PROJECT STEPS FOR CREATING TESTING NIGERIAN CUSTOMS SERVICE DATA WITH PYTEST USING GitHub Codespaces, GitHub Actions, AWS codebuild, and debugging with AI.

BREAKDOWN THE PROJECT.

FIRST STAGE

i) Create a virtual environment

ii) Create your repo

iii) Introduce GitHub Codespaces, and

iv) use Linux commands to create make file, requirement.txt, auction.py, test_auction.py, transform_auction.py, test_transform_auction.py, import the cleaned and unclean CSV files, and create a folder to organize them in a folder.

SECOND STAGE

i) Import the code for auction.py, transform_auction.py, test_auction.py and test_transform_auction.py,

ii) Test using the makefile.

THIRD STAGE

I) Test on GitHub Actions (Basically doing everything we have done locally in the cloud.)

ii) Create the .yml file by setting up your workflow.

```
# WORK FLOW NAME
name: Python application test with Github Actions  # it makes the workflow a humanly-readable.

# TRIGGERS THE BUILD
on: [push] # This makes the workflow run every time we push the code to the repository

# THE JOBS SECTION
jobs:      # This defines a job named build. Jobs are groups of steps that run in a GitHub-hosted
build:     # environment.
```

```
# RUNNER ENVIROMENT
runs-on: ubuntu-latest # This job runs in a fresh Ubuntu virtual machine provided by GitHub.

# CHECKOUT THE REPOSITORY
steps:
- uses: actions/checkout@v2    # This action clones your code into the runner's workspace.

# SET UP THE PYTHON VERSION
- name: Set up Python 3.12.1
uses: actions/setup-python@v1
with:
python-version: 3.10.17

# INSTALL THE NECESSARY DEPENDENCIES FROM THE MAKE FILE IN OUR REPO
- name: Install dependencies
run: |
make install
- name: Lint with pylint
run: |
make lint
- name: Test with pytest
run: |
make test
- name: format with python Black
run: |
make format
```

iii) Copy the status badge and paste into the Read Me.

FOURTH STAGE

i) Clone your GitHub repo in AWS Cloudshell using your HTTPS or SSH connector.

ii) Change the directory to the cloned directory.

iii) Confirm if we have our files imported into the AWS cloud shell using the **ls**

iv) Run the **make all** command

v) Build and test the code on AWS Codebuild by inserting the build command structure.

version: 0.2   **# BUILDSPEC VERSION WHICH IS USUALLY VERSION 0.2**

```
phases:
  install:
    commands:
      - make install

  build:
    commands:
      - make lint
      - make test
      - make format
```

vi) store the buildspec.yml file in your repo by creating a .yml file

vi) Switch to the use a build spec file to enable AWS to trigger the buildspec file in our GitHub repo.

v) Click Create Build Project.

STEPS TO DEBUGGING LINT PYLINT WARNINGS

 # To avoid warning errors from the make file.

|| true

 # The First make lint warning is using the wrong variable name URLs in the auction.py file because there is a built-in URL module named URLs in some frameworks for example (Django's Django.urls). So pylint is sending preventive measures against using that variable name because we might decide to build a solution with a microservice(s) in the future.

auction.py:64:26: Redefining name 'urls' from outer scope (line 85) (redefined-outer-name)

 # The second make lint warning is telling us that our code to retrieve the URLs will hang indefinitely if we don't add a timeout

auction.py:70:23: issing timeout argument for method 'requests.get' can cause your program to hang indefinitely (missing-timeout)

 # The third error was kind of causing me confusion because chat GPT was not giving me the direct answers I wanted so I renamed the variable. Maybe it was because I used the name data structure and still used it as a variable, and it might a confusion for someone else I the future.

test_transform_auction.py:38:26: Redefining name 'simulated_data_structure' from outer scope (line 22) (redefined-outer-name)

 # The fifth make lint warning is telling us to remove unused libraries and functions

test_transform_auction.py:2:0: W0611: Unused numpy imported as np (unused-import)

test_transform_auction.py:2:0: W0611: Unused numpy imported as np (unused-import)

test_transform_auction.py:6:0: W0611: Unused read_url_table imported from transform_auction (unused-import)

********** Module transform_auction

transform_auction.py:3:0: W0611: Unused numpy imported as np (unused-import)

 # The sixth make lint warning tells us not to use the same variable names for two different tasks to avoid confusion and follow the Pep 20 principles

auction.py:64:26: W0621: Redefining name 'links' from outer scope (line 88) (redefined-outer-name)

 # The seventh error is make lint warning us not to reuse a variable used previously but we will ignore it because the code structure requires us to recall the fixtures to be able to test each transformed column and will not be used directly when deployed to production.