton value False, or 'not self.peek().isdecimal()' if testing for falsiness (singleton-comparison)
mypl_lexer.py:55:8: R1702: Too many nested blocks (6/5) (too-many-nested-blocks)
mypl_lexer.py:240:20: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)
mypl_lexer.py:254:20: C0103: Variable name "fullNextString" doesn't conform to snake_case naming style (invalid-name)
mypl_lexer.py:259:24: C0103: Variable name "fullNextString" doesn't conform to snake_case naming style (invalid-name)
mypl_lexer.py:262:20: R1705: Unnecessary "elif" after "return", remove the leading "el" from "elif" (no-else-return)
mypl_lexer.py:47:4: R0911: Too many return statements (49/6) (too-many-return-statements)
mypl_lexer.py:47:4: R0912: Too many branches (72/12) (too-many-branches)
mypl_lexer.py:47:4: R0915: Too many statements (176/50) (too-many-statements)
mypl_lexer.py:47:4: R1710: Either all return statements in a function should return an expression, or none of them should. (inconsistent-return-statements)
mypl_lexer.py:9:0: W0614: Unused import(s) dataclass and Enum from wildcard import of mypl_token (unused-wildcard-import)
mypl_lexer.py:10:0: W0614: Unused import(s) MyPLError, ParserError, StaticError and VMError from wildcard import of mypl_error (unused-wildcard-import)

-----------------------------------
Your code has been rated at 5.80/10
```



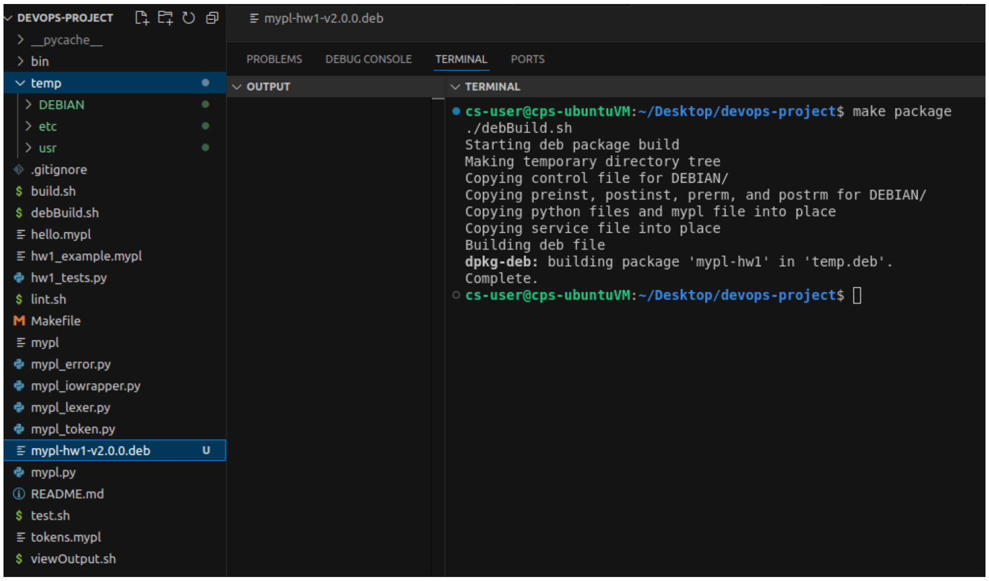
```
mypl_lexer.py:47:4: R0915: Too many statements (176/50) (too-many-statements)
mypl_lexer.py:47:4: R1710: Either all return statements in a function should return an expression, or none of them should. (inconsistent-return-statements)
mypl_lexer.py:9:0: W0614: Unused import(s) dataclass and Enum from wildcard import of mypl_token (unused-wildcard-import)
mypl_lexer.py:10:0: W0614: Unused import(s) MyPLError, ParserError, StaticError and VMError from wildcard import of mypl_error (unused-wildcard-import)

-----------------------------------
Your code has been rated at 5.80/10

Linting mypl_token.py file...
************* Module mypl_token
mypl_token.py:30:0: C0303: Trailing whitespace (trailing-whitespace)
mypl_token.py:44:0: C0305: Trailing newlines (trailing-newlines)
mypl_token.py:33:0: C0115: Missing class docstring (missing-class-docstring)

-----------------------------------
Your code has been rated at 7.00/10

Linting mypl.py file...
************* Module mypl
mypl.py:26:8: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:35:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:37:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:50:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:51:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:64:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:78:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:91:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:105:0: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:130:12: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:132:15: C0303: Trailing whitespace (trailing-whitespace)
mypl.py:150:0: C0305: Trailing newlines (trailing-newlines)
mypl.py:13:0: W0401: Wildcard import mypl_iowrapper (wildcard-import)
mypl.py:14:0: W0401: Wildcard import mypl_lexer (wildcard-import)
mypl.py:15:0: W0401: Wildcard import mypl_token (wildcard-import)
mypl.py:18:17: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
mypl.py:38:19: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
mypl.py:48:4: R1722: Consider using 'sys.exit' instead (consider-using-sys-exit)
mypl.py:38:19: W0613: Unused argument 'in_stream' (unused-argument)
mypl.py:52:19: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
mypl.py:61:4: R1722: Consider using 'sys.exit' instead (consider-using-sys-exit)
mypl.py:52:19: W0613: Unused argument 'in_stream' (unused-argument)
mypl.py:65:19: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
mypl.py:75:4: R1722: Consider using 'sys.exit' instead (consider-using-sys-exit)
mypl.py:65:19: W0613: Unused argument 'in_stream' (unused-argument)
mypl.py:79:16: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
mypl.py:89:4: R1722: Consider using 'sys.exit' instead (consider-using-sys-exit)
mypl.py:79:16: W0613: Unused argument 'in_stream' (unused-argument)
mypl.py:92:20: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
mypl.py:102:4: R1722: Consider using 'sys.exit' instead (consider-using-sys-exit)
mypl.py:92:20: W0613: Unused argument 'in_stream' (unused-argument)
mypl.py:108:4: C0103: Constant name "about" doesn't conform to UPPER_CASE naming style (invalid-name)
mypl.py:114:4: C0103: Constant name "help_msg" doesn't conform to UPPER_CASE naming style (invalid-name)
mypl.py:116:4: C0103: Constant name "help_msg" doesn't conform to UPPER_CASE naming style (invalid-name)
mypl.py:118:4: C0103: Constant name "help_msg" doesn't conform to UPPER_CASE naming style (invalid-name)
mypl.py:120:4: C0103: Constant name "help_msg" doesn't conform to UPPER_CASE naming style (invalid-name)
mypl.py:122:4: C0103: Constant name "help_msg" doesn't conform to UPPER_CASE naming style (invalid-name)
mypl.py:124:4: C0103: Constant name "help_msg" doesn't conform to UPPER_CASE naming style (invalid-name)
mypl.py:132:8: W0702: No exception type(s) specified (bare-except)
mypl.py:131:36: R1732: Consider using 'with' for resource-allocating operations (consider-using-with)
mypl.py:134:12: R1722: Consider using 'sys.exit' instead (consider-using-sys-exit)
mypl.py:11:0: W0611: Unused import io (unused-import)
mypl.py:14:0: W0614: Unused import(s) Token, dataclass, Enum, LexerError, ParserError, StaticError and VMError from wildcard import of mypl_lexer (unused-wildcard-import)

-----------------------------------
Your code has been rated at 3.58/10

Done linting
```

Debian package screenshots

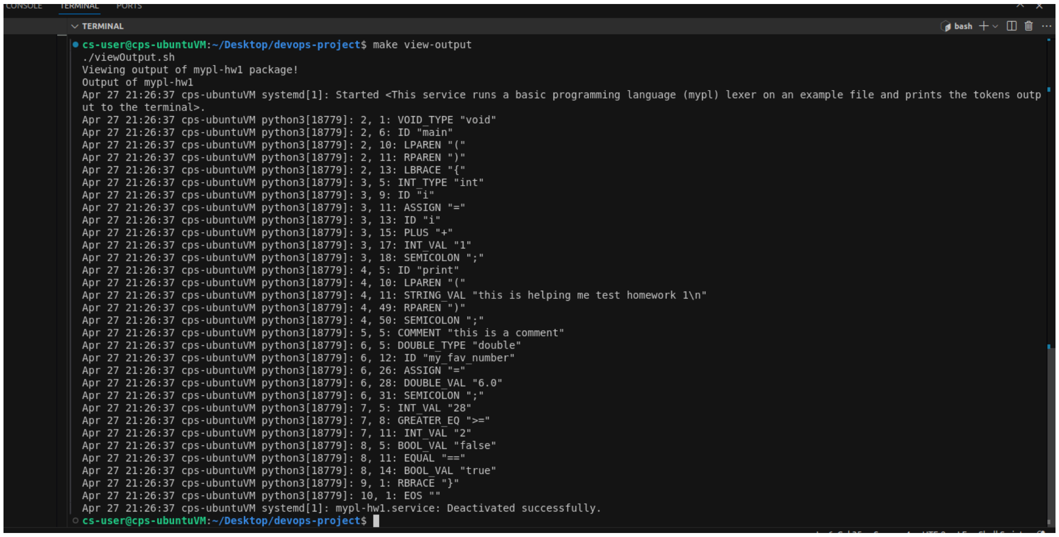


```
Starting deb package build
Making temporary directory tree
Copying control file for DEBIAN/
Copying preinst, postinst, prerm, and postrm for DEBIAN/
Copying python files and mypl file into place
Copying service file into place
Building deb file
dpkg-deb: building package 'mypl-hw1' in 'temp.deb'.
Complete.
```

```
▶ Run actions/upload-artifact@v4
With the provided path, there will be 1 file uploaded
Artifact name is valid!
Root directory input is valid!
Beginning upload of artifact content to blob storage
Uploaded bytes 5909
Finished uploading artifact content to blob storage!
SHA256 digest of uploaded artifact zip is 8484942e7d1b87d880239315edab590b9b42cae51b07e46d9b075cef2942766f
Finalizing artifact upload
Artifact mypl-hw1-v2.0.0.deb.zip successfully finalized. Artifact ID 3025422475
Artifact mypl-hw1-v2.0.0.deb has been successfully uploaded! Final size is 5909 bytes. Artifact ID is 3025422475
Artifact download URL: https://github.com/laurenchin8/devops-project/actions/runs/14718237985/artifacts/3025422475
```



```
cs-user@cps-ubuntuVM:~/Desktop/devops-project$ make package
./debBuild.sh
Starting deb package build
Making temporary directory tree
Copying control file for DEBIAN/
Copying preinst, postinst, prerm, and postrm for DEBIAN/
Copying python files and mypl file into place
Copying service file into place
Building deb file
dpkg-deb: building package 'mypl-hw1' in 'temp.deb'.
Complete.
cs-user@cps-ubuntuVM:~/Desktop/devops-project$
```



```
cs-user@cps-ubuntuVM:~/Desktop/devops-project$ make view-output
./viewOutput.sh
Viewing output of mypl-hw1 package!
Output of mypl-hw1
Apr 27 21:26:37 cps-ubuntuVM systemd[1]: Started <This service runs a basic programming language (mypl) lexer on an example file and prints the tokens outp
ut to the terminal>.
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 2, 1: VOID_TYPE "void"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 2, 6: ID "main"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 2, 10: LPAREN "("
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 2, 11: RPAREN ")"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 2, 13: LBRACE "{"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 3, 5: INT_TYPE "int"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 3, 9: ID "i"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 3, 11: ASSIGN "="
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 3, 13: ID "i"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 3, 15: PLUS "+"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 3, 17: INT_VAL "1"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 3, 18: SEMICOLON ";"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 4, 5: ID "print"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 4, 10: LPAREN "("
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 4, 11: STRING_VAL "this is helping me test homework 1\n"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 4, 49: RPAREN ")"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 4, 50: SEMICOLON ";"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 5, 5: COMMENT "this is a comment"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 6, 5: DOUBLE_TYPE "double"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 6, 12: ID "my_fav_number"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 6, 26: ASSIGN "="
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 6, 28: DOUBLE_VAL "6.0"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 6, 31: SEMICOLON ";"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 7, 5: INT_VAL "28"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 7, 8: GREATER_EQ ">="
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 7, 11: INT_VAL "2"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 8, 5: BOOL_VAL "false"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 8, 11: EQUAL "=="
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 8, 14: BOOL_VAL "true"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 9, 1: RBRACE "}"
Apr 27 21:26:37 cps-ubuntuVM python3[18779]: 10, 1: EOS ""
Apr 27 21:26:37 cps-ubuntuVM systemd[1]: mypl-hw1.service: Deactivated successfully.
cs-user@cps-ubuntuVM:~/Desktop/devops-project$
```

Post-Effort Reflection Write-Up

Techniques and Tools Used

For this final project, I used python’s pytest and pylint packages to help me with the “test” and “lint” actions for the CI/CD pipeline. I used VSCode as my IDE and I used GitHub to implement this project, just like we did throughout the course. I used a Makefile with targets for each of the “build”, “test”, “lint”, and “package” tasks and then I used shell scripts that I ran for each target to keep the Makefile clean.

Key Steps Taken to Build the Package

First I created the Makefile with targets for “build”, “test”, “lint”, and “package”.
Then, I created the shell script files for each “build”, “test”, and “lint” which were all fairly simple.
Then, I had to create and refine the debBuild.sh file and all of the necessary package files like the control file and the preinst, postinst, prerm, and postrm files.
Then, I could run the “make package” command to create the mypl-hw1-v2.0.0.deb file.
Then, I created a viewOutput.sh file and a “view-ouput” makefile target to be able to see the terminal output of the program!

Challenges Encountered

A large challenge that I encountered was just fitting the already finished project/assignment to include the DevOps methodologies. It would have been much easier to incorporate this while working on the assignment. Specifically, the deb-packaging portion took me a while and got quite complicated with making sure I got all the files copied where they need to be, running the correct commands, and figuring out how to show the output because this project was configured to output the results to the terminal.

Evaluation of DevOps Methodologies

In the future, using DevOps Methodologies such as container usage and packaging can be very useful skills and tools for developers to use more consistently. If this becomes more of a standard for software developers, we will have less of the disconnect between developers and operations employees. While it takes more effort for developers, in the long run it could actually save them time that they spend having to help ops debug the code they delivered. If developers can use DevOps tools like containerization and packaging, they can create that consistent environment so their code runs the same each time no matter the device, so ops won’t have to hear the argument “it ran on my computer” again. Through this final project of converting an old project to DevOps, I also learned that it’s much easier to implement these DevOps tools and methodologies while you are working on the project as opposed to afterwards, so if developers can use some of these DevOps methodologies while they are developing their product, I think this would prove very helpful.

Reflection on Linux and DevOps Course

This Linux and DevOps course has given me a new perspective on software development because it reminded me of the importance of the connection between the developers and the deployment/operations team. We made the joke numerous times throughout the semester that software developers pass along their code and say that “it is ops problem now”. When the ops team tries to compile and run the code and it breaks, developers say that it worked on their machines and this creates a problem. This course helped remind me of this disconnection and the importance of this integration of DevOps so that we don’t come across this problem in the workplace. This course taught me the importance of tools like GitHub Actions and how important it is as developers to have the operations team in mind when you are developing, because it is very important that your code works in their environment too not just their own!