

Project 2748776

WGS (1 Acropora pulchra sample) 

Edit title

This project was created in connection with [Request 2978645](#)

Start new project with Quick Biology Inc.

Status: **Results Delivered**

Next step: **Download results and provide feedback**

✔ **Get confirmed quote** → ✔ **Accept quote** → ✔ **Arrange payment** → ✔ **Start project** → ✔ **Get results** → Enter review

▶ **Project Notes - (click to expand)**

Last edited 25 days ago by Kristyn Liu

- Messages & Activity

Quotes

Provider

Files

Terms & Procedures

Data

Collaborators (2)

Review

Data Transfer

This project is eligible for 3 months of free and secure unlimited data storage on Genohub. We have allocated a dedicated S3 bucket in the Amazon Web Services (AWS) cloud. Here is the access information unique to this project:

- Bucket name: **genohub2748776**
- Access key ID: **AKIAXFEWT4A6TOBYPYFO**
- Secret key: **Z9jOE21AMUR79WXiln+CAVzrfz+4lIvguypGwC7m**
- Expiration: 

Mar 10, 2024

You access the storage bucket and store and retrieve data using either the AWS CLI or a graphical application like Cyberduck. If you're comfortable using the command line interface we recommend using AWS CLI. Otherwise Cyberduck is a good alternative. See the instructions below for each of these methods.

Method 1: Using Cyberduck (Mac or Windows)

Cyberduck is a free application with a graphical user interface with support for AWS S3.

**Step 1: Download and install [Cyberduck](#) if you don't already have it installed.**

After installing Cyberduck, open the application and go to **Preferences**. In the **Transfers** section go to the **Checksum** tab and check the **Verify checksum** boxes for both Downloads and Uploads. You only have to do this once. With this setting Cyberduck automatically verifies the md5 checksum as files are uploaded/downloaded.

**Step 2: Connect to the Genohub project bucket:**

- In Cyberduck click the **Open Connection** button to see the connection dialog box
- From the first dropdown select **Amazon S3**
- For **Server** enter **genohub2748776.s3.amazonaws.com**

Copy
- Leave **Port** as is (443)
- For **Access Key ID** enter the project access key: **AKIAXFEWT4A6TOBYPYFO**

Copy
- For **Secret Access Key** enter the project secret key: **Z9jOE21AMUR79WXiln+CAVzrfz+4lIvguypGwC7m**

Copy
- Click the **Connect** button

**Step 3: Simply drag and drop files or folders into or out of Cyberduck**

Method 2: Using the AWS Command Line Interface (Mac, Windows or Linux)

Step 1: Install the AWS CLI

If you already have the AWS CLI (Command Line Interface) installed you can skip this step. Otherwise follow [these instructions to install AWS CLI version 2](#).

Step 2: Configure AWS CLI

In a terminal window run this command:

**aws configure**

For **AWS Access Key ID**, enter: **AKIAXFEWT4A6TOBYPYFO**

Copy

For **AWS Secret Access Key**, enter: **Z9jOE21AMUR79WXiln+CAVzrfz+4lIvguypGwC7m**

Copy

For **Default Region Mame** enter **us-east-1**

Leave **Default output format** blank (None)

### Step 3: Access the project bucket to upload or download data

#### Viewing bucket contents

To see the contents of the project bucket:

```
aws s3 ls s3://genohub2748776 --recursive
```

#### Downloading files

To download everything from the project bucket to your local directory:

```
aws s3 sync s3://genohub2748776 .
```

To download only a specific file to your local directory (replace *[filename]* with the actual file name):

```
aws s3 cp s3://genohub2748776/[filename] .
```

#### Uploading files

To upload data to the project bucket:

Assuming all of the data related to this project is stored in a local directory you can use the 'sync' command from inside the directory to upload all its files and subdirectories to the project bucket. It's smart enough to upload only new or updated files:

```
aws s3 sync . s3://genohub2748776
```

To upload a specific file from your local directory to the project bucket:

```
aws s3 cp [filename] s3://genohub2748776 (replace [filename] with the actual file name)
```

If you want to place a one-off file in a directory simply include the directory name in the target path and a directory with that name will automatically be created if it doesn't already exist. For example:

```
aws s3 cp [filename] s3://genohub2748776/[directory_name]/[filename] (replace [filename] and [directory_name] with the actual file name and directory name respectively)
```

#### Deleting files from the bucket

Please be careful about deleting files because once a file is deleted from the project bucket it can't be recovered.

To delete a specific file:

```
aws s3 rm s3://genohub2748776/[filename]
```

To delete files matching a pattern you would need to use the `--include` and `--exclude` flags. See [this page](#) for more information about the include/exclude filters. Once files are deleted they can't be recovered so we recommend running the `rm` command using the `--dryrun` flag first, which will show you the list of files that would be deleted, without actually deleting anything. If everything looks good then rerun the command without the `--dryrun` flag to actually perform the deletion.

For example to delete all files ending in **fastq.gz** run this command first to see what files will be deleted:

```
aws s3 rm s3://genohub2748776 --recursive --dryrun --exclude "*" --include "*.fastq.gz"
```

If you're sure you want to delete those files, rerun the same command without the `--dryrun` flag to actually delete them:

```
aws s3 rm s3://genohub2748776 --recursive --exclude "*" --include "*.fastq.gz"
```

#### More information

For more specific use cases see the [AWS S3 Command Reference](#). If you need help, post a message to the project and we'll be happy to assist.

