

## Zadanie 1 – opis

W zadaniu nr 1 moim celem było pobranie zestawu danych ze strony

<https://data.nasdaq.com/data/ECONOMIST-the-economist-big-mac-index/usage/quickstart/api>

i zapisaniu go za pomocą skryptu Pythona na S3.

Po wejściu na ww. stronę, pobrałem dwa pliki .CSV dla Polski i Rumunii.

Następnie na stronie [aws.amazon.com](https://aws.amazon.com) wszedłem do IAM dashboard aby stworzyć użytkownika

The screenshot shows the AWS IAM console interface. On the left is a navigation menu with options like 'Dashboard', 'Access management', 'Users', 'Roles', 'Policies', etc. The main content area displays the 'Summary' page for a user named 'keepay18'. It shows the 'User ARN' as 'arn:aws:iam::577283502558:user/keepay18', the 'Path' as '/', and the 'Creation time' as '2022-07-07 16:55 UTC+0200'. Below this, there are tabs for 'Permissions', 'Groups (1)', 'Tags', 'Security credentials', and 'Access Advisor'. The 'Permissions' tab is active, showing 'Permissions policies (2 policies applied)'. It lists two policies: 'Attached directly' (AWSCompromisedKeyQuarantineV2) and 'Attached from group'. At the bottom, there is a 'Show 1 more' link.

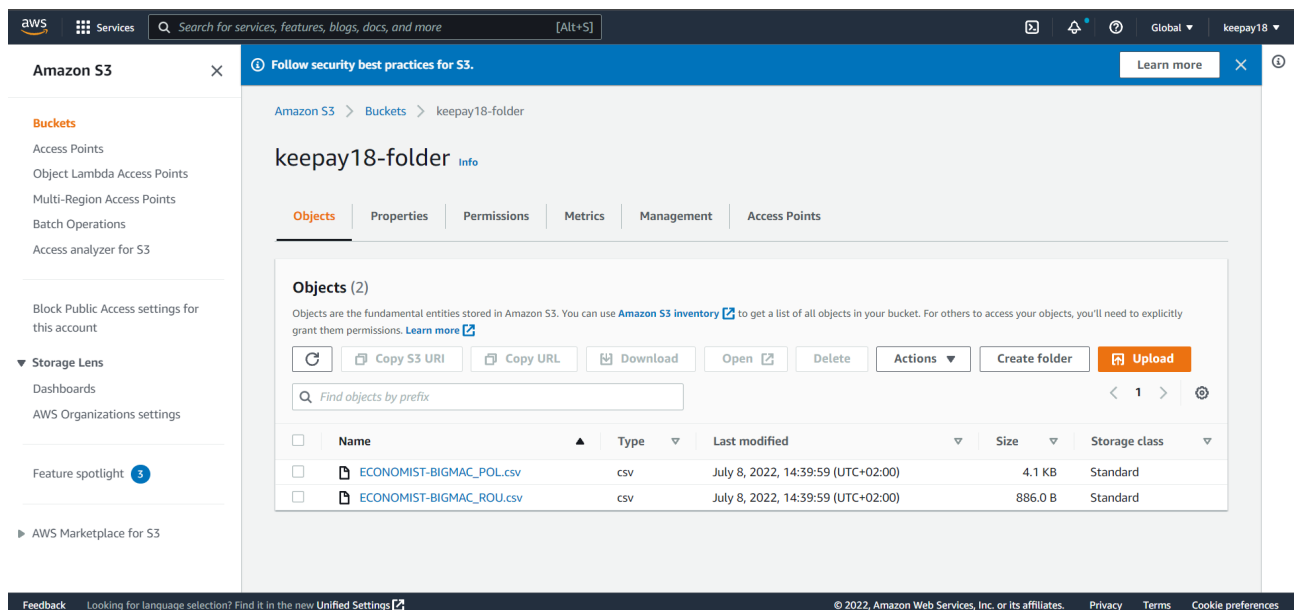
Po utworzeniu użytkownika konieczne było również stworzenie grupy

The screenshot shows the AWS IAM console interface for a user group named 'S3Storage'. The left navigation menu is visible. The main content area displays the 'Summary' page for the 'S3Storage' group. It shows the 'User group name' as 'S3Storage', the 'Creation time' as 'July 07, 2022, 16:55 (UTC+02:00)', and the 'ARN' as 'arn:aws:iam::577283502558:group/S3Storage'. Below this, there are tabs for 'Users', 'Permissions', and 'Access Advisor'. The 'Users' tab is active, showing 'Users in this group (1)'. It lists one user: 'keepay18'. The user's 'Groups' are listed as '1', 'Last activity' as 'None', and 'Creation time' as '3 days ago'.

Politykę prywatności ustawiłem na „AmazonS3FullAccess” i dodałem wcześniej utworzonego użytkownika.

Po utworzeniu grupy skopiowałem „Access Key ID” i „Secret access key”, które będą mi potrzebne w skrypcie.

W kolejnym kroku przeszedłem do zakładki S3 aby utworzyć „bucket”



Po wykonaniu tych czynności przeszedłem do skryptu i na samym początku zainstalowałem „Amazon AWS Python SDK”, poprzez wpisanie komendy „pip install boto3”

W pierwszych liniach kodu zaimportowałem potrzebne biblioteki:

```
1 import os
2 import boto3
3 from botocore.exceptions import ClientError
```

Przypisałem również do zmiennych skopiowane wcześniej klucze jak i nazwę mojego folderu:

```
5 access_key = 'AKIAYM2GHGHPFOYRRU4A'
6 access_secret = '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'
7
8 bucket_name = 'keepay18-folder'
```

W kolejnym kroku utworzyłem instancję klasy boto3 i użyłem na niej metody client, w argumentach której zapisałem: nazwę serwisu i zmienne z kluczami:

```
16 client_s3 = boto3.client('s3', aws_access_key_id=access_key,
17                             aws_secret_access_key=access_secret)
```

Do zmiennej „data\_file\_folder” przypisałem ścieżkę skąd będą wysyłane pliki do S3:

```
14 data_file_folder = os.path.join(os.getcwd(), 'files')
```

Następnie skorzystałem z pętli for używając do tego wyjątków aby przesłać pliki do „bucketu”. Na utworzonej wcześniej instancji klasy zastosowałem metodę upload\_file, w argumentach której umieściłem utworzone zmienne. Skorzystałem jeszcze z metody print aby wyświetlała, które pliki się wysyłają:

```
22 for file in os.listdir(data_file_folder):
23     if not file.startswith('~'):
24         try:
25             print("Uploading file {}".format(file))
26             client_s3.upload_file(
27                 os.path.join(data_file_folder, file),
28                 bucket_name,
29                 file
30             )
31         except ClientError as e:
32             print('Credential is incorrect')
33             print(e)
34         except Exception as e:
35             print(e)
```

Po wykonaniu skryptu, pliki znajdują się już w S3.