12 min
Video: Amazon Lex Walkthrough Part 2 10 min
Video: Exercise 1.1 Introduction 2 min
Reading: Exercise 1.1 Creating a Simple Bot with Lex 40 min
Quiz: Amazon Lex 5 questions
Creating a Serverless Website with S3
Video: Creating a Serverless Website with Amazon S3 2 min
Video: Creating a Serverless Website with S3 Demo 7 min
Video: Exercise 1.2 Introduction 1 min

Exercise 1.2: Creating a S3 Bucket and Configuring as a Static Website

Exercise 1.2: Creating a S3 bucket and configuring as a static website

Now we have our chatbot fundamentals out of the way. I would like to put that to one side and revisit it in week 4.

What I'd like us to focus on next is the building of a database driven web application that figures out the temperature when we tell it a particular city.

As you would imagine, creating an application like this involves many moving parts.

We will need:

- a front end web site

12 min
Video: Amazon Lex Walkthrough Part 2 10 min
Video: Exercise 1.1 Introduction 2 min
Reading: Exercise 1.1 Creating a Simple Bot with Lex 40 min
Quiz: Amazon Lex 5 questions
Creating a Serverless Website with S3
Video: Creating a Serverless Website with Amazon S3 2 min
Video: Creating a Serverless Website with S3 Demo 7 min
Video: Exercise 1.2 Introduction 1 min
Reading: Exercise 1.2: Creating a S3 Bucket and Configuring as a Static

we saw stage if stage by stage and tirst focus on just the web front end, and not concern ourselves yet with wiring it up to a backend. We only need to worry about hosting some HTML, CSS and little bit of javaScript.

We already have a website front end all prepared for you in a zip file. I did promise you no coding remember :)

All you need to do is download this zip file, unzip it and push it into the cloud. We will use S3 (Simple Storage Service) which can happily "host" these kinds of static websites.

Later on, after we have this website uploaded and everything is configured for website hosting. We can get a little fancy and add a content delivery network in front of it. Ready for global domination :). Aiden will talk to you about that in one of the upcoming videos.

At things in good time though, let's get this static website up and running on S3 first.

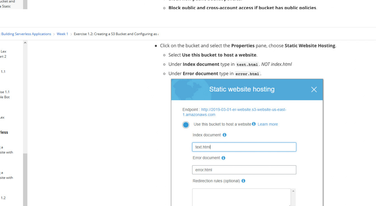
You'll need to create what we call a "bucket" to store our web objects (HTML, CSS, and JS files), and set it up correctly using a bucket policy so it can act as a host of our static website.

It's not difficult to do this, just follow these steps carefully, and your site will be up in no time.

1. Steps to create a S3 bucket and configure it as an static website.

- Sign in to the AWS Management Console and in the **Find Services** search box type S3 and choose **S3**.
- Click **Create bucket**.
  - In the **Bucket name** field, type a unique DNS-compliant name for your new bucket.

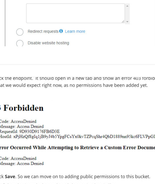
12 min
Video: Amazon Lex Walkthrough Part 2 10 min
Video: Exercise 1.1 Introduction 2 min
Reading: Exercise 1.1 Creating a Simple Bot with Lex 40 min
Quiz: Amazon Lex 5 questions
Creating a Serverless Website with S3
Video: Creating a Serverless Website with Amazon S3 2 min
Video: Creating a Serverless Website with S3 Demo 7 min
Video: Exercise 1.2 Introduction 1 min
Reading: Exercise 1.2: Creating a S3 Bucket and Configuring as a Static



- Click **Next** again.
- Under **Public access settings for this bucket** de-select the following:
  - **Block new public ACLs and uploading public objects**.
  - **Remove public access granted through public ACLs**.
  - **Block new public bucket policies**.
  - **Block public and cross-account access if bucket has public policies**.

12 min
Video: Amazon Lex Walkthrough Part 2 10 min
Video: Exercise 1.1 Introduction 2 min
Reading: Exercise 1.1 Creating a Simple Bot with Lex 40 min
Quiz: Amazon Lex 5 questions
Creating a Serverless Website with S3
Video: Creating a Serverless Website with Amazon S3 2 min
Video: Creating a Serverless Website with S3 Demo 7 min
Video: Exercise 1.2 Introduction 1 min
Reading: Exercise 1.2: Creating a S3 Bucket and Configuring as a Static

- Click on the bucket and select the **Properties** pane, choose **Static Website Hosting**.
  - Select **Use this bucket to host a website**.
  - Under **Index document** type in `index.html`. *AWS index.html*
  - Under **Error document** type in `error.html`.



12 min
Video: Amazon Lex Walkthrough Part 2 10 min
Video: Exercise 1.1 Introduction 2 min
Reading: Exercise 1.1 Creating a Simple Bot with Lex 40 min
Quiz: Amazon Lex 5 questions
Creating a Serverless Website with S3
Video: Creating a Serverless Website with Amazon S3 2 min
Video: Creating a Serverless Website with S3 Demo 7 min
Video: Exercise 1.2 Introduction 1 min
Reading: Exercise 1.2: Creating a S3 Bucket and Configuring as a Static

- Click the endpoint. It should open in a new tab and show an error 403 forbidden. Which is exactly what we would expect right now, as no permissions have been added yet.

403 Forbidden

- Code: AccessDenied
- Message: Access Denied
- RequestId: 9D01ED9178FBD02E
- HostId: uFfQ6qPgXqZBvJ4h7FgqFEcVYeBc+TZPvjRkwQkD1H0uuHFkc6FLVPpGH6Ojb+LTN7Kb6=

An Error Occurred While Attempting to Retrieve a Custom Error Document

- Code: AccessDenied
- Message: Access Denied

- Click **Save**. So we can move on to adding public permissions to this bucket.
  - Select the **Permissions** tab and choose **Bucket Policy**.
  - Paste the following code below in the bucket policy editor. Make sure to change the Resource arn with your bucket name.
  - You must change `arn:aws:s3:::2019-3-01-eb-website` to your bucket, e.g `arn:aws:s3:::2019-mm-dd-xx-website`

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AddPerm",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::2019-3-01-eb-website/*"
        }
    ]
}
```


12 min
Video: Amazon Lex Walkthrough Part 2 10 min
Video: Exercise 1.1 Introduction 2 min
Reading: Exercise 1.1 Creating a Simple Bot with Lex 40 min
Quiz: Amazon Lex 5 questions
Creating a Serverless Website with S3
Video: Creating a Serverless Website with Amazon S3 2 min
Video: Creating a Serverless Website with S3 Demo 7 min
Video: Exercise 1.2 Introduction 1 min
Reading: Exercise 1.2: Creating a S3 Bucket and Configuring as a Static

- Click **Save**.
- This bucket will now have public access as it will host our web objects.

Try it now. Go to that tab where your website is hosted (where we had the 403 forbidden message), and press refresh.

You will notice you can now see absolutely nothing. i.e you will get a 404 error.

This is because we have nothing in out bucket to show, so the browser is telling us, "hope can't find anything here". At least this proves that the bucket is now public and we did everything right so far #winning.

Now let's get the website up there.

2. Steps to upload objects to an existing S3 bucket.

- Download the ZIP file at:

```
https://s3.amazonaws.com/aws-bootcamp/dw-TM-100-SCDAWS-19-1/s3website.zip
```

- Unzip it on your local machine, and you will see the following folder structure:

12 min
Video: Amazon Lex Walkthrough Part 2 10 min
Video: Exercise 1.1 Introduction 2 min
Reading: Exercise 1.1 Creating a Simple Bot with Lex 40 min
Quiz: Amazon Lex 5 questions
Creating a Serverless Website with S3
Video: Creating a Serverless Website with Amazon S3 2 min
Video: Creating a Serverless Website with S3 Demo 7 min
Video: Exercise 1.2 Introduction 1 min
Reading: Exercise 1.2: Creating a S3 Bucket and Configuring as a Static

```
s3website/
    .gitignore
    .gulrc.js
    .npmrc
    .nvmrc
    .nvm-version
    .nvm-version-node
    .prettierrc
    .prettier.ignore
    .stylelint.rc
    .stylelint.ignore
    .editorconfig
    .eslint.rc
    .eslint.ignore
    .babel.rc
    .babel.ignore
    .travis.yml
    .travis.ignore
    .codecov.yml
    .codecov.ignore
    .gulp.js
    .gulp
```