

Final Project: Converting an Old Project to DevOps

Screenshots

CI/CD pipeline

This screenshot shows the GitHub Actions interface for the 'mypl-hw1' workflow. The workflow is titled 'laurencin8 is testing out GitHub Actions' and is in a 'Success' state. It was triggered by a push to the 'main' branch. The total duration of the workflow is 1m 16s, and it has produced 1 artifact. The workflow consists of four jobs: 'build', 'test', 'lint', and 'build-deb-package'. The 'Summary' tab is selected, showing a list of jobs and their status. The 'Run details' tab is also visible, showing usage and workflow file information.

laurencin8 / devops-project

mypl-hw1 GitHub Actions

laurencin8 is testing out GitHub Actions #1

Re-run all jobs

Summary

Jobs

- build
- test
- lint
- build-deb-package

Run details

- Usage
- Workflow file

Triggered via push 3 minutes ago

laurencin8 pushed 6464783 main

Status: Success

Total duration: 1m 16s

Artifacts: 1

mypl-hw1-build-test.yml

on: push

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

Build, test, lint stages

Build stage

This screenshot shows the details of the 'build' job in the 'mypl-hw1' workflow. The job is in a 'Success' state and was completed 4 minutes ago. The job log shows the following steps:

- Set up job (0s)
- Check out repository code (1s)
- Run echo "The laurencin8/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)
- 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

Summary

Jobs

build

test

lint

build-deb-package

Run details

Usage

Workflow file

test

succeeded 4 minutes ago in 24s

Search logs

12s

0s

6s

1 Run make test

4 ./test.sh

5 Testing opl hwi assignment...

6 Installing pytest...

7

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

9

10 Reading package lists...

11 Building dependency tree...

12 Reading state information...

13 python3-pip is already the newest version (24.0dfsg-1ubuntu1.1).

14 0 upgraded, 0 newly installed, 0 to remove and 28 not upgraded.

15 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>

16 Running the 28 unit tests...

17 ===== test session starts =====

18 platform linux -- Python 3.12.3, pytest-8.3.5, pluggy-1.5.0

19 rootdir: /home/runner/work/devops-project/devops-project

20 collected 28 items

21

22 hwi_tests.py [100%]

23

24 ===== 28 passed in 0.16s =====

Lint stage

Summary

Jobs

build

test

lint

build-deb-package

Run details

Usage

Workflow file

lint

succeeded 6 minutes ago in 21s

Search logs

17s

Lint python code

272 Linting mypl_token.py file...

273 ***** Module mypl_token

274 mypl_token.py:30:0: C0303: Trailing whitespace (trailing-whitespace)

275 mypl_token.py:44:0: C0305: Trailing newlines (trailing-newlines)

276 mypl_token.py:33:0: C0115: Missing class docstring (missing-class-docstring)

277

278 -----

279 Your code has been rated at 7.00/10

280

281 Linting mypl.py file...

282 ***** Module mypl

283 mypl.py:26:8: C0303: Trailing whitespace (trailing-whitespace)

284 mypl.py:35:0: C0303: Trailing whitespace (trailing-whitespace)

285 mypl.py:37:0: C0303: Trailing whitespace (trailing-whitespace)

286 mypl.py:50:0: C0303: Trailing whitespace (trailing-whitespace)

287 mypl.py:51:0: C0303: Trailing whitespace (trailing-whitespace)

288 mypl.py:64:0: C0303: Trailing whitespace (trailing-whitespace)

289 mypl.py:78:0: C0303: Trailing whitespace (trailing-whitespace)

290 mypl.py:91:0: C0303: Trailing whitespace (trailing-whitespace)

291 mypl.py:105:0: C0303: Trailing whitespace (trailing-whitespace)

292 mypl.py:110:12: C0303: Trailing whitespace (trailing-whitespace)

293 mypl.py:112:15: C0303: Trailing whitespace (trailing-whitespace)

294 mypl.py:150:0: C0305: trailing newlines (trailing-newlines)

295 mypl.py:11:0: W0401: Wildcard import mypl_lowerupper (wildcard-import)

296 mypl.py:14:0: W0401: Wildcard import mypl_lexer (wildcard-import)

297 mypl.py:15:0: W0401: Wildcard import mypl_token (wildcard-import)

298 mypl.py:19:17: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)

299 mypl.py:38:19: W0621: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)

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

```
lint
succeeded 7 minutes ago in 21s

▼ ⬢ Lint python code

139 Linting mypl_error.py file...
140 ***** Module mypl_error
141 mypl_error.py:22:0: C0303: Trailing whitespace (trailing-whitespace)
142 mypl_error.py:24:0: C0303: Trailing whitespace (trailing-whitespace)
143 mypl_error.py:43:48: C0303: Trailing whitespace (trailing-whitespace)
144 mypl_error.py:54:48: C0303: Trailing whitespace (trailing-whitespace)
145 mypl_error.py:64:44: C0303: Trailing whitespace (trailing-whitespace)
146 mypl_error.py:69:0: C0305: Trailing newlines (trailing-newlines)
147 mypl_error.py:25:0: C0103: Function name "lexerError" doesn't conform to snake_case naming style (invalid-name)
148 mypl_error.py:36:0: C0103: Function name "ParserError" doesn't conform to snake_case naming style (invalid-name)
149 mypl_error.py:47:0: C0103: Function name "StaticError" doesn't conform to snake_case naming style (invalid-name)
150 mypl_error.py:57:0: C0103: Function name "WError" doesn't conform to snake_case naming style (invalid-name)
151
152 -----
153 Your code has been rated at 0.91/10
154
155 Linting mypl_lowerupper.py file...
156 ***** Module mypl_lowerupper
157 mypl_lowerupper.py:16:0: C0303: Trailing whitespace (trailing-whitespace)
158 mypl_lowerupper.py:32:0: C0303: Trailing whitespace (trailing-whitespace)
159 mypl_lowerupper.py:51:0: C0303: Trailing whitespace (trailing-whitespace)
160 mypl_lowerupper.py:54:0: C0303: Trailing newlines (trailing-newlines)
161 mypl_lowerupper.py:29:8: W0107: Unnecessary pass statement (unnecessary-pass)
162
163 -----
164 Your code has been rated at 7.83/10
```

```
lint
succeeded 8 minutes ago in 21s

▼ ⬢ Lint python code

166 Linting mypl_lexer.py file...
167 ***** Module mypl_lexer
168 mypl_lexer.py:33:0: C0303: Trailing whitespace (trailing-whitespace)
169 mypl_lexer.py:38:0: C0303: Trailing whitespace (trailing-whitespace)
170 mypl_lexer.py:43:0: C0303: Trailing whitespace (trailing-whitespace)
171 mypl_lexer.py:55:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
172 mypl_lexer.py:60:8: C0325: Unnecessary parens after 'while' keyword (superfluous-parens)
173 mypl_lexer.py:62:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
174 mypl_lexer.py:64:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
175 mypl_lexer.py:69:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
176 mypl_lexer.py:73:0: C0303: Trailing whitespace (trailing-whitespace)
177 mypl_lexer.py:75:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
178 mypl_lexer.py:79:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
179 mypl_lexer.py:83:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
180 mypl_lexer.py:87:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
181 mypl_lexer.py:91:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
182 mypl_lexer.py:95:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
183 mypl_lexer.py:99:8: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
184 mypl_lexer.py:103:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
185 mypl_lexer.py:107:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
186 mypl_lexer.py:113:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
187 mypl_lexer.py:117:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
188 mypl_lexer.py:121:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
189 mypl_lexer.py:125:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
190 mypl_lexer.py:127:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
191 mypl_lexer.py:134:0: C0325: Unnecessary parens after 'while' keyword (superfluous-parens)
192 mypl_lexer.py:135:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
193 mypl_lexer.py:136:0: C0301: Line too long (109/100) (line-too-long)
194 mypl_lexer.py:141:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
195 mypl_lexer.py:151:0: C0303: Trailing whitespace (trailing-whitespace)
196 mypl_lexer.py:153:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
197 mypl_lexer.py:157:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
198 mypl_lexer.py:166:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
199 mypl_lexer.py:170:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
200 mypl_lexer.py:175:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
201 mypl_lexer.py:179:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
202 mypl_lexer.py:188:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
203 mypl_lexer.py:192:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
204 mypl_lexer.py:200:34: C0303: Trailing whitespace (trailing-whitespace)
205 mypl_lexer.py:201:0: C0303: Trailing whitespace (trailing-whitespace)
206 mypl_lexer.py:203:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
207 mypl_lexer.py:208:0: C0325: Unnecessary parens after 'while' keyword (superfluous-parens)
208 mypl_lexer.py:217:0: C0303: Trailing whitespace (trailing-whitespace)
209 mypl_lexer.py:219:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
210 mypl_lexer.py:229:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
211 mypl_lexer.py:233:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
212 mypl_lexer.py:243:34: C0303: Trailing whitespace (trailing-whitespace)
213 mypl_lexer.py:247:34: C0303: Trailing whitespace (trailing-whitespace)
214 mypl_lexer.py:248:0: C0303: Trailing whitespace (trailing-whitespace)
```

```

1 lint.py:python code
223 mypl_lsear.py:280-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
224 mypl_lsear.py:281-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
225 mypl_lsear.py:286-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
226 mypl_lsear.py:289-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
227 mypl_lsear.py:292-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
228 mypl_lsear.py:295-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
229 mypl_lsear.py:298-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
230 mypl_lsear.py:301-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
231 mypl_lsear.py:304-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
232 mypl_lsear.py:307-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
233 mypl_lsear.py:318-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
234 mypl_lsear.py:319-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
235 mypl_lsear.py:316-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
236 mypl_lsear.py:319-8: CR125: Unnecessary parens after 'elif' keyword (superfluous-parens)
237 mypl_lsear.py:318-8: CR181: Trailing whitespace (trailing-whitespace)
238 mypl_lsear.py:318-8: CR181: Trailing whitespace (trailing-whitespace)
239 mypl_lsear.py:311-8: CR125: Unnecessary parens after 'if' keyword (superfluous-parens)
240 mypl_lsear.py:316-26: CR181: Trailing whitespace (trailing-whitespace)
241 mypl_lsear.py:337-8: CR184: Final newline missing (missing-final-newline)
242 mypl_lsear.py:5129: W0611: TODO: finish the rest of the next_token function (fixme)
243 mypl_lsear.py:9:4: W0601: wildcard import mypl_token (wildcard-import)
244 mypl_lsear.py:18:8: W0601: wildcard import mypl_error (wildcard-import)
245 mypl_lsear.py:44:4: CR116: Missing function or method docstring (missing-function-docstring)
246 mypl_lsear.py:126-26: W0602: Redefining built-in 'exec' (redefined-builtin)
247 mypl_lsear.py:26-26: W0602: Redefining built-in 'str' (redefined-builtin)
248 mypl_lsear.py:55-8: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)
249 mypl_lsear.py:68-18: CR121: Comparison 'self.conf(ch) == False' should be 'self.conf(ch) is False' if checking for the singleton value False, or 'not self.conf(ch)' if testing for falsiness (singleton-comparison)
250 mypl_lsear.py:81-48: CR121: Comparison 'self.conf(ch) == False' should be 'self.conf(ch) is False' if checking for the singleton value False, or 'not self.conf(ch)' if testing for falsiness (singleton-comparison)
251 mypl_lsear.py:172-28: R1705: Unnecessary "elif" after "return", remove the "elif" and de-indent the code inside it (no-else-return)
252 mypl_lsear.py:55-8: R1702: Too many nested blocks (6/5) (too-many-nested-blocks)
253 mypl_lsear.py:157-28: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)
254 mypl_lsear.py:176-28: R1705: Unnecessary "elif" after "return", remove the "elif" and de-indent the code inside it (no-else-return)
255 mypl_lsear.py:182-28: R1705: Unnecessary "return" after "elif", remove the "elif" and de-indent the code inside it (no-else-return)
256 mypl_lsear.py:213-31: CR121: Comparison 'self.pseek().isdecimal()' is False' should be 'self.pseek().isdecimal() is False' if checking for the singleton value False, or 'not self.pseek().isdecimal()' if testing for falsiness (singleton-comparison)
257 mypl_lsear.py:55-8: R1702: Too many nested blocks (6/5) (too-many-nested-blocks)
258 mypl_lsear.py:348-28: R1705: Unnecessary "elif" after "return", remove the "elif" and de-indent the code inside it (no-else-return)
259 mypl_lsear.py:352-28: R1705: Unnecessary "elif" after "return", remove the "elif" and de-indent the code inside it (no-else-return)
260 mypl_lsear.py:250-24: CR181: Variable name 'fullIdentifier' doesn't conform to code style naming style (invalid-name)
261 mypl_lsear.py:262-28: R1705: Unnecessary "elif" after "return", remove the leading "el" from "elif" (no-else-return)
262 mypl_lsear.py:47:4: R0911: Too many return statements (60/6) (too-many-return-statements)
263 mypl_lsear.py:47:4: R0912: Too many branches (72/12) (too-many-branches)
264 mypl_lsear.py:47:4: R0915: Too many statements (326/60) (too-many-statements)
265 mypl_lsear.py:47:4: R1710: Either all return statements in a function should return an expression, or none of them should. (inconsistent-return-statements)
266 mypl_lsear.py:9:8: W0614: Unused import(s) dataclasses and from from wildcard import mypl_token (unused-wildcard-import)
267 mypl_lsear.py:18:8: W0614: Unused import(s) MyPlError, ParseError, StatError and ValueError from wildcard import mypl_error (unused-wildcard-import)
268
269
270 Your code has been rated at 5.80/10

```

```

added 11 minutes ago in 2/ta
✓ List python code
264 appl_loader.py:47-8: MN15: Too many statements (176/50) (too-many-statements)
265 appl_loader.py:47-8: E1718: Either all return statements in a function should return an expression, or none of them should. (inconsistent-return-statements)
266 appl_loader.py:8: MN64: Unused import(s) dataclass and from wildcard import of appl_token (unused-wildcard-import)
267 appl_loader.py:18-8: MN64: Unused import(s) MyFileNotFoundException, ParameterError, StaticError and ValueError from wildcard import of appl_error (unused-wildcard-import)
268
269 -----
270
271 Your code has been rated at 5.00/10
272
273 Listing appl_token.py file...
274 ===== Module appl_token
275 appl_token.py:30-8: CM003: Trailing whitespace (trailing-whitespace)
276 appl_token.py:41-8: CM005: Trailing newlines (trailing-newlines)
277 appl_token.py:53-8: CM135: Missing class docstring (missing-class-docstring)
278
279 -----
280
281 Your code has been rated at 7.00/10
282
283 Listing appl.py file...
284 ===== Module appl
285 appl.py:26-8: CM003: Trailing whitespace (trailing-whitespace)
286 appl.py:35-8: CM003: Trailing whitespace (trailing-whitespace)
287 appl.py:37-8: CM003: Trailing whitespace (trailing-whitespace)
288 appl.py:50-8: CM003: Trailing whitespace (trailing-whitespace)
289 appl.py:51-8: CM003: Trailing whitespace (trailing-whitespace)
290 appl.py:64-8: CM003: Trailing whitespace (trailing-whitespace)
291 appl.py:78-8: CM003: Trailing whitespace (trailing-whitespace)
292 appl.py:91-8: CM003: Trailing whitespace (trailing-whitespace)
293 appl.py:105-8: CM003: Trailing whitespace (trailing-whitespace)
294 appl.py:118-12: CM003: Trailing whitespace (trailing-whitespace)
295 appl.py:132-15: CM003: Trailing whitespace (trailing-whitespace)
296 appl.py:138-8: CM005: Trailing newlines (trailing-newlines)
297
298 appl.py:13-8: MN001: Wildcard import appl_loader (wildcard-import)
299 appl.py:14-8: MN001: Wildcard import appl_loader (wildcard-import)
300 appl.py:15-8: MN001: Wildcard import appl_token (wildcard-import)
301 appl.py:18-17: MN021: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
302 appl.py:18-18: MN021: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
303 appl.py:46-4: E1722: Consider using 'sys.exit' instead (consider-using-sys-exit)
304 appl.py:38-18: MN013: Unused argument 'in_stream' (unused-argument)
305 appl.py:53-18: MN021: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
306 appl.py:61-4: E1722: Consider using 'sys.exit' instead (consider-using-sys-exit)
307
308 appl.py:53-18: MN013: Unused argument 'in_stream' (unused-argument)
309 appl.py:65-18: MN021: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
310 appl.py:76-18: MN013: Unused argument 'in_stream' (unused-argument)
311 appl.py:82-28: MN021: Redefining name 'in_stream' from outer scope (line 128) (redefined-outer-name)
312 appl.py:102-4: E1722: Consider using 'sys.exit' instead (consider-using-sys-exit)
313 appl.py:92-28: MN013: Unused argument 'in_stream' (unused-argument)
314
315 appl.py:102-4: CM183: Constant name 'about' doesn't conform to UPPER_CASE naming style (invalid-name)
316 appl.py:114-4: CM183: Constant name 'help_msg' doesn't conform to UPPER_CASE naming style (invalid-name)
317 appl.py:116-4: CM183: Constant name 'help_msg' doesn't conform to UPPER_CASE naming style (invalid-name)
318 appl.py:118-4: CM183: Constant name 'help_msg' doesn't conform to UPPER_CASE naming style (invalid-name)
319 appl.py:120-4: CM183: Constant name 'help_msg' doesn't conform to UPPER_CASE naming style (invalid-name)
320 appl.py:122-4: CM183: Constant name 'help_msg' doesn't conform to UPPER_CASE naming style (invalid-name)
321 appl.py:124-4: CM183: Constant name 'help_msg' doesn't conform to UPPER_CASE naming style (invalid-name)
322 appl.py:132-8: M0702: No exception type(s) specified (bare-exception)
323
324 appl.py:131-36: E1732: Consider using 'with' for resource-allocating operations (consider-using-with)
325 appl.py:134-13: E1732: Consider using 'sys.exit' instead (consider-using-sys-exit)
326
327 appl.py:13-8: MN013: Unused import(s) Token, dataclass, from, LoaderError, ParameterError, StaticError and ValueError from wildcard import of appl_loader (unused-wildcard-import)
328
329 -----
330
331 Your code has been rated at 3.50/10
332
333 Done linting

```

Debian package screenshots

Summary

Jobs

Run details

Usage

Workflow file

build

test

lint

build-deb-package

build-deb-package

succeeded 12 minutes ago in 6s

Build the debian package0s

Save debian package1s

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

1 ▶ Run actions/upload-artifact@v4
9 With the provided path, there will be 1 file uploaded
10 Artifact name is valid!
11 Root directory input is valid!
12 Beginning upload of artifact content to blob storage
13 Uploaded bytes 5909
14 Finished uploading artifact content to blob storage!
15 SHA256 digest of uploaded artifact zip is 8484942e7d1b87d880239315edab590b9b42cae51b07e46d9b075cef2942766f
16 Finalizing artifact upload
17 Artifact mypl-hw1-v2.0.0.deb.zip successfully finalized. Artifact ID 3025422475
18 Artifact mypl-hw1-v2.0.0.deb has been successfully uploaded! Final size is 5909 bytes. Artifact ID is 3025422475
19 Artifact download URL: https://github.com/Laurenchin8/devops-project/actions/runs/14718237985/artifacts/3025422475
```

DEVOPS-PROJECT

mypl-hw1-v2.0.0.deb

temp

DEBIAN

etc

usr

.gitignore

build.sh

debBuild.sh

hello.mypl

hw1_example.mypl

hw1_tests.py

lint.sh

Makefile

mypl

mypl_error.py

mypl_lowrapper.py

mypl_lexer.py

mypl_token.py

mypl-hw1-v2.0.0.deb

mypl.py

README.md

test.sh

tokens.mypl

viewOutput.sh

OUTPUT

TERMINAL

```
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$
```

CONSOLE

TERMINAL

PORTS

TERMINAL

bash

```
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: LBACE "{"
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 \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: RRACE "}"
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!