**Python local debugging using VScode**



“Local testing enables you to verify changes to skill code without deploying skill code to Amazon Web Services (AWS) Lambda. You can test your skill without leaving VS Code by using the VS Code Alexa simulator page. You can also test your local skill by using other methods, such as the Alexa simulator in the Alexa developer console and the ASK Command Line Interface (CLI).”

<https://github.com/alexa/alexa-skills-kit-sdk-for-python/tree/master/ask-sdk-local-debug>

Here we’ll see how to set up local debugging for AWS hosted lambda (not Alexa hosted)

**Prerequisites**

ASK toolkit for VSCode

In VS Code, open the View menu and choose Extensions. In the search box, type Alexa Skills Kit. Choose Alexa Skills Kit (ASK) Toolkit, and then click Install.

**Create skill**

In VS code terminal (or any terminal) start new skill

***ask new***

Please follow the wizard to start your Alexa skill project ->

? Choose a modeling stack for your skill: ***Interaction Model***

? Choose the programming language you will use to code your skill: ***Python***

? Choose a method to host your skill's backend resources: **AWS *Lambda***

Host your skill code on AWS Lambda (requires AWS account).

? Choose a template to start with: ***Hello world***

? Please type in your skill name: ***local debug***

? Please type in your folder name for the skill project (alphanumeric): ***localdebug***

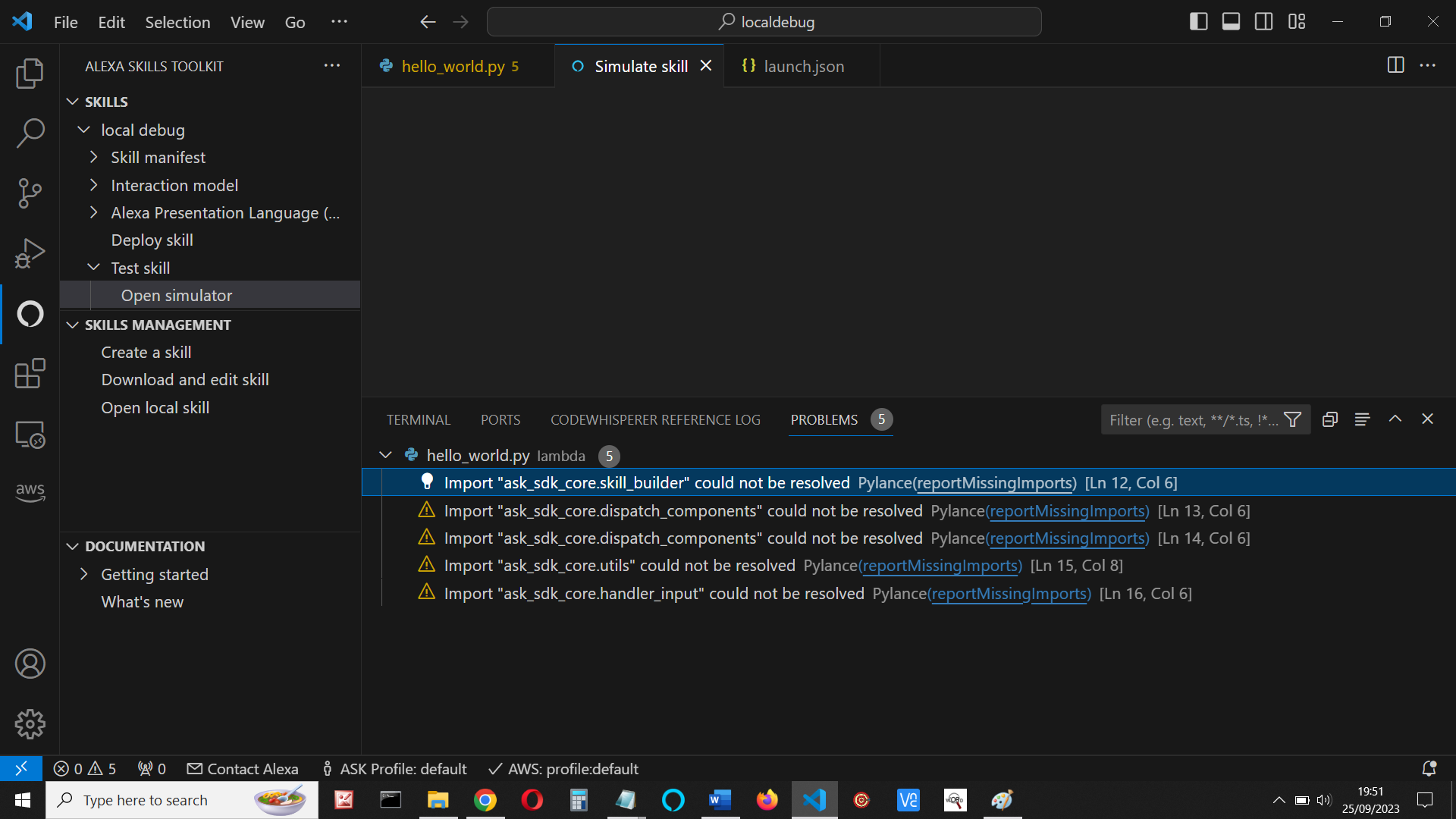
You should get Project for skill "local debug" is successfully created

**Check it works**

Check invocation: "invocationName": "local debug",

Remove the comment from the hello intent .ask("add a reprompt ..

If you have any, missing link problems e.g. ask-sdk-core,



solve it using

***pip install ask-sdk-core***

Save, deploy and test. (Either in CLI or ASK toolkit)

***ask deploy***

***ask dialog***

**Start a virtual environment (env)** (pip install virtualenv)

virtualenv env

.\env\Scripts\activate

(Add workspace folder if asked)

**Install ask-sdk-local-debug and add debugger configuration**

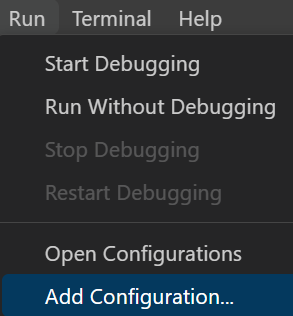
cd to **lambda folder** and add the sdk local debug:

***pip install ask-sdk-local-debug -t .*** (the dot is important)

**Add debugger configuration**.

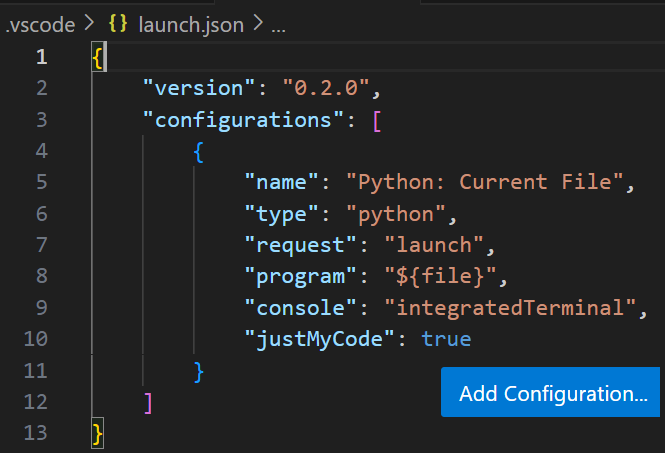
1. Create a launch.json file for your skill:

In VScode, display your hello\_world.py code, then select Run menu, and then choose Add Configuration



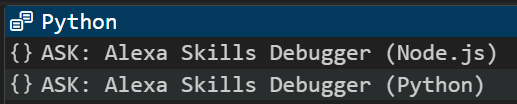
This creates a launch.json file debugger for the current file.

Remove the // comments.

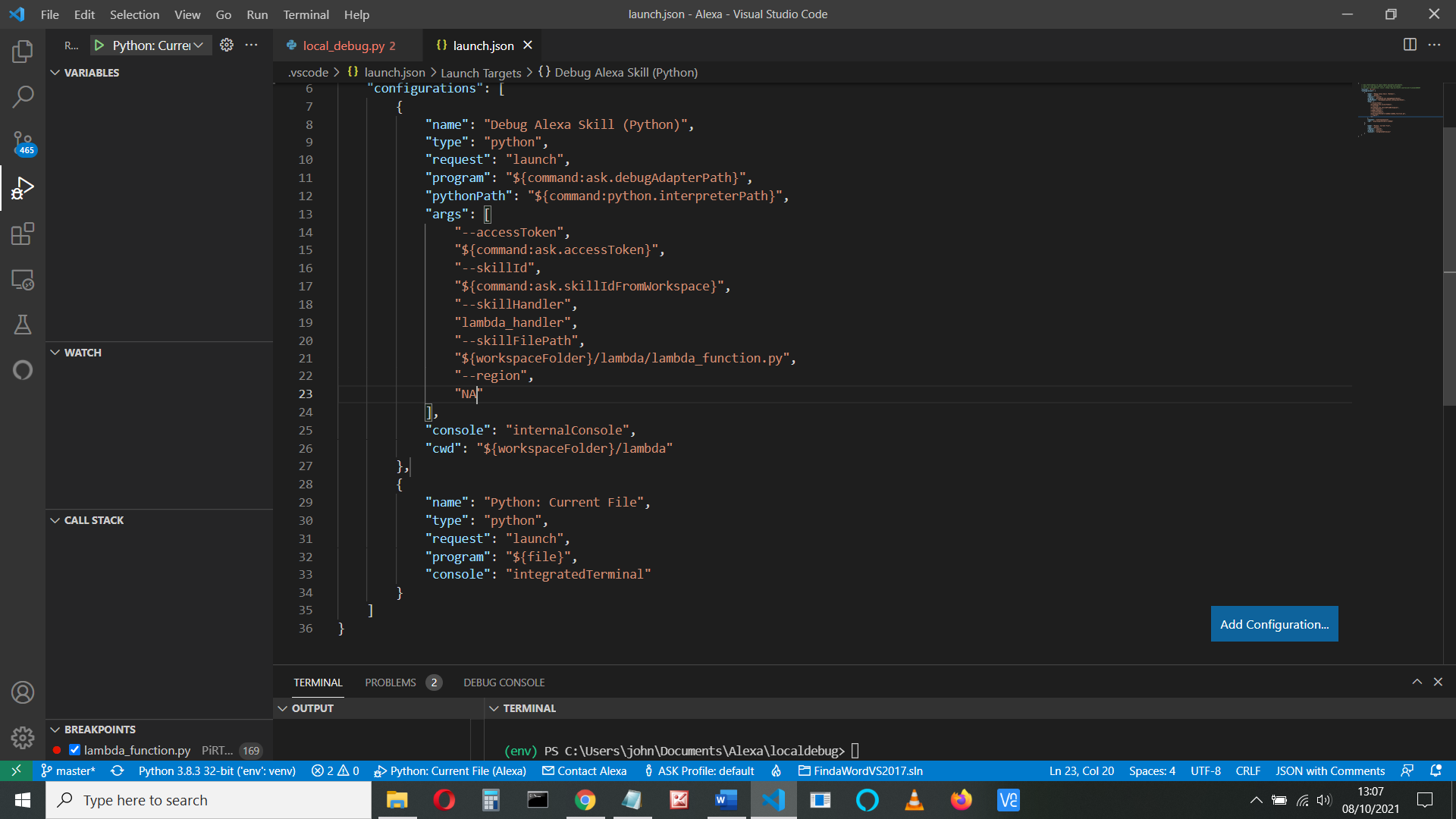


Now click Add Cofiguration again.

1. In drop-down box, choose **ASK: Alexa Skills Debugger (Python)**



So that your launch.json looks like this



Check your region is correct – we’re in the EU

                "--region",

                "EU"

Modify launch.json

We don’t actually need the Python: Current File, but I’ll leave it.

Remove the // comment lines if you haven’t done so.

Change PythonPath to python. Change the bold lines below.

Change python line to "**python":** "C:/Python311/python", (or wherever your python is)

You can find that with windows using where or where.exe from C: e.g:

C:\>**where python**

C:\Python311\python.exe

C:\Users\john\AppData\Local\Programs\Python\Python310\python.exe

Change the “program” line to point to the **debugger invoker (**not the debugger itself)

"program": **"${workspaceFolder}./lambda/ask\_sdk\_local\_debug/local\_debugger\_invoker.py",**

If you have created an AWS hosted file, you need to change these lines, as below:

"--skillHandler",

**"handler",**

"--skillFilePath",

**"${workspaceFolder}/lambda/hello\_world.py",**

**(**or whatever your lambda py file is called)

Otherwise, for an Alexa Hosted skill, leave them alone (as lambda\_handler and lambda\_function.py)

My launch.json is now:

{

"version": "0.2.0",

"configurations": [

{

"name": "Python: Current File",

"type": "python",

"request": "launch",

"program": "${file}",

"console": "integratedTerminal",

"justMyCode": true

},

{

"name": "Debug Alexa Skill (Python)",

"type": "python",

"request": "launch",

"program": **"${workspaceFolder}./lambda/ask\_sdk\_local\_debug/local\_debugger\_invoker.py",**

"**python":** "C:/Python311/python",

"args": [

"--accessToken",

"${command:ask.accessToken}",

"--skillId",

"${command:ask.skillIdFromWorkspace}",

"--skillHandler",

**"handler",**

"--skillFilePath",

**"${workspaceFolder}/lambda/hello\_world.py",**

"--region",

**"EU"**

],

"console": "internalConsole",

"cwd": "${workspaceFolder}/lambda"

}

]

}

**Debug your file**

Set breakpoint

Open 2 terminals. In one type **ask run**

This will give you an hour of debugging.

Start the debugger

In the second terminal type **ask dialog**

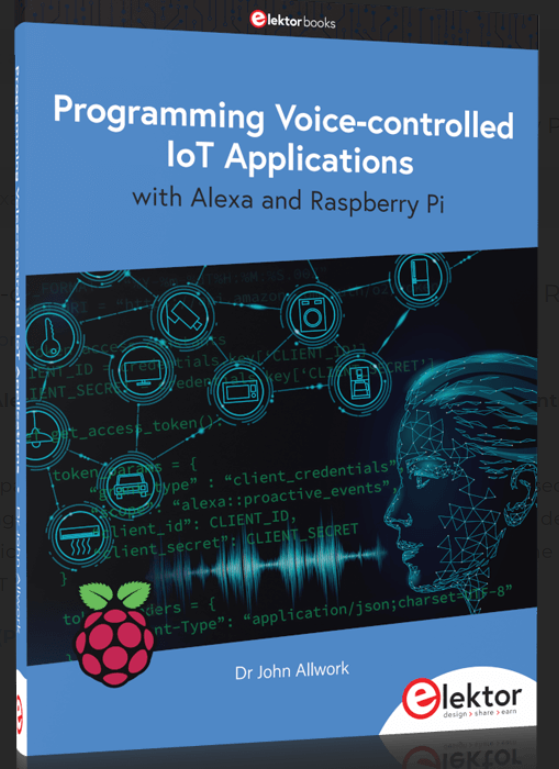
And debug your code!

Run to breakpoint, **change**, watch or inspect variables.

https://github.com/microsoft/vscode/issues/130170

Note that if you change your code you need to restart the debugger, but you don’t have to restart debug session (ask dialog)

**My book**

****

**https://www.elektor.com/programming-voice-controlled-iot-applications-with-alexa-and-raspberry-pi**

**References**

Node local debugging. Craig Walls:

https://www.youtube.com/watch?v=HhFZYJYvHbk

https://www.andreasjakl.com/local-debugging-of-alexa-skills-with-visual-studio-code/

<https://developer.amazon.com/en-US/docs/alexa/hosted-skills/alexa-hosted-skills-ask-cli.html>

<https://code.visualstudio.com/docs/python/debugging>

<https://github.com/alexa/alexa-skills-kit-sdk-for-python/tree/master/ask-sdk-local-debug>

More on VS code debugging

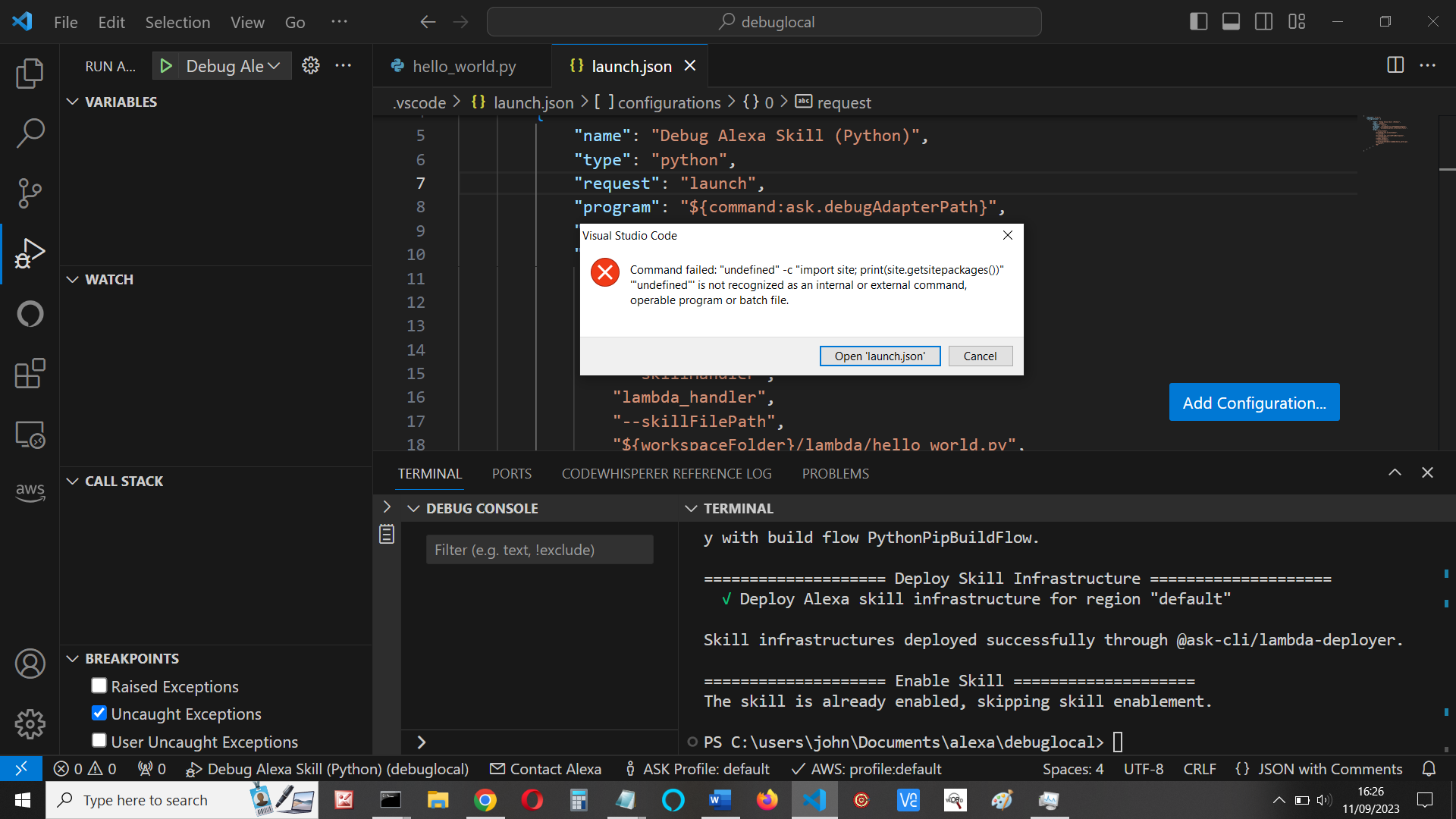
<https://code.visualstudio.com/docs/editor/debugging>

<https://github.com/alexa/ask-toolkit-for-vscode/issues/99>

Rahul dabble labs

<https://www.youtube.com/watch?v=-6EwAqmVjaY>

**Troubleshooting:**



It’s not getting ask.debugAdapterPath

See https://github.com/alexa/ask-toolkit-for-vscode/issues/270

My temporary solution was to directly link the debugger file.  
In the launch.json file replace:  
"program": "${command:ask.debugAdapterPath}"

with  
"program": "${workspaceFolder}/.venv/lib/python3.11/site-packages/ask\_sdk\_local\_debug/local\_debugger\_invoker.py",

find it. Mine’s at:

C:\Users\john\Documents\Alexa\localdebug\env\Lib\site-packages\ask\_sdk\_local\_debug\local\_debugger\_invoker.py

And replace in launch.json:

"program": "$C:\\Users\\john\\Documents\\Alexa\\localdebug\\env\\Lib\\site-packages\\ask\_sdk\_local\_debug\\local\_debugger\_invoker.py",

**Exception has occurred: ModuleNotFoundError**

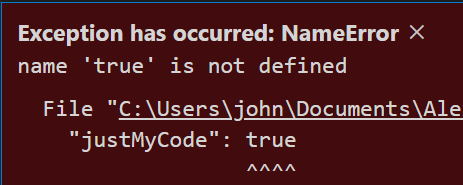
No module named 'ask\_sdk\_local\_debug'

File "C:\Users\john\Documents\Alexa\debuglocal\env\Lib\site-packages\ask\_sdk\_local\_debug\local\_debugger\_invoker.py", line 21, in <module> from ask\_sdk\_local\_debug.local\_debugger import LocalDebugger

Command failed: "undefined" -c "import site; print(site.getsitepackages())"

Check: "program": "${workspaceFolder… is correct (point to debugger installer,py)

Error: ‘true is not defined’



You’ve executed the wrong debugger.

Try again, selecting **Alexa Skills Debugger (Python)** from the green triangle drop down box