**AWS S3 Object Lambda – Explained for Beginners**

AWS **S3 Object Lambda** allows you to **modify and process files in S3 when they are requested**, without storing extra copies. Instead of keeping multiple versions of a file, **S3 Object Lambda automatically changes the file before sending it to the user**.

For example, if you store **high-resolution images** in S3, you can use **S3 Object Lambda** to **resize them on the fly** when users request them.

**Why Use S3 Object Lambda?**

* **Saves Storage Space** – No need to store different versions of the same file.
* **Customizes Data Per User** – Different users can see different versions of the same file based on rules.
* **Works with Existing Applications** – No need to change how your application accesses S3.
* **Supports Various Use Cases** – You can filter, format, modify, or convert data before delivering it.

**How Does S3 Object Lambda Work?**

1. **A user requests a file from S3**
2. **S3 Object Lambda runs an AWS Lambda function** to process the file
3. **Lambda modifies the file** (for example, resizing an image or filtering sensitive data)
4. **The modified file is returned to the user** instead of the original file

The **original file remains unchanged** in S3.

**Use Cases for S3 Object Lambda**

* **Watermarking Images** – Add a watermark to images without storing separate watermarked versions.
* **Resizing Images** – Provide different image sizes for different devices without keeping extra copies.
* **Filtering Data** – Hide sensitive information, such as masking personal data in documents.
* **Format Conversion** – Convert files on demand, such as converting a JSON file into a CSV file before download.

**How to Set Up S3 Object Lambda?**

**Step 1: Create an AWS Lambda Function**

1. Go to **AWS Lambda Console**
2. Click **Create Function** and choose **Author from scratch**
3. Select a **Runtime** (such as Python, Node.js, or Java)
4. Write code that processes the requested file
5. Deploy the Lambda function

**Step 2: Create an S3 Object Lambda Access Point**

1. Go to **AWS S3 Console**
2. Click **Access Points** and choose **Create Object Lambda Access Point**
3. Select your **S3 bucket**
4. Attach the **Lambda function** that will process the files
5. Click **Create Access Point**

Now, whenever a file is requested through this access point, **the Lambda function will modify it before delivering it to the user**.

**Example: Convert JSON to CSV on the Fly**

If a user requests a JSON file, but they need it as a CSV, **S3 Object Lambda can convert it before sending**.

Here’s an example Lambda function that converts JSON to CSV:

import json

import csv

import boto3

def lambda\_handler(event, context):

s3 = boto3.client('s3')

bucket = event['getObjectContext']['inputS3Url']

response = s3.get\_object(Bucket=bucket, Key='data.json')

json\_data = json.loads(response['Body'].read().decode('utf-8'))

csv\_data = "name,age,city\n"

for item in json\_data:

csv\_data += f"{item['name']},{item['age']},{item['city']}\n"

return {

'statusCode': 200,

'body': csv\_data,

'headers': {'Content-Type': 'text/csv'}

}

With this setup, **users downloading the file will receive a CSV instead of the original JSON**, without storing extra copies.

**Things to Keep in Mind**

* **No Extra Storage Costs** – You do not need to store modified versions of files.
* **Lambda Processing Costs Apply** – You are charged for the execution time of the Lambda function.
* **Limited to GET Requests** – S3 Object Lambda only works when retrieving files, not modifying them directly.
* **Requires IAM Permissions** – Lambda needs permission to read files from S3.

**Final Thoughts**

AWS **S3 Object Lambda** is a **powerful tool** that allows you to **process and modify files in real time** without storing extra versions. Whether you need to **resize images, filter sensitive information, or convert file formats**, **S3 Object Lambda can do it on demand**, saving storage space and improving flexibility.

Would you like help setting up an **S3 Object Lambda function** for your specific needs?