* Python is an easiest language.
* Computer created by electronic circuit and deals in binary language 0 and 1.
* Computer Languages

1. Low Level Language : deals with hardware
2. Middle Level Language
3. High Level Language : Python

* Interpreter : translate after every line like Python
* Compiler : translate after completion of task like C#
* A compiler typically runs a program faster because it translates an entire program at once. Meanwhile, the interpreter translates a program line by line which makes its overall speed and execution slower.
* Python is an Interface language as we can connect everything with Python from backend.
* ASCII Codes : Keep pressing “Alt” key and press keys “6” and “5” from right side Numeric Keys of keyboard and when you release “Alt” key then you get “A” as “65” is ASCII code for “A”.
* The default language of AI is Python.
* To work online at Python language, you can use google “COLAB”

1. Go to <https://colab.research.google.com/>
2. Go to “New notebook” or go to “File” option, select “New notebook in Drive”
3. print("HELLO WORLD")
4. Run and get “HELLO WORLD” printed

* Google “COLAB” provides high configuration 130GB hard disk, 12 GB – RAM and high speed CPU at runtime. Even we can runtime configuration to more high speed by going to “Runtime” option then select “Change runtime type” and select CPU.
* Python is case sensitive language.

Example: correct format: print(“ABC”) while wrong format: Print(“ABC”)

* Open command prompt and run commands:

1. docker pull python:3.9-slim

Download and install python image

1. docker ps

It returns information related to images running container at docker

* <https://github.com/panaversity/learn-cloud-native-modern-ai-python/tree/main/07_hello_world_containerize_python>
* “py” is extension of Python file.
* To check Python installation, in cmd write command

python –version

* You can download Python from “Microsoft Store”
* Deploy project at docker container:

1. Create new folder in any drive like “TestProject1” then open folder in Visual Studio Code
2. Create new file “app.py” and write in file

print(“HELLO WORLD”)

if file not save then “dot” sign appears with filename at top, after saving file close sign appears

1. Open “Terminal” from “View” option, select “Terminal” or shortcut keys Ctrl + ~
2. In terminal, command to run file

python app.py

1. We can run above command in command prompt
2. We can go to Python terminal by command

python

then you can write python code in python terminal

print(“DEF”)

to exit python terminal

exit()

1. Create a file with name – “Dockerfile” and you get docker logo with file. Remember that docker file name should be “Dockerfile”.
2. In docker file, write

FROM python:3.12-slim

WORKDIR /app

COPY . /app/

CMD ["python", "app.py"]

Here, “FROM python:3.12-slim” uses python image (downloaded from dockerhub)

“WORKDIR /app” will create “app” directory at linux

“COPY . /app/” will copy all file present at “.” (dot) current directory of windows and paste to linux directory “app”

“CMD ["python", "app.py"]” execute command “python app.py” where space is replaced by comma and enclosed in double quotes

* Docker container working

1. Create new folder in any drive like “TestProject2” then open folder in Visual Studio Code
2. Create new file “app.py” and write in file

print(“ABC”)

1. Create a file with name – “Dockerfile”.In docker file, write

FROM python:3.12-slim

WORKDIR /app

COPY . /app/

CMD ["python", "app.py"]

1. In terminal, run command

docker build -t helloworldfirstapp .

Here, docker will create an image with name “helloworldfirstapp” and “.” (dot) is path of docker file and you can check image at Docker Desktop

“-t” is tag switch.

1. In terminal, run command

docker run helloworldfirstapp

Here, “helloworldfirstapp” image will run in docker container and you will get below output

ABC

Docker container executed then closed

1. To keep docker container open after execution

docker run -it helloworldfirstapp

* Whenever we execute docker container run command then a new container created
* VS Code -> Extensions -> install “DevContainers“ which contains official Microsoft checkmark. After installation, you can get “Remote Explorer” in left side menu items.
* After installation, reload window by pressing ”Ctrl + Shift + P” then write “Reload Window” and select option “Reload Window”
* Instead of working in terminal, we can work directly in Docker container as

1. Create new folder in any drive like “TestProject3” then open folder in Visual Studio Code
2. Create new file “app.py” and write in file

print(“ABC”)

1. Create a file with name – “Dockerfile”.In docker file, write

FROM python:3.12-slim

WORKDIR /app

COPY . /app/

CMD ["python", "app.py"]

1. VS Code
2. Remote Explorer (in left menu)
3. IF no containers are present in Docker container then “Open Folder in Container” button, you will get new window

ELSE click on “+” button then select “Open Current Folder in Container” (if you want to deploy folder in which you are present in VS Code) OR select “Open Folder in Container” (if you want to deploy another folder)

1. Select project folder which you want to deploy to Docker container
2. Select folder and click on “Open” button
3. Select “Add configuration to user data folder”
4. Select “from ‘Dockerfile’”
5. Click “OK” button
6. In terminal, you can verify Docker deployment as path change to

root@940aae09d9ae:/workspaces/TestPorject4#”

1. You can check Docker Desktop and found that
2. a new docker image is created with name like

vsc-testporject4-7e5fa702aec7c6b9

1. a new container is running with name like

awesome\_solomon

and image of this container is “vsc-testporject4-7e5fa702aec7c6b9” (mentioned in point (a))

* Previously, we have to work on code and then deploy code on dev container but now, we can work directly on Docker container and test changes to get same environment like production.
* Install “Python” extension in VS Code then you will get play button at top right side in python file. If you click at play button then python file code executes and output is shown in “Python” terminal.
* Install “Dev Containers” extension in VS Code.
* Python is weakly-typed/dynamic language.

Example: a = “xyz”

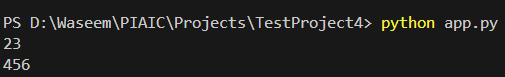
A = 100

* For data type safety, install “Pylance” extension in VS Code.
* Shortcut keys to open terminal

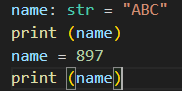
1. Ctrl + J
2. Ctrl + ` (key before 1)

* In python, semi-colon is not needed in the end of statement
* Python code
* name: str = "23"
* age:int = 456
* print (name)
* print(age)

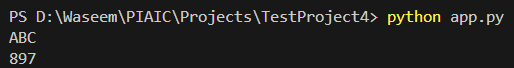
Output:



* Python is dynamic type



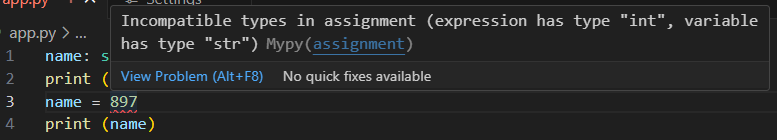
Output:



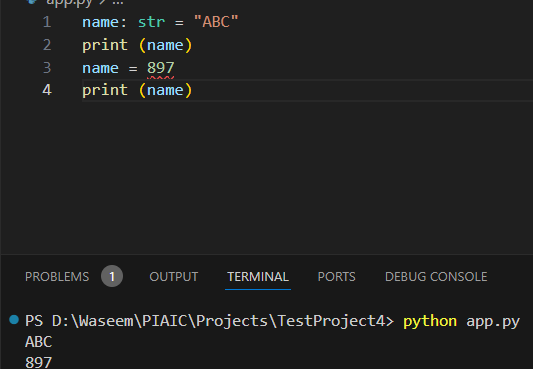
* Install extensions in VS Code, “Pylance” and “”Mypy Type Checker” having Microsoft check
* Even after installation, go to settings from left menu options, write “type check” in search box

1. Select “Mypy Type Checker”, in the last, set drop down value from “off” to “always” for field “Mypy-type-checker: Show Notifications”
2. Select “Pylance”, set drop down value from “off” to “basic” for field “Python › Analysis: Type Checking Mode”

After following any one step from above two, we get wrong value setting underlined with red color and on hovering text, it shows reason.



But even after following above step, we will be able to execute python file and get output.



Because these extensions “Pylance” and “”Mypy Type Checker” just highlight data type casting issue but they are not integrated with python compiler.

* pip install : To install modules locally, you need to create and activate what is called a virtual environment, so “pip install” installs to the folder where that virtual environment is located, instead of globally (which may require administrator privileges). If we use “pip install” to install packages then if we delete container then all extensions would also deleted.

e.g.: pip install pandas

* Python is weakly type language.
* A Docker image cannot be removed if a container is present for that image. First, we have to remove container then we will be able to remove image.