**NHÓM 3**

1. **Đoàn Vũ Minh Thanh**
2. **Huỳnh Văn Trinh**
3. **Bạch Ngọc Minh Trúc**
4. **Đặng Thị Cẩm Tú**
5. **Mai Hạ Vy**

**BÀI TẬP THẢO LUẬN NHÓM BUỔI 3**

*“Viết chương trình quét thông tin trên thẻ”*

*(Nguồn tham khảo: AWS Develop Guide* [1]*)*

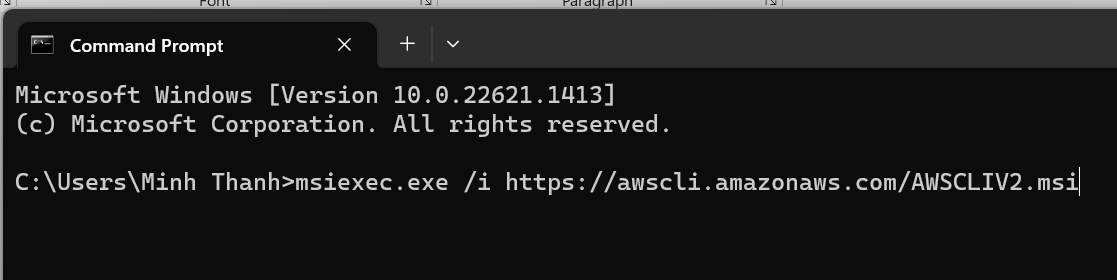
# Bước 1: tạo tài khoản AWS

Nhóm sử dụng tài khoản do thầy cung cấp

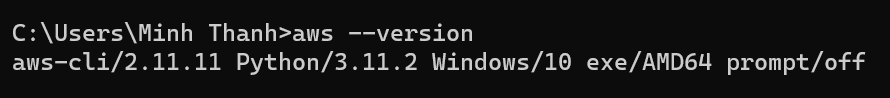
# Bước 2: Cài đặt AWS CLI và AWS SDKs

Để cài đặt AWS CLI, ở cửa sổ Command Prompt, nhập dòng lệnh

“**msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi**”

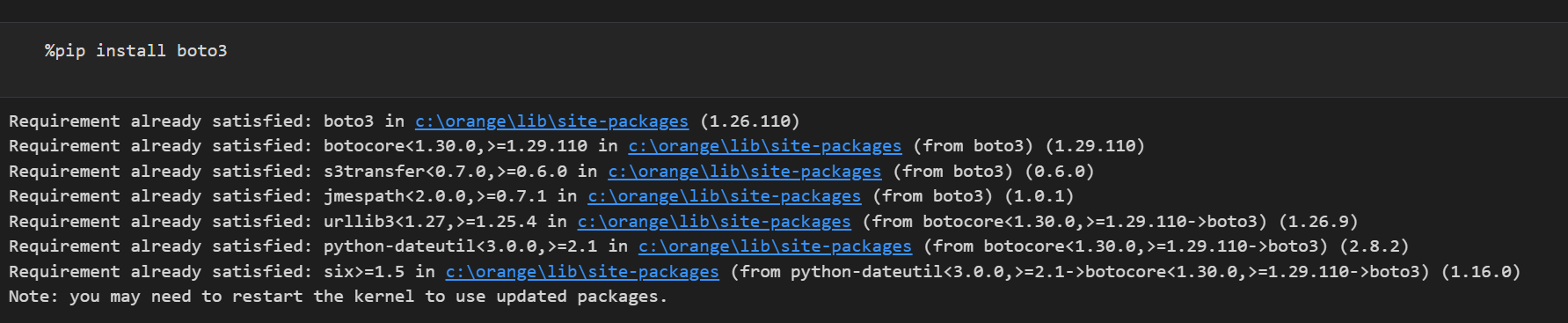


Sau khi cài đặt thành công, kiểm tra phiên bản AWC CLI đã cài đặt “**aws --version**”



Để cài đặt AWS SDKs cho ngôn ngữ lập trình Python, mở 1 file .py hoặc .ipynb và nhập dòng lệnh

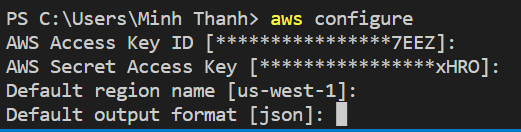
“**%pip install boto3**”

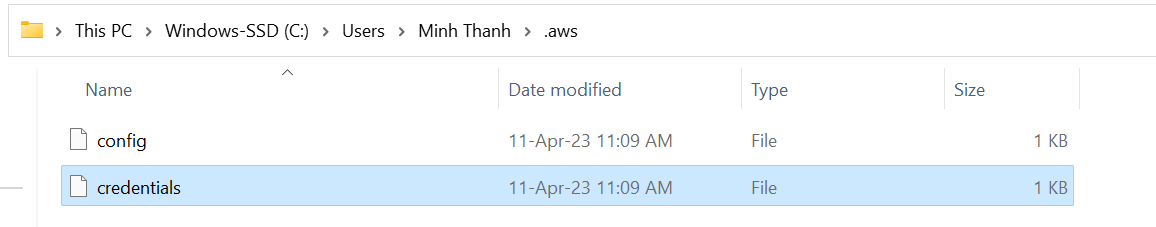


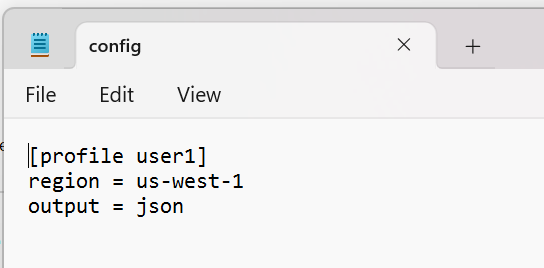
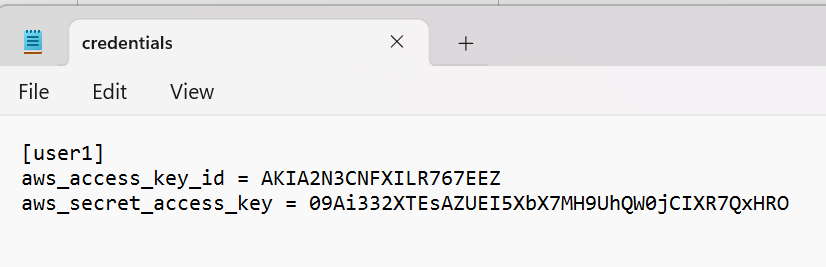
# Bước 3: Cài đặt cấu hình (configuration) cho AWS CLI tương tác với AWS

Nhóm sử dụng access key id và secret access key thầy đã cung cấp trên MS Teams.

Sau đó, để liên kết với AWS, nhóm nhập lệnh “aws configure” và khai báo các thông tin cần thiết gồm: AWS Access Key ID, AWS Secret Access Key, region name, output format



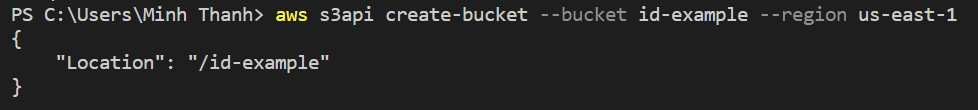




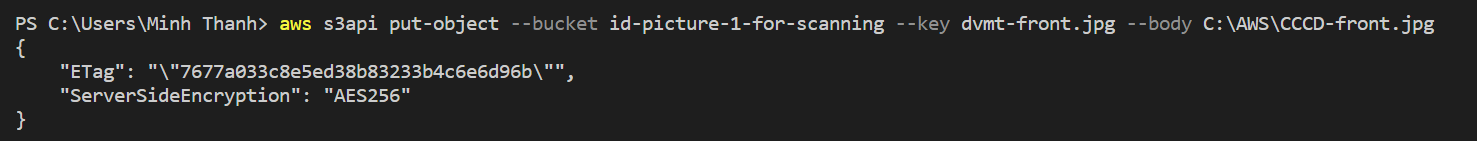
# Bước 4: Đăng tải hình ảnh thẻ lên cloud Amazon S3

Để đăng tải hình ảnh lên s3, phải tạo trước 1 bucket để chứa object là hình ảnh đăng tải. Bucket giống như 1 folder, và objects là các file chứa trong bucket (folder) đó.

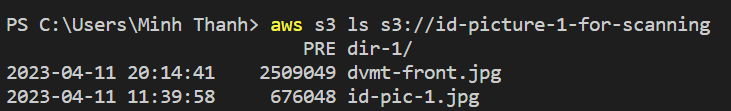
Để tạo bucket, nhập dòng lệnh “**aws s3api create-bucket --bucket [tên bucket] --region [tên region]**”



Sau khi đã tạo thành công bucket, đăng tải hình ảnh muốn quét thông tin lên bucket vừa tạo bằng lệnh “**aws s3api put-object --bucket [tên bucket] --key dir-1/[key của object] --body [path dẫn đến nơi lưu object hiện tại]**”



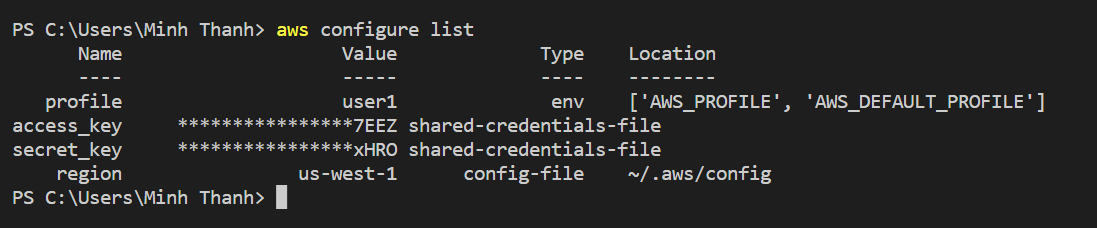
Để kiểm tra xem file được đăng tải thành công chưa, nhập lệnh “**aws s3 ls s3://[tên bucket]**”

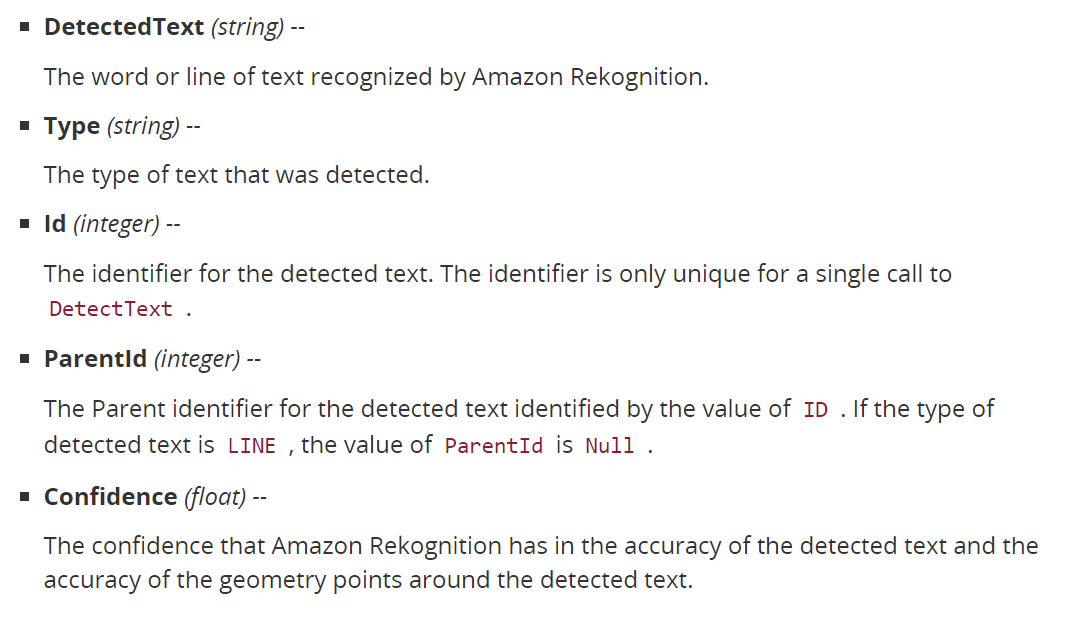
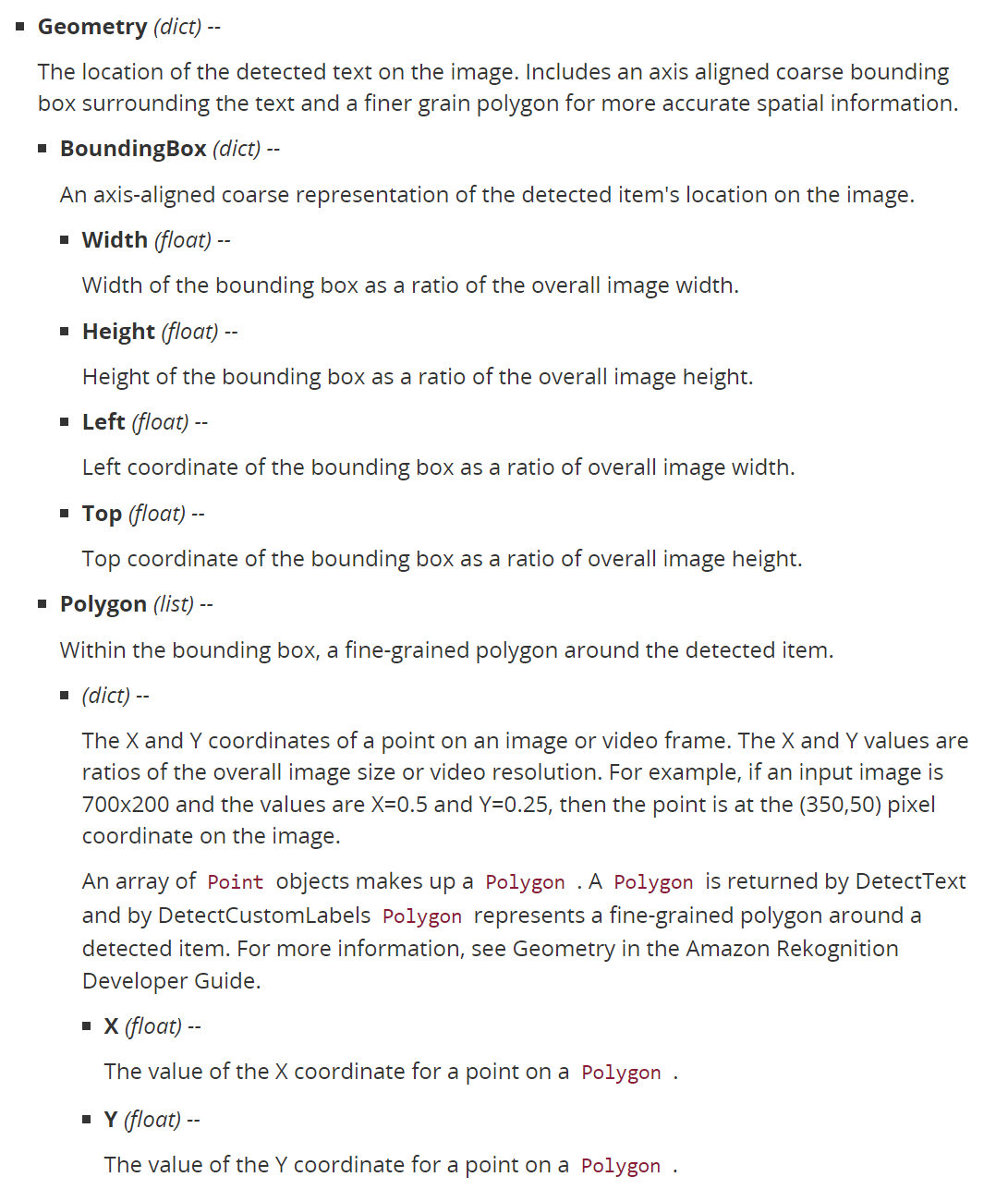


# Bước 5: Thực hiện scan ảnh



* Khai báo hàm nhận dạng văn bản detect\_text() với 3 tham số đầu vào là session, photo - ảnh cần quét văn bản, và bucket – nơi chứa văn bản (object) trong cloud s3.
  + Để kết nối đến dịch vụ AWS s3, ta tạo biến session sử dụng hàm Session trong thư viện boto3 với tham số “profile\_name” là tên người dùng đã khai báo ở phần cài đặt cấu hình

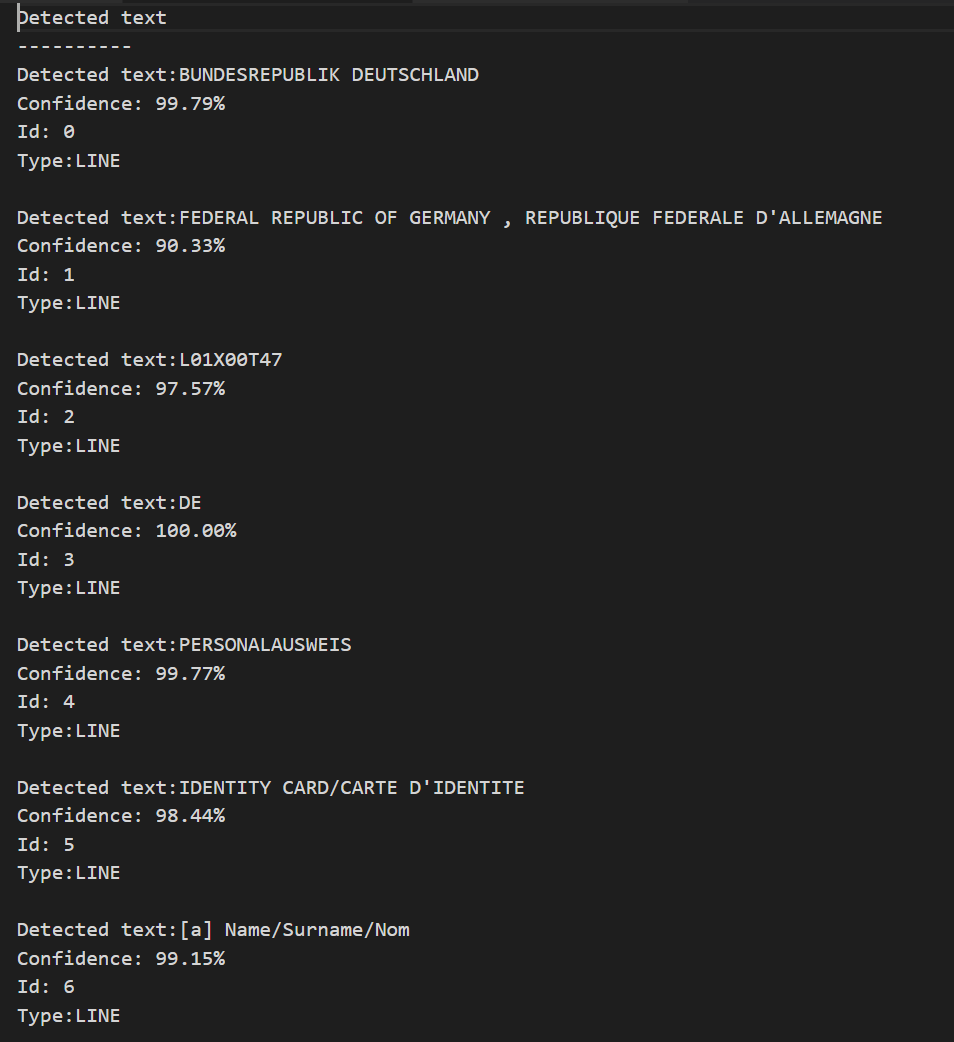


* + Tiếp theo, ta tạo biến client và sử dụng hàm Client trong thư viện session để khai báo dịch vụ cần sử dụng là nhận dạng – rekognition
  + Biến response sẽ khai báo việc sử dụng dịch vụ nhận dạng văn bản detect\_text qua hình ảnh Image, được lấy từ thư viện s3 bằng cách gọi tên bucket chứa ảnh và tên ảnh. Output của hàm detect\_text là một từ điển (dict) với 2 key là “TextDetections” và “TextModelVersion”, do vậy response là một từ điển – dict.
  + Lưu kết quả quét văn bản vào biến textDetections bằng cách lấy giá trị (value) (là một danh sách chứa nhiều từ điển con) của key “TextDetections”
  + Thông tin các văn bản quét được từ ảnh sẽ lưu vào 1 danh sách (list), mỗi loại thông tin là 1 từ điển con với khóa là tên của loại thông tin, và giá trị là nội dung của loại thông tin đó. Các loại thông tin hàm cung cấp gồm:
    - 
    - 
    - Ở đây nhóm in ra các loại thông tin: nội dung văn bản quét được, độ chính xác, số thứ tự của đoạn văn bản quét ra được, loại văn bản
  + Nhóm thực hiện vòng lặp in từng đoạn văn bản quét được từ hình ảnh được chỉ định với các loại thông tin mà nhóm đã chọn ở phía trên.
  + Hàm sẽ trả về số lượng các đoạn văn bản vừa quét được từ hình ảnh



* Khai báo hàm làm việc chính main():
  + Khai báo session
  + Khai báo môi trường/không gian ảo (cloud) lưu trữ tài liệu là s3
  + Khai báo biến my\_bucket là tên bucket sẽ lưu trữ các hình ảnh được đăng tải lên s3
  + Sử dụng vòng lặp for để in thông tin văn bản quét được với từng ảnh lưu trong bucket

# Kết quả



(1 phần output)

# References

|  |  |
| --- | --- |
| [1] | A. D. Guide, "AWS - Amazon Web Service," [Online]. Available: https://docs.aws.amazon.com/rekognition/latest/dg/text-detecting-text-procedure.html?fbclid=IwAR0CpAnIdEaoWv33R3VMrG0TNDB5X87VrFmz5voZLzJsVbHROUORG0S9bJk. [Accessed 13 April 2023]. |