

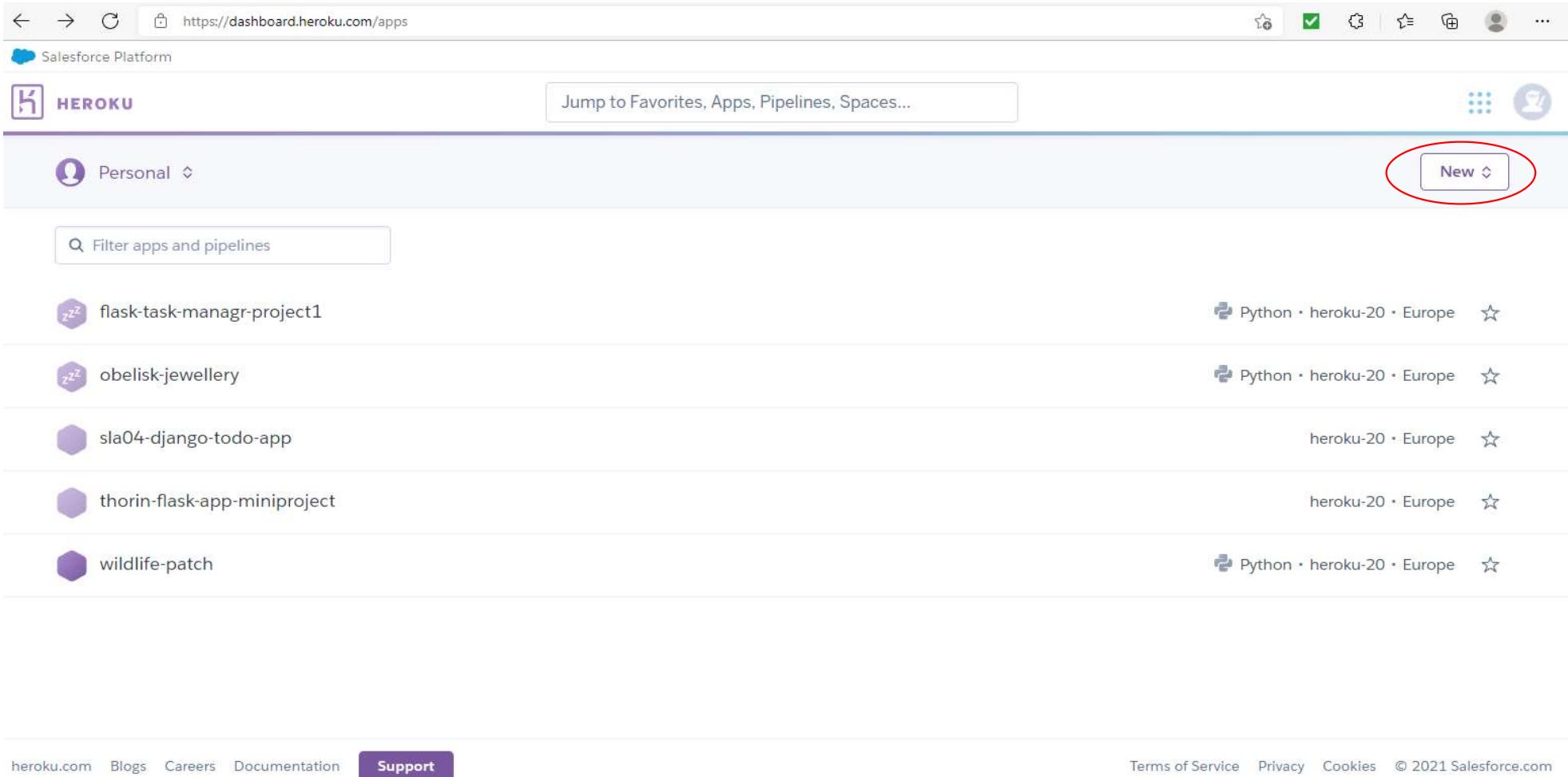
# Deployment

Deployment of the project will make use of Heroku and Amazon Web Services (AWS). Heroku provides an isolated virtual server on which to run the application but AWS also offers a whole host of cloud-based services.

All the static files like JavaScript and CSS, plus the product images will be stored in s3 (simple storage system).

## Part 1

### Click 'New' when logged into Heroku



The screenshot shows the Heroku dashboard interface. At the top, there is a header bar with various icons and a URL bar showing <https://dashboard.heroku.com/apps>. Below the header is a navigation bar with the 'Salesforce Platform' logo, the 'HEROKU' logo, and a search bar labeled 'Jump to Favorites, Apps, Pipelines, Spaces...'. On the left, there is a 'Personal' dropdown menu and a search bar for 'Filter apps and pipelines'. The main area displays a list of apps:

App Name	Language	Region	Star Rating
flask-task-managr-project1	Python	heroku-20 • Europe	☆
obelisk-jewellery	Python	heroku-20 • Europe	☆
sla04-django-todo-app		heroku-20 • Europe	☆
thorin-flask-app-miniproject		heroku-20 • Europe	☆
wildlife-patch	Python	heroku-20 • Europe	☆

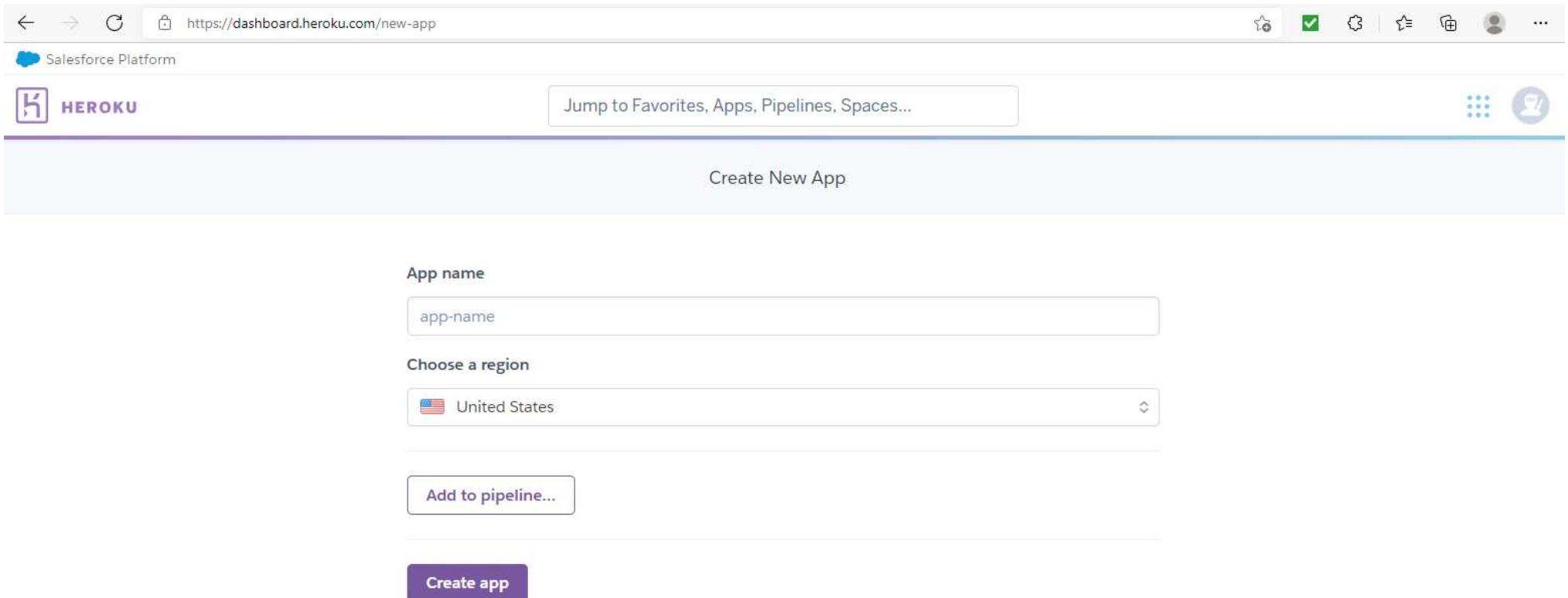
In the top right corner of the dashboard, there is a 'New' button, which is circled in red to indicate it should be clicked. At the bottom of the page, there are links for 'heroku.com', 'Blogs', 'Careers', 'Documentation', 'Support', 'Terms of Service', 'Privacy', 'Cookies', and a copyright notice for '© 2021 Salesforce.com'.

# Click 'Create new app'

The screenshot shows the Heroku dashboard at <https://dashboard.heroku.com/apps>. The top navigation bar includes links for Salesforce Platform, Personal (dropdown), and a search bar labeled 'Jump to Favorites, Apps, Pipelines, Spaces...'. On the right, there's a 'New' dropdown menu with two options: 'Create new app' (highlighted with a red circle) and 'Create new pipeline'. Below this, a list of apps is displayed:

App Name	Language	Region	Favorites
flask-task-managr-project1	Python	heroku-20 • Europe	☆
obelisk-jewellery	Python	heroku-20 • Europe	☆
sla04-django-todo-app		heroku-20 • Europe	☆
thorin-flask-app-miniproject		heroku-20 • Europe	☆
wildlife-patch	Python	heroku-20 • Europe	☆

# Input app name, select region and then click ‘Create app’



The screenshot shows the Heroku dashboard at <https://dashboard.heroku.com/new-app>. The page title is "Create New App". The "App name" field contains "app-name". The "Choose a region" dropdown is set to "United States". There is a "Add to pipeline..." button. A prominent purple "Create app" button is at the bottom.

App name  
app-name

Choose a region  
United States

Add to pipeline...

Create app

# Click on ‘Resources’

The screenshot shows the Heroku dashboard for the app 'obelisk-jewellery'. The 'Resources' tab is highlighted with a red circle. In the 'Add-ons' section, the 'Heroku Postgres' add-on is listed with a red circle around it. A red arrow points from the text 'Type in ‘Postgres’ here, then Heroku will find Postgres' to the search bar in the 'Add-ons' section.

https://dashboard.heroku.com/apps/obelisk-jewellery/resources

Salesforce Platform

HEROKU

Jump to Favorites, Apps, Pipelines, Spaces...

Personal > obelisk-jewellery

GitHub sanson0/obelisk\_jewellery main

Overview Resources Deploy Metrics Activity Access Settings

Free Dynos Change Dyno Type

web gunicorn obelisk\_jewellery.wsgi:application \$0.00

Add-ons

Type in ‘Postgres’ here, then Heroku will find Postgres

Find more add-ons

Quickly add add-ons from Elements

Heroku Postgres Attached as DATABASE Hobby Dev Free \$0.00

Estimated Monthly Cost

heroku.com Blogs Careers Documentation Support Terms of Service Privacy Cookies © 2021 Salesforce.com

Select the free plan for this project

## Go to Gitpod

To use Postgres we need to go back to gitpod and install dj\_database\_url, and psycopg2.

Type ‘pip3 install dj\_database\_url’ and ‘pip3 install psycopg2-binary’

Then freeze the requirements with pip3  
freeze > requirements.txt

This ensures Heroku installs all apps requirements during deployment.

Look in  
requirements.txt

```
1  asgiref==3.2.3
2  boto3==1.12.42
3  botocore==1.15.42
4  chardet==3.0.4
5  dj-database-url==0.5.0
6  django==3.0.1
7  django-allauth==0.41.0
8  django-countries==6.0
9  django-crispy-forms==1.8.1
10 django-storages==1.9.1
11 docutils==0.15.2
12 gunicorn==20.0.4
13 idna==2.8
14 jmespath==0.9.5
15 oauthlib==3.1.0
16 Pillow==7.0.0
17 psycopg2-binary==2.8.5
18 python3-openid==3.1.0
19 pytz==2019.3
20 requests==2.22.0
21 requests-oauthlib==1.3.0
22 s3transfer==0.3.3
23 sqlparse==0.3.0
24 stripe==2.42.0
25 urllib3==1.25.7
```

Database setup.

Import dj\_database\_url in settings.py.

Then down in the databases setting, comment out the default configuration.

And replace the default database with a call to dj\_database\_url.parse  
And give it the database URL from Heroku.

Which you can either get from your config variables in your app settings tab.

Or from the command line by typing Heroku config.

```
Requirements.txt      * settings.py *
114 LOGIN_REDIRECT_URL = '/'
115
116 WSGI_APPLICATION = 'boutique_ado.wsgi.application'
117
118
119 # Database
120 # https://docs.djangoproject.com/en/3.0/ref/settings/#databases
121
122 # DATABASES = {
123 #     'default': {
124 #         'ENGINE': 'django.db.backends.sqlite',
125 #         'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),
126 #     }
127 # }
128
129 DATABASES = {
130     'default': dj_database_url.parse('postgres://zzwe3rguhfnfs:88814a80171b74f4e012fb2e00d02a81f
131 }
132
133
134 # Password validation
135 # https://docs.djangoproject.com/en/3.0/ref/settings/#auth-passwordValidators
136
137 AUTH_PASSWORD_VALIDATORS = [
138     {
139         'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
140     },
141     {
142         'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
143     },
144     {
145         'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
146     },
147     {
148         'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
149     },
150 }
```

## Changing the database

Next, connect to our new Heroku database and run migrations, but just before that happens, product information from the database SQLite needs to be transferred to the new database Postgres.

Part of the deployment process involves using a Postgres database instead of sqlite3. All the products were manually loaded into the sqlite3 database so swapping databases will cause the product data to disappear!

Follow this set of instructions if the product data has been manually loaded: -

1. Make sure your manage.py file is connected to your mysql database
2. Use this command to backup your current database and load it into a db.json file:

```
Python3 manage.py dumpdata --exclude auth.permission --  
exclude contenttypes > db.json
```

3. Connect your manage.py file to your postgres database
4. Then use this command to load your data from the db.json file into postgres:

```
Python3 manage.py loaddata db.json
```

Now connecting to Postgres, run all the migrations again

Run Python 3 manage.py migrate.

It will apply all those migrations and get the database all set up.

Use python3 manage.py createsuperuser to set up login details

Then Heroku app and database are ready to go.

```
119 # Database
120 # https://docs.djangoproject.com/en/3.0/ref/settings/#databases
121
122 DATABASES = {
123     'default': {
124         'ENGINE': 'django.db.backends.sqlite',
125         'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),
126     }
127 }
128
129
130 # Password validation
131 # https://docs.djangoproject.com/en/3.0/ref/settings/#auth-password-validators
132
133 AUTH_PASSWORD_VALIDATORS = [
134     {
135         'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
136     },
137 ]
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
389
```

Remove the Heroku database config. and uncomment the original so the database URL doesn't end up in version control. Add, commit and push code to Github

## Part 2

Set up an if statement in settings.py.

When the app is running on Heroku where the database URL environment variable will be defined, connect to Postgres and otherwise, connect to SQLite.



```
110 ACCOUNT_USERNAME_MIN_LENGTH = 4
111 LOGIN_URL = '/accounts/login/'
112 LOGIN_REDIRECT_URL = '/'
113
114 WSGI_APPLICATION = 'obelisk_jewellery.wsgi.application'
115
116
117 # Database
118 # https://docs.djangoproject.com/en/3.2/ref/settings/#databases
119
120
121 if 'DATABASE_URL' in os.environ:
122     DATABASES = {
123         'default': dj_database_url.parse(os.environ.get('DATABASE_URL'))
124     }
125 else:
126     DATABASES = {
127         'default': {
128             'ENGINE': 'django.db.backends.sqlite3',
129             'NAME': BASE_DIR / 'db.sqlite3',
130         }
131     }
132
133
134
135
136
137 }
```

Next, install Gunicorn, which will act as our webserver, type in Gitpod: -  
pip3 install gunicorn

and freeze that into our requirements file, pip3 freeze > requirements.txt

Now create a Procfile, to tell Heroku to create a web dyno, which will run Gunicorn and serve our Django app.



A screenshot of a GitHub commit page. The commit message is "sanson0 Fixed deployment bug". It shows 1 contributor and 1 line of code added to the Procfile.



1 lines (1 sloc) | 48 Bytes

```
1 web: gunicorn obelisk_jewellery.wsgi:application
```

Add this code to Procfile

In Heroku config vars, disable collectstatic equals 1  
So that Heroku won't try to collect static files when we deploy.

The screenshot shows the Heroku dashboard for the app "obelisk-jewellery".

**GitHub repo:** sanson0/obelisk\_jewellery

**Heroku git URL:** https://git.heroku.com/obelisk-jewellery.git

**Config Vars:**

KEY	VALUE	Setting
DATABASE_URL	postgres://	Hidden
DISABLE_COLLECTSTATIC	1	Hidden
SECRET_KEY		Hidden
KEY	VALUE	Add

A red oval highlights the row for `DISABLE_COLLECTSTATIC`, which has a value of `1`. This indicates that static files will not be collected during deployment.

**Buildpacks:**

Add buildpack

Add the hostname of the Heroku app to allowed hosts in settings.py, and add localhost in here so that gitpod will still work too.

```
16 # Build paths inside the project like this: BASE_DIR / 'subdir'.
17 BASE_DIR = Path(__file__).resolve().parent.parent
18
19
20 # Quick-start development settings - unsuitable for production
21 # See https://docs.djangoproject.com/en/3.2/howto/deployment/checklist/
22
23 # SECURITY WARNING: keep the secret key used in production secret!
24 SECRET_KEY = os.environ.get('SECRET_KEY', '')
25
26 # SECURITY WARNING: don't run with debug turned on in production!
27 DEBUG = 'DEVELOPMENT' in os.environ
28
29 ALLOWED_HOSTS = ['obelisk-jewellery.herokuapp.com', 'localhost']
30
31
32 # Application definition
33
34 INSTALLED_APPS = [
35     'django.contrib.admin',
36     'django.contrib.auth',
37     'django.contrib.contenttypes',
38     'django.contrib.sessions',
39     'django.contrib.messages',
40     'django.contrib.staticfiles',
41     'django.contrib.sites',
42     'allauth'.
```

Go to Gitpod.

Deploy the app, by adding and committing all changes, and pushing to Github with git push.

Use git push Heroku master (or main if main branch is utilised) to deploy to Heroku.

Then the app is deployed, without any static files for now.

Set Heroku up to automatically deploy when code is pushed to Github.  
Go to the app in Heroku and on the deploy tab, set connect to Github.

The screenshot shows the Heroku dashboard for the app 'obelisk-jewellery'. The 'Deploy' tab is selected and highlighted with a red circle. On the left, there's a section for adding the app to a pipeline, with instructions to create a new pipeline or choose an existing one. On the right, there's a section for adding the app to a stage in a pipeline, with information about pipelines connecting multiple apps and enabling review apps. Below these sections, there's a dropdown menu labeled 'Choose a pipeline'. At the bottom, there's a 'Deployment method' section with three options: 'Heroku Git' (selected), 'GitHub Connected' (highlighted with a red circle), and 'Container Registry'. A note below says 'Search for required repository. And then click connect.'

https://dashboard.heroku.com/apps/obelisk-jewellery/deploy/github

Salesforce Platform

HEROKU

Personal > obelisk-jewellery

GitHub sanson@/obelisk\_jewellery main

Overview Resources Deploy Metrics Activity Access Settings

Add this app to a pipeline

Create a new pipeline or choose an existing one and add this app to a stage in it.

Pipelines let you connect multiple apps together and **promote code** between them. [Learn more.](#)

Choose a pipeline

Add this app to a stage in a pipeline to enable additional features

Pipelines connected to GitHub can enable **review apps**, and create apps for new pull requests. [Learn more.](#)

Deployment method

Heroku Git  
Use Heroku CLI

GitHub Connected

Container Registry  
Use Heroku CLI

Search for required repository.  
And then click connect.

The screenshot shows the Heroku Dashboard for the app 'ck28780-boutique-ado'. The 'Deploy' tab is selected. At the top, there are sections for connecting to a pipeline and GitHub. Below these, a 'Choose a pipeline' dropdown is shown. Under 'Deployment method', there are buttons for 'Heroku Git' (using the Heroku CLI) and 'GitHub' (Connect to GitHub). A 'Search for a repository to connect to' input field contains 'ck28780/boutique'. A note at the bottom states: 'Missing a GitHub organization? Ensure Heroku Dashboard has team access.'

Click ‘connect’ and ‘Enable automatic deploys’. This allows deployment of changes on each push to github

The screenshot shows the Heroku dashboard for the app 'obelisk-jewellery'. At the top, it says 'App connected to GitHub' and 'Code diffs, manual and auto deploys are available for this app.' Below this, there's a section titled 'Automatic deploys' which says 'Enables a chosen branch to be automatically deployed to this app.' A note in a box says: 'You can now change your main deploy branch from "master" to "main" for both manual and automatic deploys, please follow the instructions [here](#).' Under 'Automatic deploys', there's a checked checkbox for 'Automatic deploys from `main`' and a note: 'Every push to `main` will deploy a new version of this app. Deployes happen automatically: be sure that this branch in GitHub is always in a deployable state and any tests have passed before you push. [Learn more](#).' There's also an unchecked checkbox for 'Wait for CI to pass before deploy' with the note: 'Only enable this option if you have a Continuous Integration service configured on your repo.' A 'Disable Automatic Deploys' button is at the bottom.

App connected to GitHub  
Code diffs, manual and auto deploys are available for this app.

Automatic deploys  
Enables a chosen branch to be automatically deployed to this app.

You can now change your main deploy branch from "master" to "main" for both manual and automatic deploys, please follow the instructions [here](#).

Automatic deploys from `main` are enabled  
Every push to `main` will deploy a new version of this app. Deployes happen automatically: be sure that this branch in GitHub is always in a deployable state and any tests have passed before you push. [Learn more](#).

Wait for CI to pass before deploy  
Only enable this option if you have a Continuous Integration service configured on your repo.

[Disable Automatic Deploys](#)

With that finished automatic deploys are enabled.  
And now every time code is push to github, the project will automatically be deployed to Heroku as well.

Next look up a Django secret key generator. Generate a key, and add it to my config variables in Heroku. This will be the key for the Heroku app.

<https://miniwebtool.com/Django-secret-key-generator>

Django Secret Key Generator

Secret key prefix (optional):

Secret key suffix (optional):

**Generate Django Secret Key**

Port Harcourt is waiting for you:  
Book your flight now.

Lufthansa

Generated Django Secret Key  
Length: 50

Hidden

Miniwebtool

Link to This Tool

Recommend This Tool

Upgrade to Premium

My Toolbox

HSBC UK

Find your mortgage deal today with HSBC UK

Learn more

Need to pick a winner for raffles, contests, drawings, giveaways, promotions or even chores?

With that created, go back to settings.py  
And replace the secret key setting with the call to get it from the environment and use an empty string as a default.

```
19
20  # Quick-start development settings - unsuitable for production
21  # See https://docs.djangoproject.com/en/3.2/howto/deployment/checklist/
22
23  # SECURITY WARNING: keep the secret key used in production secret!
24  SECRET_KEY = os.environ.get('SECRET_KEY', '')
25
26  # SECURITY WARNING: don't run with debug turned on in production!
27  DEBUG = 'DEVELOPMENT' in os.environ
28
29  ALLOWED_HOSTS = ['obelisk-jewellery.herokuapp.com', 'localhost']
30
31
32  # Application definition
33
34  INSTALLED_APPS = [
35      'django.contrib.admin',
36      'django.contrib.auth',
37      ...
38  ]
```

Now commit these changes and push them to github.  
Navigating to Heroku we can see there's a build in progress  
and automatic deployments are working.

The screenshot shows the Heroku dashboard for the app 'obelisk-jewellery'. At the top, there's a banner for Heroku Pipelines. Below it, the 'Installed add-ons' section shows a single instance of Heroku Postgres. The 'Dyno formation' section indicates the app is using free dynos. In the 'Collaborator activity' section, it shows 2 deploys by 's04anson@gmail.com'. The main area displays the latest activity log:

- s04anson@gmail.com: Build in progress Just now · [View build progress](#)
- s04anson@gmail.com: Set SECRET\_KEY config var Today at 10:08 PM · v8
- s04anson@gmail.com: Deployed d65e0c28 Today at 9:51 PM · v7 · [Compare diff](#)
- s04anson@gmail.com: Build succeeded Today at 9:50 PM · [View build log](#)
- s04anson@gmail.com: Deployed f49958ce Yesterday at 3:20 PM · v6

← → ⌂ ⌄ https://dashboard.heroku.com/apps/obelisk-jewellery/activity

Salesforce Platform

HEROKU Jump to Favorites, Apps, Pipelines, Spaces...

Personal > obelisk-jewellery

GitHub sanson0/obelix\_jewellery main

Overview Resources Deploy Metrics Activity Access Settings

Activity Feed

  s04anson@gmail.com: Deployed 2536ab83  
Today at 3:34 PM · v14 · [Compare diff](#)

  s04anson@gmail.com: Build succeeded  
Today at 3:33 PM · [View build log](#)

  s04anson@gmail.com: Remove DISABLE\_COLLECTSTATIC config var  
Today at 3:15 PM · v13 · [Roll back to here](#)

  s04anson@gmail.com: Set USE\_AWS config var  
Today at 3:13 PM · v12 · [Roll back to here](#)

  s04anson@gmail.com: Set AWS\_SECRET\_ACCESS\_KEY config var  
Today at 3:12 PM · v11 · [Roll back to here](#)

  s04anson@gmail.com: Set AWS\_ACCESS\_KEY\_ID config var  
Today at 3:11 PM · v10 · [Roll back to here](#)

## Part 3

### Create account in AWS



The screenshot shows the AWS homepage at <https://aws.amazon.com>. The top navigation bar includes links for Contact Us, Support, English, My Account, Sign In, and a prominent orange "Create an AWS Account" button, which is circled in red. Below the header, the main heading reads "Start Building on AWS Today" with a subtext about building sophisticated applications. A large orange "Get Started for Free" button is centered. The page is divided into two main sections: "For Builders" and "For Decision Makers". The "For Builders" section features icons for launching an application and learning, along with descriptions for both. The "For Decision Makers" section features icons for optimizing business value and reinventing with data, also with accompanying descriptions.

Contact Us   Support ▾   English ▾   My Account ▾   Sign In   **Create an AWS Account**

Products   Solutions   Pricing   Documentation   Learn   Partner Network   AWS Marketplace   Customer Enablement   Events   Explore More  

## Start Building on AWS Today

Whether you're looking for compute power, database storage, content delivery, or other functionality, AWS has the services to help you build sophisticated applications with increased flexibility, scalability and reliability

**Get Started for Free**

**For Builders**  
For developers or anyone interested in learning how to build on AWS today

 [Launch Your First Application](#)  
Start building with short step-by-step tutorials

 [Learn, Build & Get Connected](#)  
Get news, download tools, and find an AWS User Group near you

**For Decision Makers**  
For organization leaders to enable innovation and transformation

 [Optimize Business Value](#)  
Reduce costs and improve agility, productivity, and resilience

 [Reinvent with Data](#)  
Unlock growth opportunities, make better decisions, and innovate faster



English ▾



## Congratulations

Thank you for signing up for AWS.

We are activating your account, which should only take a few minutes. You will receive an email when this is complete.

[Go to the AWS Management Console](#)

[Sign up for another account](#) or [contact sales](#).

# Sign in by accessing the AWS management console under ‘My account’

https://eu-west-2.console.aws.amazon.com/console/home?region=eu-west-2#

AWS Management Console

**AWS services**

▼ Recently visited services  
Your recently visited AWS services appear here.

► All services

**Build a solution**  
Get started with simple wizards and automated workflows.

**Launch a virtual machine**  
With EC2  
2-3 minutes

**Build a web app**  
With Elastic Beanstalk  
6 minutes

**Stay connected to your AWS resources on-the-go**

 AWS Console Mobile App now supports four additional regions. Download the AWS Console Mobile App to your iOS or Android mobile device. [Learn more](#)

**Explore AWS**

**AWS Training**  
Free digital courses to help you develop your skills.  
[Learn more](#)

**AWS Certification**

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Type S3 into the search bar and click on 'S3'

The screenshot shows the AWS console search results for the query 's3'. The search bar at the top contains the text 's3'. Below the search bar, the results are categorized into 'Services' and 'Features'.

**Services** (7)

- S3** Scalable Storage in the Cloud
- S3 Glacier** Archive Storage in the Cloud
- Athena** Query Data in S3 using SQL
- AWS Snow Family** Large Scale Data Transport

**Features**

- Amazon S3 File Gateway**

Each service entry includes a small icon and a brief description. A red circle highlights the 'S3' service entry in the 'Services' list.

# Click 'Create bucket'

The screenshot shows the AWS S3 console interface. On the left, there's a sidebar with various options like 'Buckets', 'Access Points', and 'Storage Lens'. The main area is titled 'Amazon S3' and shows an 'Account snapshot' with a 'View Storage Lens dashboard' button. Below that is a section for 'Buckets (1) Info' with a table showing one bucket named 'obelisk-jewellery' in the EU (London) region with public access. At the top of this section is a row of buttons: a grey 'C' button, a 'Copy ARN' button, an 'Empty' button, a 'Delete' button, and a prominent orange 'Create bucket' button, which is circled in red. There's also a search bar labeled 'Find buckets by name' and some pagination controls.

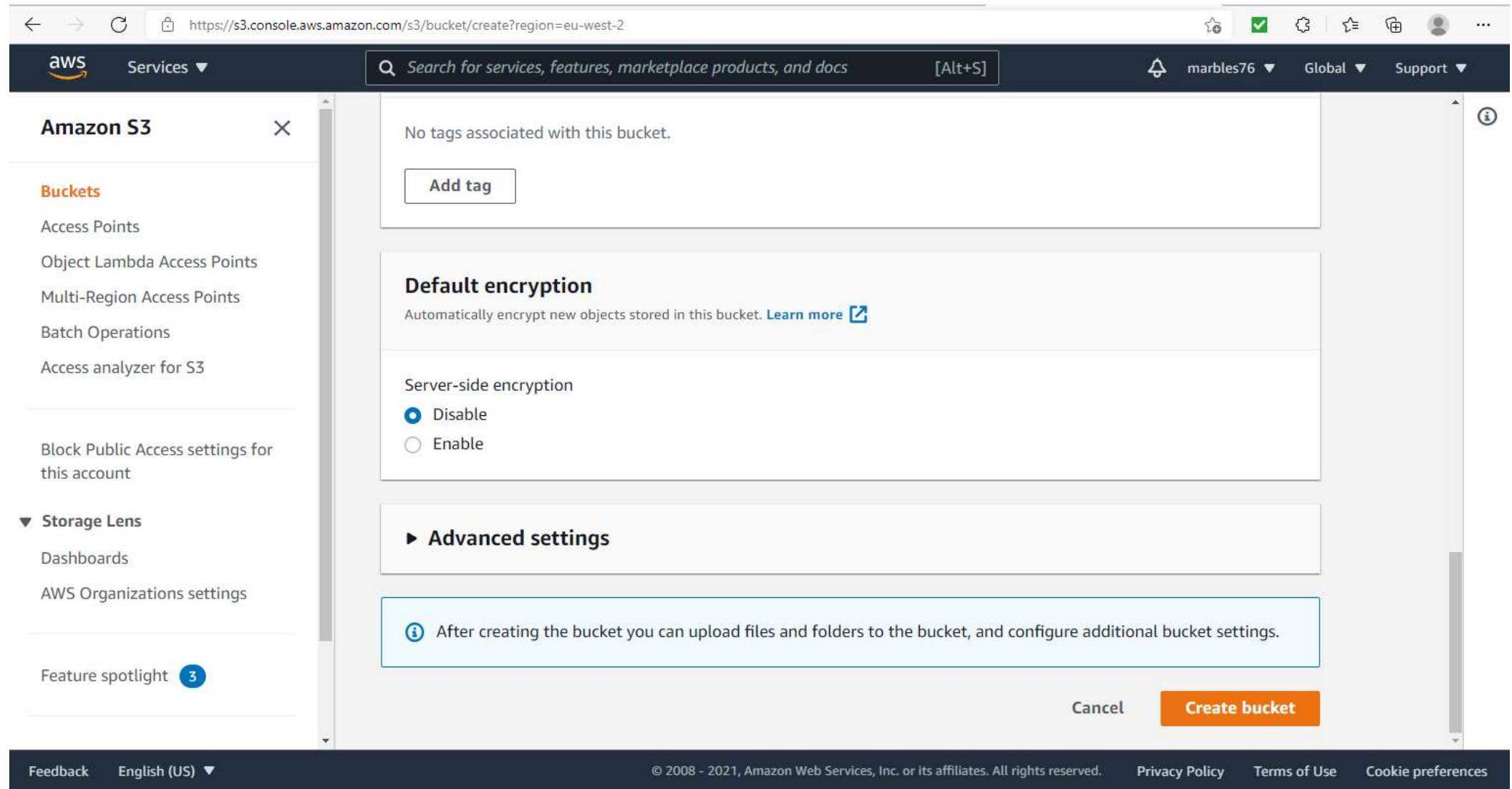
Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Set name and region, uncheck ‘Block all public access’

The screenshot shows the AWS S3 'Create bucket' interface. In the 'General configuration' section, the 'Bucket name' field contains 'myawsbucket'. Below it, a note states: 'Bucket name must be unique and must not contain spaces or uppercase letters. See rules for bucket naming'. The 'AWS Region' dropdown is set to 'EU (London) eu-west-2'. Under 'Copy settings from existing bucket - optional', there is a note: 'Only the bucket settings in the following configuration are copied.' and a 'Choose bucket' button. At the bottom of the page, under 'Block Public Access settings for this bucket', there is a checkbox that is currently checked.

Acknowledge that the bucket will be public, since this bucket will need to be public in order to allow public access to our static files.

# Click 'Create bucket'



The screenshot shows the 'Create bucket' wizard in the AWS S3 console. The left sidebar lists navigation options like 'Buckets', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', 'Access analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens', 'Dashboards', 'AWS Organizations settings', and 'Feature spotlight'. The main area displays configuration options: 'No tags associated with this bucket.' with an 'Add tag' button; 'Default encryption' (disabled); 'Server-side encryption' (disabled); 'Advanced settings' (indicated by a plus sign); and a note: 'After creating the bucket you can upload files and folders to the bucket, and configure additional bucket settings.' At the bottom right are 'Cancel' and 'Create bucket' buttons.

No tags associated with this bucket.

Add tag

**Default encryption**  
Automatically encrypt new objects stored in this bucket. [Learn more](#)

Server-side encryption

Disable  
 Enable

► Advanced settings

After creating the bucket you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel **Create bucket**

# Click on bucket name

The screenshot shows the AWS S3 console interface. On the left, a sidebar menu is open under the 'Amazon S3' heading, showing options like 'Buckets', 'Access Points', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', 'Access analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens', 'Dashboards', and 'AWS Organizations settings'. Below this is a 'Feature spotlight' section with a '3' badge. The main content area has a search bar at the top. A large red oval highlights the first row of the 'Buckets' table, which contains the following data:

Name	AWS Region	Access	Creation date
obelisk-jewellery	EU (London) eu-west-2	Public	October 17, 2021, 16:35:22 (UTC+01:00)

At the bottom of the page, there are links for 'Feedback', 'English (US)', 'Privacy Policy', 'Terms of Use', and 'Cookie preferences'.

# Click on ‘Properties’ tab

The screenshot shows the Amazon S3 console interface. On the left, there's a sidebar with 'Amazon S3' at the top, followed by sections for 'Buckets', 'Storage Lens', and 'Feature spotlight'. The main area is titled 'obelisk-jewellery' and shows that the bucket is 'Publicly accessible'. Below this, there are tabs: 'Objects' (highlighted with a red circle), 'Properties' (underlined), 'Permissions', 'Metrics', 'Management', and 'Access Points'. The 'Properties' tab is currently active. Underneath, there's a section for 'Objects (2)'. It includes buttons for 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', and 'Actions'. There's also a 'Create folder' button and an 'Upload' button. A search bar says 'Find objects by prefix' with a value of '1'. At the bottom, there's a table with columns: Name, Type, Last modified, Size, and Storage class. Two entries are listed: 'media/' (Folder) and 'static/' (Folder). The footer contains links for 'Feedback', 'English (US)', 'Privacy Policy', 'Terms of Use', and 'Cookie preferences'.

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Click on ‘Use this bucket to host a website’ Also, fill in index and error documents with index.html and error.html, then click ‘Save’.

The screenshot shows the AWS S3 Bucket Properties page for the bucket 'obelisk-jewellery'. The left sidebar lists 'Buckets' and other options like 'Storage Lens'. The main content area has two sections: 'Requester pays' and 'Static website hosting'. Under 'Requester pays', it shows 'Requester pays' is set to 'Disabled'. Under 'Static website hosting', it shows 'Static website hosting' is 'Enabled', 'Hosting type' is 'Bucket hosting', and the 'Bucket website endpoint' is listed as <http://obelisk-jewellery.s3-website.eu-west-2.amazonaws.com>. There are 'Edit' buttons for both sections.

Amazon S3

Buckets

- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- Access analyzer for S3

Block Public Access settings for this account

▼ Storage Lens

- Dashboards
- AWS Organizations settings

Feature spotlight 3

Requester pays

When enabled, the requester pays for requests and data transfer costs, and anonymous access to this bucket is disabled. [Learn more](#)

Requester pays

Disabled

Edit

Static website hosting

Use this bucket to host a website or redirect requests. [Learn more](#)

Static website hosting

Enabled

Hosting type

Bucket hosting

Bucket website endpoint

When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. [Learn more](#)

<http://obelisk-jewellery.s3-website.eu-west-2.amazonaws.com>

Feedback English (US) ▾

© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Privacy Policy Terms of Use Cookie preferences

# Click on the ‘Permissions’ tab

The screenshot shows the AWS S3 console interface. In the top navigation bar, the URL is https://s3.console.aws.amazon.com/s3/buckets/obelisk-jewellery?region=eu-west-2&tab=permissions. The main content area displays the bucket 'obelisk-jewellery'. A red circle highlights the 'Permissions' tab in the top navigation bar of the bucket's details page. Below the navigation bar, there are sections for 'Permissions overview' (with a 'Public' access status) and 'Block public access (bucket settings)', which includes a note about granting public access through various methods like ACLs and bucket policies.

https://s3.console.aws.amazon.com/s3/buckets/obelisk-jewellery?region=eu-west-2&tab=permissions

aws Services ▾ Search for services, features, marketplace products, and docs [Alt+S] marbles76 ▾ Global ▾ Support ▾

Amazon S3 obelisk-jewellery

obelisk-jewellery [Info](#)

Publicly accessible

Objects Properties Permissions Metrics Management Access Points

Permissions overview

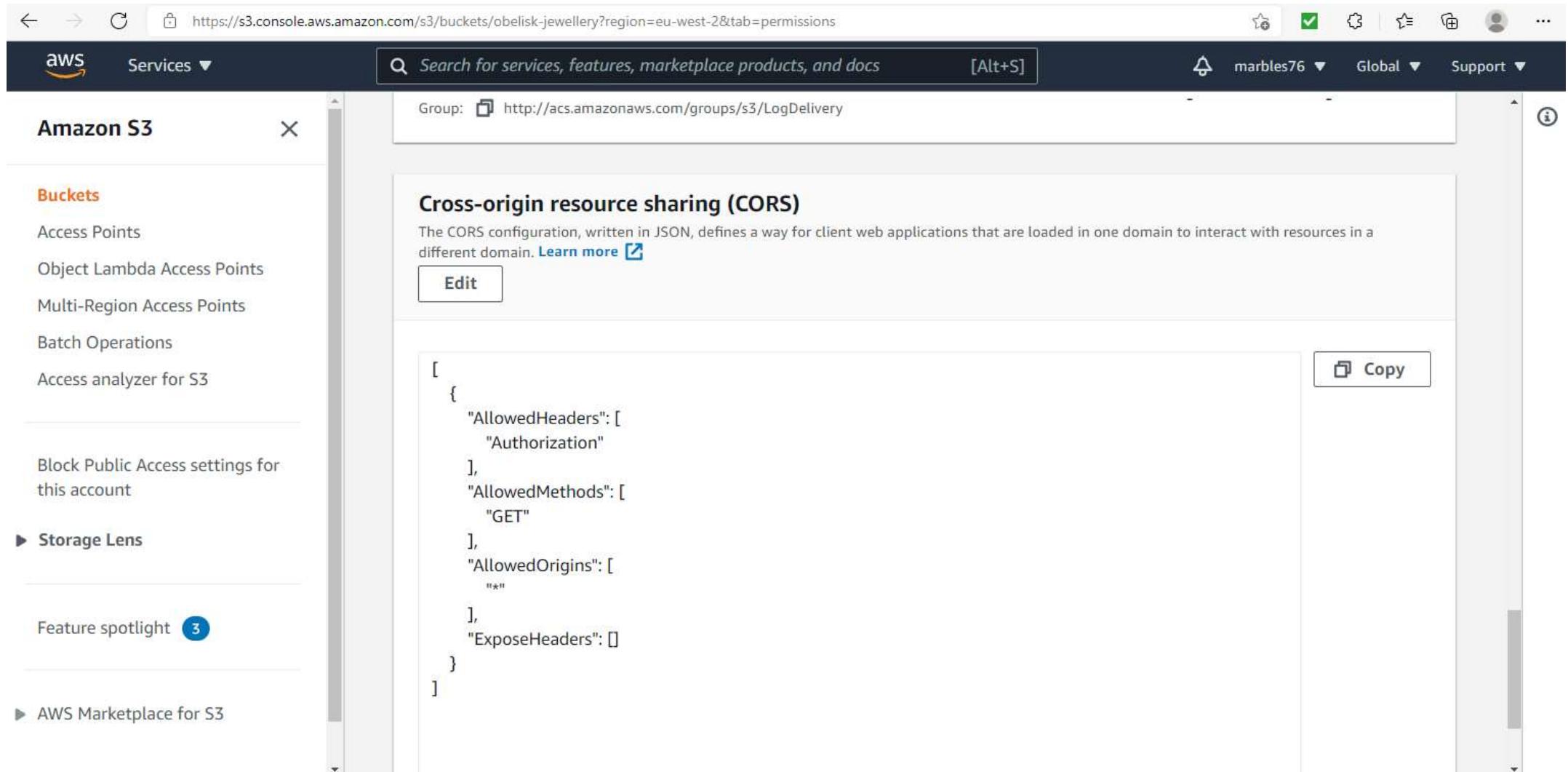
Access ⚠ Public

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Scroll down to CORS section and use this code



The screenshot shows the AWS S3 console with the CORS configuration for a bucket named "obelisk-jewellery". The CORS configuration is set to a group named "LogDelivery". The JSON code for the CORS configuration is displayed:

```
[{"AllowedHeaders": ["Authorization"], "AllowedMethods": ["GET"], "AllowedOrigins": ["*"], "ExposeHeaders": []}]
```

A "Copy" button is available to copy the JSON code.

**Cross-origin resource sharing (CORS)**  
The CORS configuration, written in JSON, defines a way for client web applications that are loaded in one domain to interact with resources in a different domain. [Learn more](#)

**Edit**

**Amazon S3**

**Buckets**

- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- Access analyzer for S3

Block Public Access settings for this account

▶ Storage Lens

Feature spotlight 3

▶ AWS Marketplace for S3

Search for services, features, marketplace products, and docs

[Alt+S]



marbles76 ▾

Global ▾

Support ▾

## Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts.

[Learn more](#)

Edit

Delete

Copy

# Select policy generator

Action is 'get object'

**AWS Policy Generator**  
The AWS Policy Generator is a tool that enables you to create policies that control access to Amazon Web Services (AWS) products and resources. For more information about creating policies, see [key concepts in Using AWS Identity and Access Management](#). Here are sample policies.

**Step 1: Select Policy Type**  
A Policy is a container for permissions. The different types of policies you can create are an [IAM Policy](#), an [S3 Bucket Policy](#), an [SNS Topic Policy](#), a [VPC Endpoint Policy](#), and an [SQS Queue Policy](#).

Select Type of Policy

**S3 Bucket Policy**

**Step 2: Add Statement(s)**  
A statement is the formal description of a single permission. See [a description of elements](#) that you can use in statements.

Effect  Allow  Deny

Principal  Use a comma to separate multiple values.

AWS Service   All Services ('\*')

Actions   All Actions ('\*')

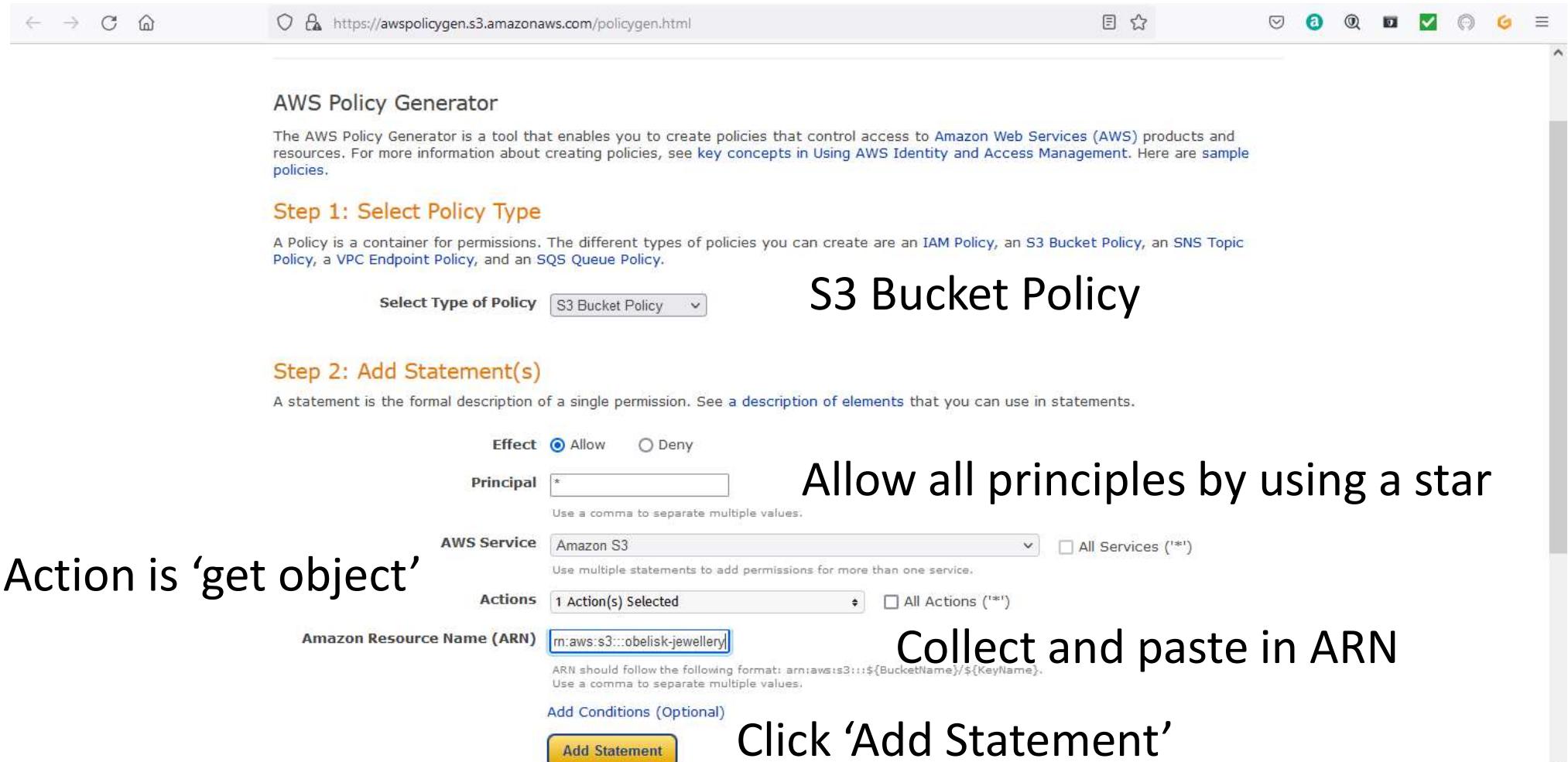
Amazon Resource Name (ARN)  ARN should follow the following format: arn:aws:s3:::\${BucketName}/\${KeyName}.  
Use a comma to separate multiple values.

Add Conditions (Optional)

Allow all principles by using a star

Collect and paste in ARN

Click 'Add Statement'



# Click 'Generate policy'

The screenshot shows the AWS Policy Generator interface at <https://awspolicygen.s3.amazonaws.com/policygen.html>. The interface has fields for Principal, AWS Service (set to Amazon S3), Actions (dropdown menu), and Amazon Resource Name (ARN). Below these are sections for Add Conditions (Optional) and a yellow 'Add Statement' button. A message indicates a statement was added, followed by a table of the statement details. At the bottom is a 'Step 3: Generate Policy' section with 'Generate Policy' and 'Start Over' buttons.

You added the following statements. Click the button below to Generate a policy.

Principal(s)	Effect	Action	Resource	Conditions
* • *	Allow	• s3:GetObject	arn:aws:s3:::obelisk-jewellery	None

**Step 3: Generate Policy**

A *policy* is a document (written in the Access Policy Language) that acts as a container for one or more statements.

**Generate Policy**   **Start Over**

This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring that your

https://awspolicygen.s3.amazonaws.com/policygen.html

Principal

Use a comma to separate multiple values.

AWS Service

Policy JSON Document

Click below to edit. To save the policy, copy the text below to a text editor.  
Changes made below will not be reflected in the policy generator tool.

```
{  
  "Id": "Policy1634485816522",  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "Stmt1634485721629",  
      "Action": [  
        "s3:GetObject"  
      ],  
      "Effect": "Allow",  
      "Resource": "arn:aws:s3:::obelisk-jewellery",  
      "Principal": "*"  
    }  
  ]  
}
```

You added

Principal

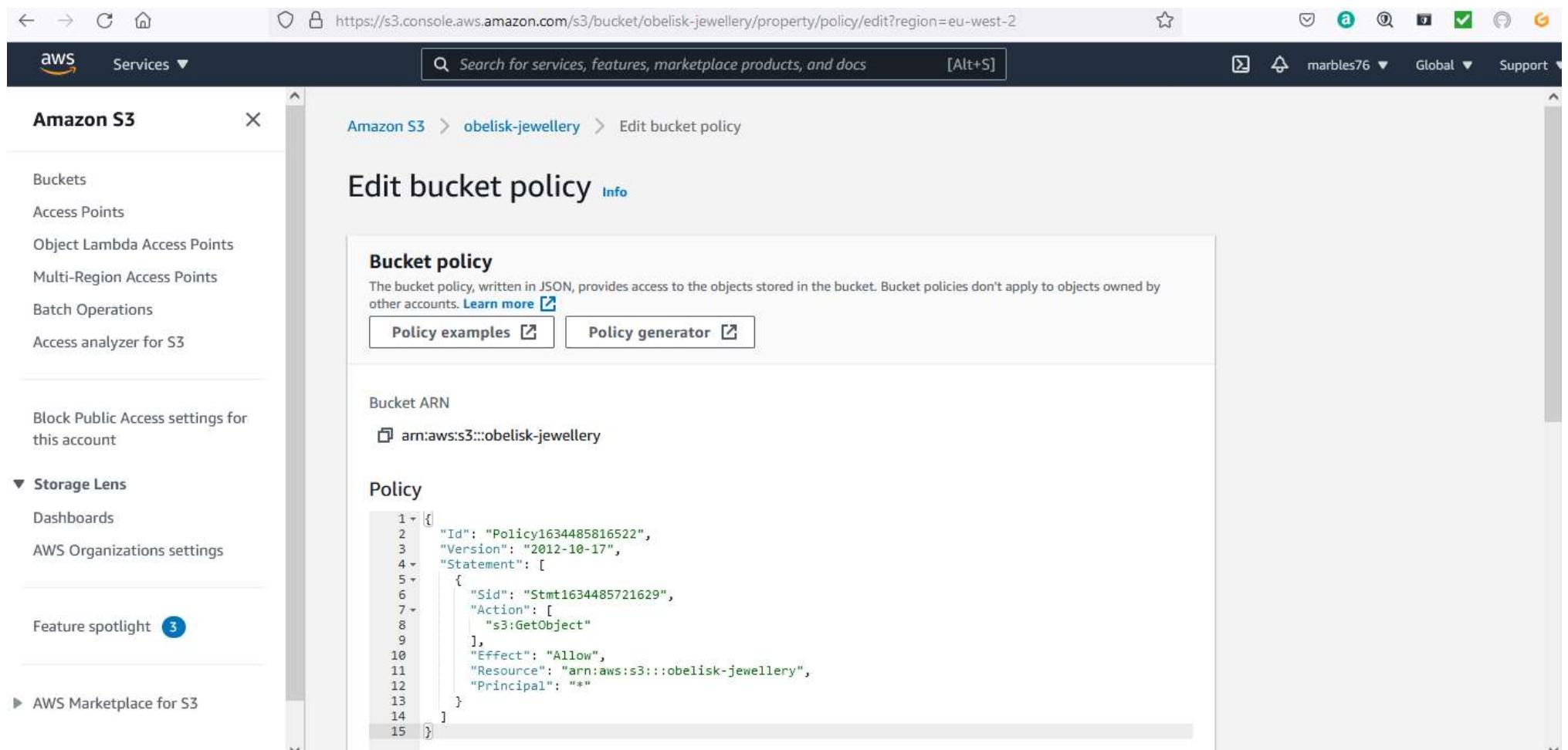
Step 3:

Copy this policy into the bucket policy

A policy is a JSON document that defines the permissions for a resource. It consists of one or more statements, each defining a principal, an action, and a resource. The principal can be a user, a group, or a role. The action can be a specific AWS service operation or a wildcard action. The resource can be a specific AWS resource or a wildcard resource. The effect can be Allow or Deny. The version of the policy is set to 2012-10-17. The policy ID is Policy1634485816522.

This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring that your

# Edit the bucket policy



The screenshot shows the AWS S3 console interface for editing a bucket policy. The URL in the browser is [https://s3.console.aws.amazon.com/s3/bucket/obelisk-jewellery\[property/policy/edit?region=eu-west-2](https://s3.console.aws.amazon.com/s3/bucket/obelisk-jewellery	property/policy/edit?region=eu-west-2). The left sidebar shows navigation options like Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and Access analyzer for S3. Below these are sections for Block Public Access settings and Storage Lens. A Feature spotlight section is also present. The main content area is titled "Edit bucket policy" and contains a "Bucket policy" section with a JSON editor. The JSON code is as follows:

```
1 {  
2   "Id": "Policy1634485816522",  
3   "Version": "2012-10-17",  
4   "Statement": [  
5     {  
6       "Sid": "Stmt1634485721629",  
7       "Action": [  
8         "s3:GetObject"  
9       ],  
10      "Effect": "Allow",  
11      "Resource": "arn:aws:s3:::obelisk-jewellery",  
12      "Principal": "*"  
13    }  
14  ]  
15 }
```

https://s3.console.aws.amazon.com/s3/buckets/obelisk-jewellery?region=eu-west-2&tab=permissions

aws Services ▾ Search for services, features, marketplace products, and docs [Alt+S] marbles76 ▾ Global ▾ Support ▾

**Amazon S3**

Buckets

- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- Access analyzer for S3

Block Public Access settings for this account

▶ Storage Lens

Feature spotlight 3

▶ AWS Marketplace for S3

### Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts.

[Learn more](#)

[Edit](#) [Delete](#)

[Copy](#)

```
{  
    "Version": "2012-10-17",  
    "Id": "Policy1634485816522",  
    "Statement": [  
        {  
            "Sid": "Stmt1634485721629",  
            "Effect": "Allow",  
            "Principal": "*",  
            "Action": "s3:GetObject",  
            "Resource": "arn:aws:s3:::obelisk-jewellery/*"  
        }  
    ]  
}
```

Add /\* onto end of resource to allow access to all resources in this bucket

Click 'Save'

The screenshot shows the AWS S3 'Edit bucket policy' page. The left sidebar is titled 'Amazon S3' and includes links for Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and Access analyzer for S3. Below these are sections for Block Public Access settings for this account, Storage Lens (Dashboards, AWS Organizations settings), Feature spotlight (with 3 notifications), and AWS Marketplace for S3. The main content area shows the 'Bucket policy' section with a JSON policy document. The policy document grants 'Allow' access to all objects in the 'obelisk-jewellery' bucket for all principals. The policy is as follows:

```
1 {  
2   "Id": "Policy1634485816522",  
3   "Version": "2012-10-17",  
4   "Statement": [  
5     {  
6       "Sid": "Stmt1634485721629",  
7       "Action": [  
8         "s3:GetObject"  
9       ],  
10      "Effect": "Allow",  
11      "Resource": "arn:aws:s3:::obelisk-jewellery/*",  
12      "Principal": "*"  
13    }  
14  ]  
15 }
```

# Go to 'Access control list'

The screenshot shows the AWS S3 console with the URL <https://s3.console.aws.amazon.com/s3/buckets/obelisk-jewellery?region=eu-west-2&tab=permissions>. A green success message box is displayed: "Successfully edited bucket policy." Below it, the "Access control list (ACL)" section is shown. A note says: "The console displays combined access grants for duplicate grantees. To see the full list of ACLs, use the Amazon S3 REST API, AWS CLI, or AWS SDKs." The table lists grants:

Grantee	Objects	Bucket ACL
Bucket owner (your AWS account) Canonical ID: 4a5ae8877248792df993f98189dcdb2ad7cad0a98e3193a9dcc3127c6f51d948	List, Write	Read, Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	-	-
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	-	-
S3 log delivery group Group: http://acs.amazonaws.com/groups/s3/LogDelivery	-	-

At the bottom, there is a "Cross-origin resource sharing (CORS)" section.

# Everyone needs access to objects list, click box

The screenshot shows the AWS S3 Access Control List (ACL) configuration page. The left sidebar lists various AWS services like Buckets, Access Points, and Storage Lens. The main content area is titled "Access control list (ACL)" and instructs users to "Grant basic read/write permissions to other AWS accounts". It shows two entries:

Grantee	Objects	Bucket ACL
Bucket owner (your AWS account)	<input checked="" type="checkbox"/> List <input checked="" type="checkbox"/> Write	<input checked="" type="checkbox"/> Read <input checked="" type="checkbox"/> Write
Everyone (public access)	<input checked="" type="checkbox"/> List <input type="checkbox"/> Write	<input type="checkbox"/> Read <input type="checkbox"/> Write
Authenticated users group (anyone with an AWS account)	<input type="checkbox"/> List <input type="checkbox"/> Write	<input type="checkbox"/> Read <input type="checkbox"/> Write

A red oval highlights the "List" checkbox for the "Everyone" entry. The bottom status bar shows the Windows taskbar with icons for File Explorer, Task View, Mail, and others, along with weather information (16°C Partly sunny).

# Tick box and save changes

The screenshot shows the AWS S3 Bucket Properties page for a bucket named "obelisk-jewellery". The left sidebar lists various S3-related options like Buckets, Access Points, and Storage Lens. The main content area displays two groups: "Authenticated users group" and "S3 log delivery group", each with "List" and "Read/Write" checkboxes. A warning message states: "When you grant access to the Everyone or Authenticated users group grantees, anyone in the world can access the objects in this bucket." Below this is a checkbox labeled "I understand the effects of these changes on my objects and buckets.", which is checked and highlighted with a red oval. At the bottom right are "Cancel" and "Save changes" buttons, with "Save changes" also highlighted with a red oval.

Authenticated users group  
(anyone with an AWS account)

Group:  http://acs.amazonaws.com/groups/global/AuthenticatedUsers

S3 log delivery group

Group:  http://acs.amazonaws.com/groups/s3/LogDelivery

**⚠️ When you grant access to the Everyone or Authenticated users group grantees, anyone in the world can access the objects in this bucket.**

[Learn more](#)

I understand the effects of these changes on my objects and buckets.

**Access for other AWS accounts**

No other AWS accounts associated with the resource.

Add grantee

Cancel **Save changes**

← → ⌂ ⌂ https://s3.console.aws.amazon.com/s3/buckets/obelisk-jewellery?region=eu-west-2&tab=permissions

Services ▾ Search for services, features, marketplace products, and docs [Alt+S]

marbles76 ▾ Global ▾ Support ▾

**Amazon S3** X Successfully edited access control list.

Buckets

Access Points

Object Lambda Access Points

Multi-Region Access Points

Batch Operations

Access analyzer for S3

Block Public Access settings for this account

**Storage Lens**

Dashboards

AWS Organizations settings

Feature spotlight 3

AWS Marketplace for S3

Amazon S3 > obelisk-jewellery

## obelisk-jewellery Info

**Publicly accessible**

Objects Properties **Permissions** Metrics Management Access Points

### Permissions overview

Access ⚠️ Public

### Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Edit

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

## Part 4

### Identity and access management – Click on ‘Services’ and then ‘IAM’

The screenshot shows the AWS Management Console interface. The top navigation bar has 'Services' highlighted with a red circle. Below it, the main content area is titled 'All services'. A section titled 'Security, Identity, & Compliance' contains a link 'IAM' which is also highlighted with a red circle. Other services listed in this category include Resource Access Manager, Cognito, Secrets Manager, GuardDuty, Inspector, Amazon Macie, AWS Single Sign-On, Certificate Manager, Key Management Service, CloudHSM, Directory Service, WAF & Shield, and AWS Firewall Manager. To the left, there's a sidebar with 'Favorites' and 'Recently visited' sections, and a search bar at the top.

Services

Search for services, features, marketplace products, and docs [Alt+S]

marbles76 Global Support

**Favorites**  
Add favorites by clicking on the star next to the service name.

**Recently visited**  
S3  
Console Home

**All services**

- Application Discovery Service
- Control Tower
- MSK
- AWS Glue DataBrew
- Amazon FinSpace
- Internet of Things
- IoT Core
- FreeRTOS
- IoT 1-Click
- IoT Analytics
- IoT Device Defender
- IoT Device Management
- IoT Events
- IoT Greengrass
- IoT SiteWise
- IoT Things Graph
- Security, Identity, & Compliance
- IAM
- Resource Access Manager
- Cognito
- Secrets Manager
- GuardDuty
- Inspector
- Amazon Macie
- AWS Single Sign-On
- Certificate Manager
- Key Management Service
- CloudHSM
- Directory Service
- WAF & Shield
- AWS Firewall Manager

**Networking & Content Delivery**

- VPC
- CloudFront
- Route 53
- API Gateway
- Direct Connect
- AWS App Mesh
- AWS Cloud Map
- Global Accelerator

**Media Services**

- Kinesis Video Streams
- MediaConnect
- MediaConvert

Feedback English (US) © 2009–2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Click on 'User groups'

The screenshot shows the AWS Identity and Access Management (IAM) console. The left sidebar is titled "Identity and Access Management (IAM)" and contains the following navigation options:

- Dashboard
- User groups** (highlighted with an orange oval)
- Users
- Roles
- Policies
- Identity providers
- Account settings

The main content area is titled "User groups (0)" and includes the following details:

- A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.
- Buttons: Refresh, Delete, Create group.
- Search bar: Filter User groups by property or group name and press enter.
- Table headers: Group name, Users, Permissions, Creation time.
- Message: No resources to display.

# Click on ‘Create user group’

The screenshot shows the AWS IAM 'Create user group' interface. On the left, a sidebar menu is open under 'Identity and Access Management (IAM)'. The 'User groups' option is selected and highlighted in orange. The main content area is titled 'Create user group' and contains a 'Name the group' section. In this section, the 'User group name' field is filled with 'manage-obelisk-jewellery'. Below the field, a note states: 'Maximum 128 characters. Use alphanumeric and '+,-,.,@,\_' characters.' Further down, there is an 'Add users to the group - Optional (0)' section with a search bar and a table header: 'User name', 'Groups', 'Last activity', and 'Creation time'. A message at the bottom of this section says 'No resources to display'. The top of the page includes the AWS logo, a search bar, and the URL https://console.aws.amazon.com/iamv2/home?#/groups/create.

Name the group, click through to ‘Create group’

https://console.aws.amazon.com/iamv2/home#/groups/details/manage-obelisk-jewellery

Search for services, features, marketplace products, and docs [Alt+S]

marbles76 ▾ Global ▾ Support ▾

## Identity and Access Management (IAM)

Dashboard

**User groups**

- Users
- Roles
- Policies
- Identity providers
- Account settings

**Access reports**

- Access analyzer
- Archive rules
- Analyzers
- Settings
- Credential report
- Organization activity
- Service control policies (SCPs)

IAM > User groups > manage-obelisk-Jewellery

# manage-obelisk-jewellery

**Delete** **Edit**

### Summary

User group name	Creation time	ARN
manage-obelisk-jewellery	October 17, 2021, 21:55 (UTC+01:00)	arn:aws:iam::230565916517:group/manage-obelisk-jewellery

**Users** **Permissions** **Access advisor**

#### Users in this group (0) Info

An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

**Search** **Remove users** **Add users**

User name	Groups	Last activity	Creation time
No resources to display			

# Click on ‘Policies’ and ‘Create policy’

The screenshot shows the AWS Identity and Access Management (IAM) console. The left sidebar is collapsed, showing the following navigation structure:

- Dashboard
- Access management
  - User groups
  - Users
  - Roles
  - Policies (highlighted with a red circle)
  - Identity providers
  - Account settings
- Access reports
  - Access analyzer
  - Archive rules
  - Analyzers
  - Settings
  - Credential report
  - Organization activity
  - Service control policies (SCPs)

The main content area displays the 'Policies' list. A blue banner at the top says "Introducing the new Policies list experience". Below it, the heading "Policies (868)" is followed by a brief description: "A policy is an object in AWS that defines permissions." A search bar and a pagination control are visible. The table has columns: Policy Name, Type, Used as, and Description. The "Create Policy" button in the Actions dropdown is circled in red.

Policy Name	Type	Used as	Description
AWSDirectConnectReadOnlyAccess	AWS managed	None	Provides read only a
AmazonGlacierReadOnlyAccess	AWS managed	None	Provides read only a
AWSMarketplaceFullAccess	AWS managed	None	Provides the ability t
ClientVPNServiceRolePolicy	AWS managed	None	Policy to enable AWS
AWSSSODirectoryAdministrator	AWS managed	None	Administrator access
AWSIoT1ClickReadOnlyAccess	AWS managed	None	Provides read only a
AutoScalingConsoleReadOnlyAccess	AWS managed	None	Provides read-only a
AmazonDMSRedshiftS3Role	AWS managed	None	Provides access to n

At the bottom, there are links for Feedback, Language selection (English (US)), Copyright notice (© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.), and Legal links (Privacy Policy, Terms of Use, Cookie preferences).

Documentation

The screenshot shows the AWS IAM 'Create policy' page. At the top, there's a navigation bar with links for 'Services', a search bar, and user information ('marbles76'). Below the header, the title 'Create policy' is displayed, followed by a sub-header explaining what a policy is. There are two tabs: 'Visual editor' (which is selected) and 'JSON'. To the right of the tabs is a link to 'Import managed policy'. The main content area is titled 'Select a service' and contains four sections: 'Service' (with a 'Choose a service' link), 'Actions' (with a 'Choose a service before defining actions' link), 'Resources' (with a 'Choose actions before applying resources' link), and 'Request conditions' (with a 'Choose actions before specifying conditions' link). At the bottom right of this section is a blue button labeled '+ Add additional permissions'. The bottom of the page includes standard footer links for 'Feedback', 'English (US)', 'Privacy Policy', 'Terms of Use', and 'Cookie preferences'.

# Go to 'JSON' tab and click on 'Import managed policy'

The screenshot shows the AWS IAM 'Create policy' interface. At the top, there's a navigation bar with links for Services, a search bar, and user information. Below the navigation is a title 'Create policy'. A horizontal toolbar has three tabs: 'Visual editor' (disabled), 'JSON' (selected and highlighted with a red oval), and 'Import managed policy' (disabled and highlighted with a red oval). The main area contains a JSON code editor with the following content:

```
1 {  
2     "Version": "2012-10-17",  
3     "Statement": []  
4 }
```

Below the editor, status indicators show Security: 0, Errors: 0, Warnings: 0, and Suggestions: 0. At the bottom of the page, there are links for Feedback, English (US), Copyright notice (© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.), Privacy Policy, Terms of Use, and Cookie preferences.

https://console.aws.amazon.com/iam/home#/policies\$new?step=edit

Services ▾

Search for services, features, marketplace products, and docs [Alt+S]

marbles76 ▾ Global ▾ Support ▾

## Create policy

A policy defines the permissions that are granted to users or groups.

Visual editor

Import managed policies

Choose policies to import

The permissions for the chosen policies will be added to your policy. You can review your policy's final permissions before you save it.

Filter policies ▾ Search Showing 868 results

	Policy name	Used as	Description
<input type="radio"/>	AccessAnalyzerServiceRolePolicy	None	Allow Access Analyzer to analyze reso...
<input type="radio"/>	AdministratorAccess	None	Provides full access to AWS services ...
<input type="radio"/>	AdministratorAccess-Amplify	None	Grants account administrative permis...
<input type="radio"/>	AdministratorAccess-AWSElasticBeanstalk	None	Grants account administrative permis...
<input type="radio"/>	AlexaForBusinessDeviceSetup	None	Provide device setup access to Alexa...
<input type="radio"/>	AlexaForBusinessFullAccess	None	Grants full access to AlexaForBusines...

Cancel Import

Feedback English (US) ▾

© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Privacy Policy Terms of Use Cookie preferences

Type 'S3' into search bar and then click 'AmazonS3FullAccess' and 'Import'

The screenshot shows the AWS IAM 'Create policy' interface. A modal window titled 'Import managed policies' is open. In the search bar at the top of the modal, the text 's3' is typed. Below the search bar, a heading says 'Choose policies to import' with a sub-instruction: 'The permissions for the chosen policies will be added to your policy. You can review your policy's final permissions before you save it.' A table lists nine results, with the second row, 'AmazonS3FullAccess', being the selected one. This row has a blue circular selection indicator next to the radio button. The 'Import' button at the bottom right of the modal is also circled in red. The background of the main IAM page shows a partially visible policy editor and some user profile icons.

Policy name	Used as	Description
AmazonDMSRedshiftS3Role	None	Provides access to manage S3 setting...
AmazonS3FullAccess	None	Provides full access to all buckets via ...
AmazonS3ObjectLambdaExecutionRolePolicy	None	Provides AWS Lambda functions perm...
AmazonS3OutpostsFullAccess	None	Provides full access to Amazon S3 on ...
AmazonS3OutpostsReadOnlyAccess	None	Provides read only access to Amazon ...
AmazonS3ReadOnlyAccess	None	Provides read only access to all buck...

# Before clicking on 'Review policy', get ARN from the bucket policy section

The screenshot shows the AWS IAM 'Create policy' page. At the top, there's a navigation bar with links for Services, Search, and User (marbles76). Below the navigation is a breadcrumb trail: 'Create policy' > '1'. The main content area is titled 'Create policy' and contains a JSON editor. The JSON code is as follows:

```
1 {  
2     "Version": "2012-10-17",  
3     "Statement": [  
4         {  
5             "Effect": "Allow",  
6             "Action": [  
7                 "s3:*",  
8                 "s3-object-lambda:/*"  
9             ],  
10            "Resource": "*"  
11        }  
12    ]  
13 }
```

Below the JSON editor, there are status indicators: Security: 0, Errors: 0, Warnings: 0, and Suggestions: 0. At the bottom of the page, there are links for Feedback, English (US), Copyright notice (© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.), Privacy Policy, Terms of Use, and Cookie preferences.

# Go to bucket policy

The screenshot shows the AWS S3 console interface. The left sidebar is collapsed, showing options like 'Buckets', 'Access Points', and 'Bucket policy'. The main content area displays the 'Bucket policy' for the 'obelisk-jewellery' bucket. The policy JSON is shown:

```
{
  "Version": "2012-10-17",
  "Id": "Policy1634485816522",
  "Statement": [
    {
      "Sid": "Stmt1634485721629",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::obelisk-jewellery/*"
    }
  ]
}
```

A large red oval highlights the 'Resource' field in the JSON, which contains the ARN 'arn:aws:s3:::obelisk-jewellery/\*'. To the right of the JSON, the text 'ARN here, copy it' is overlaid. A 'Copy' button is also visible next to the JSON.

Block public access to buckets and objects granted through **new** public bucket or access point policies  
⚠ Off

Block public and cross-account access to buckets and objects through **any** public bucket or access point policies  
⚠ Off

**Bucket policy**  
The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. [Learn more](#)

[Edit](#) [Delete](#)

[Copy](#)

ARN here, copy it

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

[Visual editor](#) [JSON](#) [Import managed policy](#)

```
1 {  
2     "Version": "2012-10-17",  
3     "Statement": [  
4         {  
5             "Effect": "Allow",  
6             "Action": [  
7                 "s3:*",  
8                 "s3-object-lambda:*"  
9             ],  
10            "Resource": [  
11                "arn:aws:s3:::obelisk-jewellery",  
12                "arn:aws:s3:::obelisk-jewellery/*"  
13            ]  
14        }  
15    ]  
16}
```

Type this extra code  
(use ARN) to only allow  
full access to the new  
bucket and everything  
within it.

Click on ‘Review policy’

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Create policy

**Add tags (Optional)**

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

**Add tag**

You can add up to 50 more tags

Cancel Previous **Next: Review**

# Click on 'Next Review'

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Type in name and description

Screenshot of the AWS IAM 'Review policy' page.

The 'Name' field contains 'obelisk-jewellery-policy'. The 'Description' field contains 'Access to S3 bucket for Obelisk Jewellery static files'.

**Review policy**

**Name:** obelisk-jewellery-policy  
Use alphanumeric and '+=\_@-' characters. Maximum 128 characters.

**Description:** Access to S3 bucket for Obelisk Jewellery static files  
Maximum 1000 characters. Use alphanumeric and '+=\_@-' characters.

**Summary**

This policy defines some actions, resources, or conditions that do not provide permissions. To grant access, policies must have an action that has an applicable resource or condition. For details, choose **Show remaining**. [Learn more](#)

Service	Access level	Resource	Request condition
Allow (1 of 297 services) <a href="#">Show remaining 296</a>	<b>Limited:</b> List, Read, Write, Permissions management, Tagging	Multiple	None

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Click 'Create policy'

This screenshot shows the 'Review policy' step in the AWS IAM 'Create policy' wizard. The policy defines an action for the S3 service.

**Summary**

This policy defines some actions, resources, or conditions that do not provide permissions. To grant access, policies must have an action that has an applicable resource or condition. For details, choose [Show remaining](#). [Learn more](#)

**Filter**

Service	Access level	Resource	Request condition
Allow (1 of 297 services) <a href="#">Show remaining 296</a>			
S3	Limited: List, Read, Write, Permissions management, Tagging	Multiple	None

**Tags**

Key	Value
No tags associated with the resource.	

\* Required

Cancel Previous **Create policy**

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Policy created successfully

The screenshot shows the AWS Identity and Access Management (IAM) Policies list page. A red circle highlights a green success message at the top: "The policy obelisk-jewellery-policy has been created." The left sidebar shows the navigation menu for IAM, with "Policies" selected. The main table lists various policies, including the newly created one.

Policy Name	Type	Used as	Description
obelisk-jewellery-policy	Customer managed	None	Access to S3 bucket
AWSDirectConnectReadOnlyAccess	AWS managed	None	Provides read only a
AmazonGlacierReadOnlyAccess	AWS managed	None	Provides read only a
AWSMarketplaceFullAccess	AWS managed	None	Provides the ability t
ClientVPNServiceRolePolicy	AWS managed	None	Policy to enable AWS
AWSSSODirectoryAdministrator	AWS managed	None	Administrator access
AWSIoT1ClickReadOnlyAccess	AWS managed	None	Provides read only a

Click on ‘User groups’ and on the group name (‘Manage-obelisk-jewellery’)

The screenshot shows the AWS Identity and Access Management (IAM) console. The left sidebar has a tree view with 'Access management' expanded, showing 'User groups' which is circled in red. Other options like 'Users', 'Roles', 'Policies', 'Identity providers', and 'Account settings' are also listed under 'Access management'. Below that is another section titled 'Access reports' with options like 'Access analyzer', 'Analyzers', 'Settings', 'Credential report', 'Organization activity', and 'Service control policies (SCPs)'. The main content area is titled 'User groups (1) Info' and contains a table with one row. The table columns are 'Group name', 'Users', 'Permissions', and 'Creation time'. The single row shows a group named 'manage-obelisk-jewellery'. This row also has a red circle around it. The 'Permissions' column for this group shows a warning icon and 'Not defined'. The 'Creation time' column shows '27 minutes ago'. At the top of the main content area, there are buttons for 'Delete' and 'Create group'.

Group name	Users	Permissions	Creation time
manage-obelisk-jewellery		⚠ 0 ⚠ Not defined	27 minutes ago

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Click on 'Attach Policies'

The screenshot shows the AWS Identity and Access Management (IAM) console. On the left, the navigation menu is expanded to show 'Access management' under 'Identity and Access Management (IAM)'. The 'User groups' option is selected. In the main content area, the 'manage-obelisk-jewellery' user group is displayed. The 'Permissions' tab is active. At the top right of this section, there is a 'Add permissions' button with a dropdown menu. The 'Attach Policies' option in this menu is highlighted with a red oval. Other options in the dropdown include 'Simulate', 'Remove', and 'Create Inline Policy'. Below this, a table header for 'Permissions policies (0)' is visible, along with a search bar and filter options for 'Policy Name', 'Type', and 'Description'. A message at the bottom of the table area says 'No resources to display'. The browser's address bar shows the URL: <https://console.aws.amazon.com/iamv2/home#/groups/details/manage-obelisk-jewellery?section=permissions>.

Identity and Access Management (IAM) X

IAM > User groups > manage-obelisk-jewellery > Add permissions

### Attach permission policies to manage-obelisk-jewellery

▶ Current permissions policies (0)

**Other permission policies (693) Info**

You can attach up to 10 managed policies to this user group. All of the users in this group inherit the attached permissions.

Filter policies by property or policy name and press enter

<input type="checkbox"/>	Policy Name <small>↗</small>	Type	Description
<input type="checkbox"/>	obelisk-jewellery-policy	Customer managed	Access to S3 bucket for
<input type="checkbox"/>	AWSDirectConnectReadOnlyAccess	AWS managed	Provides read only acce
<input type="checkbox"/>	AmazonGlacierReadOnlyAccess	AWS managed	Provides read only acce
<input type="checkbox"/>	AWSMarketplaceFullAccess	AWS managed	Provides the ability to su
<input type="checkbox"/>	AWSSSOAdministrator	AWS managed	Administrator access for
<input type="checkbox"/>	AWSIoT1ClickReadOnlyAccess	AWS managed	Provides read only acce

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Search for policy just created

<input type="checkbox"/>	Policy Name <small>↗</small>	Type	Description
<input type="checkbox"/>	obelisk-jewellery-policy	Customer managed	Access to S3 bucket for
<input type="checkbox"/>	AWSDirectConnectReadOnlyAccess	AWS managed	Provides read only acce
<input type="checkbox"/>	AmazonGlacierReadOnlyAccess	AWS managed	Provides read only acce
<input type="checkbox"/>	AWSMarketplaceFullAccess	AWS managed	Provides the ability to su
<input type="checkbox"/>	AWSSSOAdministrator	AWS managed	Administrator access for
<input type="checkbox"/>	AWSIoT1ClickReadOnlyAccess	AWS managed	Provides read only acce

Click on policy and click 'Attach Policy'

← → C ⌂ https://console.aws.amazon.com/iamv2/home#/groups/details/manage-obelisk-jewellery?section=permissions ⌂ Search for services, features, marketplace products, and docs [Alt+S] marbles76 Global Support

aws Services ▾

Identity and Access Management (IAM) Policies attached to this user group

IAM > User groups > manage-obelisk-jewellery

## manage-obelisk-jewellery

Delete Edit

Summary

User group name manage-obelisk-jewellery	Creation time October 17, 2021, 21:55 (UTC+01:00)	ARN arn:aws:iam::230565916517:group/manage-obelisk-jewellery
---	--	---

Users Permissions Access advisor

Permissions policies (1) Info You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter

Policy Name Type Description

obelisk-jewellery-policy Customer managed Access to S3 bucket for Obelisk Jewell..

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Click on 'Users' and then 'Add User'

The screenshot shows the AWS Identity and Access Management (IAM) service in the AWS Management Console. The URL in the browser is <https://console.aws.amazon.com/iamv2/home#/users>. The left sidebar has a tree view with 'Identity and Access Management (IAM)' selected. Under 'Access management', 'User groups' is collapsed, and 'Users' is selected and highlighted with a red oval. Other options like 'Roles', 'Policies', 'Identity providers', and 'Account settings' are listed. Under 'Access reports', 'Access analyzer', 'Archive rules', 'Analyzers', 'Settings', 'Credential report', 'Organization activity', and 'Service control policies (SCPs)' are listed. The main content area shows a blue banner at the top stating 'Introducing the new Users list experience' with a message about redesigning the experience. Below the banner, the path 'IAM > Users' is shown. A table header for 'Users (0)' includes columns for 'User name', 'Groups', 'Last activity', 'MFA', 'Password age', and 'Active key age'. A search bar says 'Find users by username or access key'. At the top right of the table are 'Delete' and 'Add users' buttons, with 'Add users' being highlighted with a red oval. Below the table, a message says 'No resources to display'. The bottom navigation bar includes links for 'Feedback', language 'English (US)', copyright notice '© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.', and 'Privacy Policy', 'Terms of Use', and 'Cookie preferences'.

# Type in user name

The screenshot shows the AWS IAM 'Add user' wizard, Step 1: Set user details. The URL is https://console.aws.amazon.com/iam/home#/users/new?step=details. The page title is 'Add user'. A navigation bar at the top includes the AWS logo, 'Services ▾', a search bar ('Search for services, features, marketplace products, and docs [Alt+S]'), and account information ('marbles76 ▾', 'Global ▾', 'Support ▾'). Below the title, a step indicator shows '1' (highlighted in blue) and '2' through '5'. The main section is titled 'Set user details' with the sub-instruction 'You can add multiple users at once with the same access type and permissions. [Learn more](#)'. It features a 'User name\*' input field and a '+ Add another user' button. The next section, 'Select AWS access type', includes instructions: 'Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)'. It lists two options: 'Access key - Programmatic access' (selected with a checked checkbox) and 'Password - AWS Management Console access' (unchecked). At the bottom, there are links for '\* Required', 'Cancel', and 'Next: Permissions'.

Add user

1 2 3 4 5

Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name\*

+ Add another user

Select AWS access type

Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Select AWS credential type\*  **Access key - Programmatic access**  
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

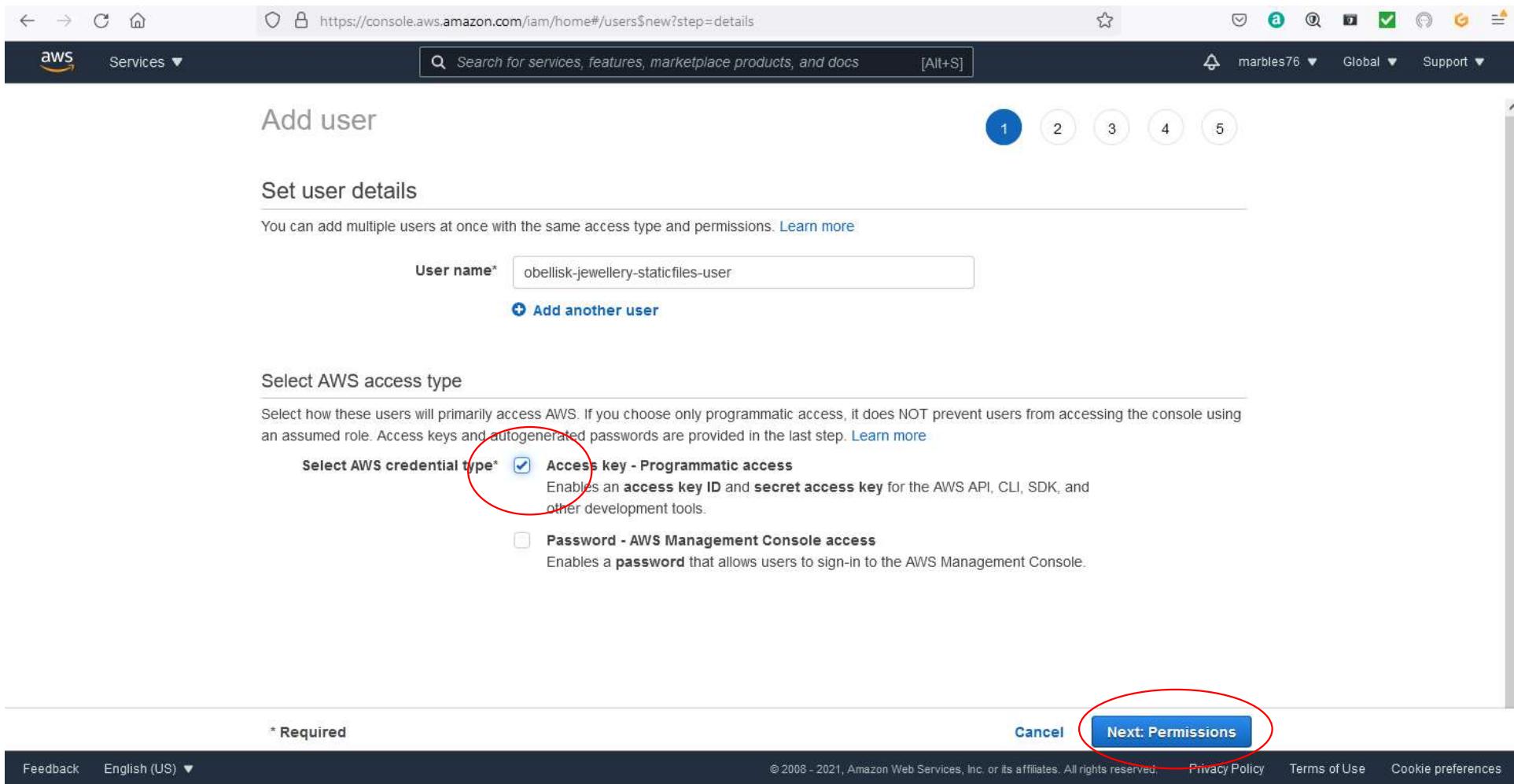
**Password - AWS Management Console access**  
Enables a **password** that allows users to sign-in to the AWS Management Console.

\* Required

Cancel Next: Permissions

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Add programmatic access



The screenshot shows the AWS IAM 'Add user' wizard at step 1: Set user details. The user name is set to 'obelisk-jewellery-staticfiles-user'. Under 'Select AWS access type', the 'Access key - Programmatic access' checkbox is selected and highlighted with a red circle. The 'Password - AWS Management Console access' checkbox is unselected. At the bottom, the 'Next: Permissions' button is also highlighted with a red circle.

Add user

Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name\* obelisk-jewellery-staticfiles-user

[+ Add another user](#)

Select AWS access type

Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Select AWS credential type\*  **Access key - Programmatic access**  
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

**Password - AWS Management Console access**  
Enables a **password** that allows users to sign-in to the AWS Management Console.

\* Required

Cancel **Next: Permissions**

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Then click on 'Next Permissions'

<https://console.aws.amazon.com/iam/home#/users/new?step=permissions&accessKey&userNames=obelisk-jewellery-sta>

Add user

1 2 3 4 5

Set permissions

Add user to group Copy permissions from existing user Attach existing policies directly

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Add user to group

Create group Refresh

Search Showing 1 result

Group	Attached policies
<input type="checkbox"/> manage-obelisk-jewellery	obelisk-jewellery-policy

Tick box to add user to group

Set permissions boundary

Cancel Previous Next: Tags

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Click 'Next Tags' and through to the end

https://console.aws.amazon.com/iam/home#/users\$new?step=tags&accessKey&userNames=obellisk-jewellery-staticfiles-1

aws Services ▾ Search for services, features, marketplace products, and docs [Alt+S]

marbles76 ▾ Global ▾ Support ▾

## Add user

1 2 3 4 5

### Add tags (optional)

IAM tags are key-value pairs you can add to your user. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this user. [Learn more](#)

Key	Value (optional)	Remove
<a href="#">Add new key</a>		

You can add 50 more tags.

Cancel Previous Next: Review

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Click 'Create User'

The screenshot shows the AWS IAM 'Add user' interface at the 'Review' step. The top navigation bar includes the AWS logo, services dropdown, search bar, and user account information. Below the header, the page title is 'Add user' and the current step is 'Review' (indicated by a blue circular progress indicator). The main content area displays 'User details' with fields: User name (obellisk-jewellery-staticfiles-user), AWS access type (Programmatic access - with an access key), and Permissions boundary (Permissions boundary is not set). The 'Permissions summary' section shows the user will be added to the 'manage-obelisk-jewellery' group. The 'Tags' section indicates no tags were added. At the bottom, there are 'Cancel', 'Previous', and 'Create user' buttons, with 'Create user' being the target of a red oval highlight.

Add user

Review

User details

User name	obellisk-jewellery-staticfiles-user
AWS access type	Programmatic access - with an access key
Permissions boundary	Permissions boundary is not set

Permissions summary

The user shown above will be added to the following groups.

Type	Name
Group	manage-obelisk-jewellery

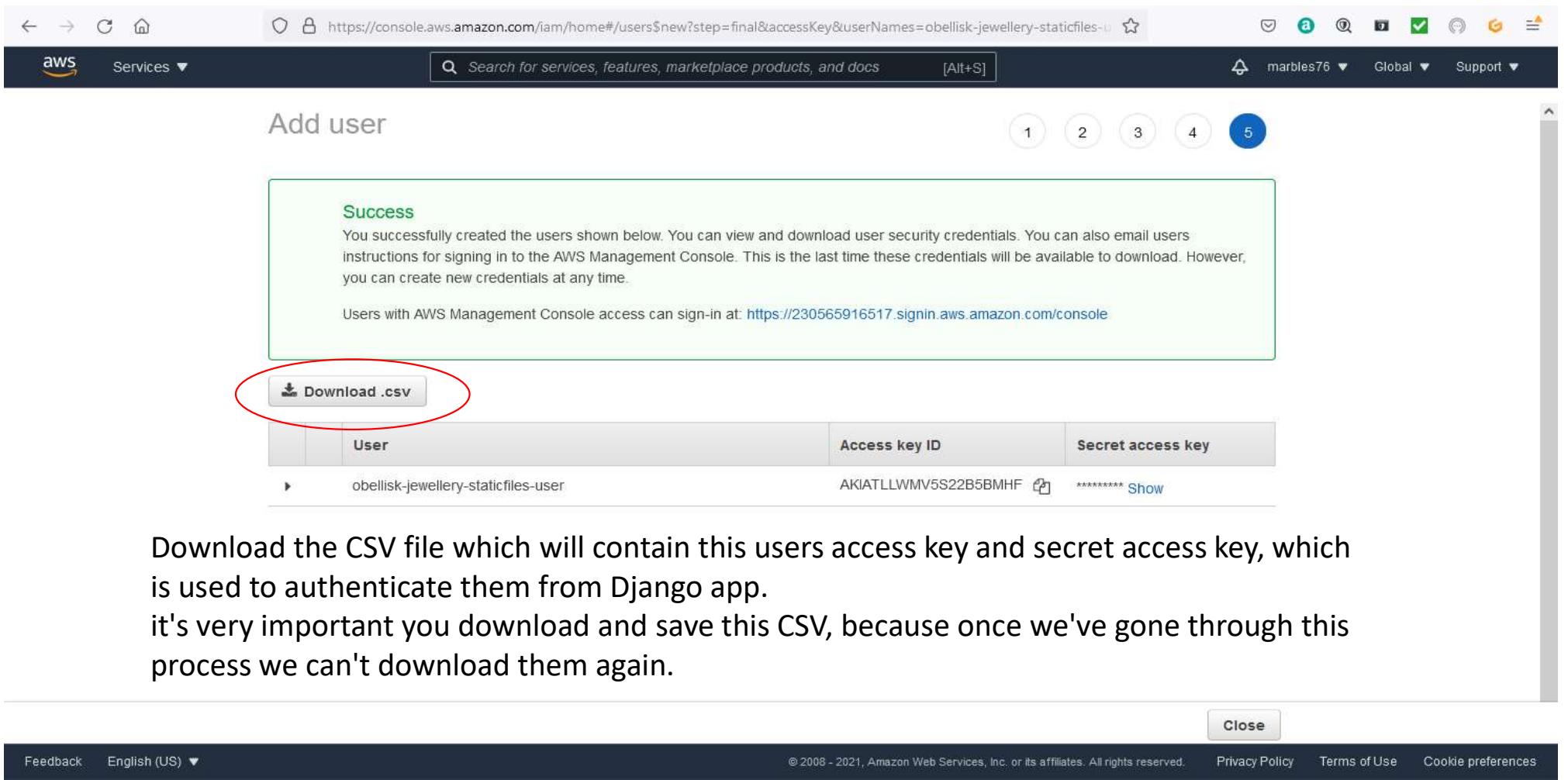
Tags

No tags were added.

Cancel Previous Create user

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Click on 'Download .csv' and store the information for future reference



The screenshot shows the AWS IAM 'Add user' success page. A red circle highlights the 'Download .csv' button. The page includes a success message, a user table, and navigation links.

**Success**  
You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://230565916517.signin.aws.amazon.com/console>

User	Access key ID	Secret access key
obellisk-jewellery-staticfiles-user	AKIATLLWMV5S22B5BMHF	***** Show

**Download .csv**

**Close**

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

https://console.aws.amazon.com/iam/home#/users\$new?step=final&accessKey&userNames=obellisk-jewellery-staticfiles-u

aws Services ▾ Search for services, features, marketplace products, and docs [Alt+S] marbles76 ▾ Global ▾ Support ▾

## Add user

1 2 3 4 5

**Success**  
You successfully created the user obellisk-jewellery-staticfiles-user. You can now sign in to the AWS Management Console using these credentials.

**Users with AWS Management Console access**

**Download .csv**

User
obellisk-jewellery-staticfiles-user

Opening new\_user\_credentials.csv

You have chosen to open:  
**new\_user\_credentials.csv**  
which is: Microsoft Excel Comma Separated Values File (222 bytes)  
from: blob:

What should Firefox do with this file?

Open with Excel (default)

Save File

Do this automatically for files like this from now on.

OK Cancel

Secret access key

\*\*\*\*\* Show

Close

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

## Part 5

Now s3 bucket has been created and the user's groups and security policies for the bucket, it's time to connect Django.

To do this, install two new packages: -

Pip3 install boto3 and pip3 install django-storages

Freeze those into the requirements.txt file so they get installed on Heroku when deploying.

The screenshot shows a Gitpod workspace interface. On the left, the Explorer sidebar lists files like views.py, base.css, base.html, settings.py, .gitignore, and requirements.txt. The requirements.txt file is open in the code editor, showing a list of Python dependencies:

```
asgiref==3.4.1
boto3==1.18.63
botocore==1.21.63
dj-database-url==0.5.0
Django==3.2.7
django-allauth==0.41.0
django-countries==7.2.1
django-crispy-forms==1.12.0
django-social-share==2.2.1
django-storages==1.12.2
gunicon==20.1.0
jmespath==0.10.0
oauthlib==3.1.1
Pillow==8.3.2
psycopg2-binary==2.9.1
python3-openid==3.2.0
pytz==2021.1
requests-oauthlib==1.3.0
s3transfer==0.5.0
sqlparse==0.4.1
stripe==2.60.0
```

Below the code editor is a terminal window showing the output of the pip freeze command:

```
s) (3.4.1)
Requirement already satisfied: pytz in /workspace/.pip-modules/lib/python3.8/site-packages (from Django>=2.2->django-storages) (2021.1)
Installing collected packages: django-storages
Successfully installed django-storages-1.22.2
MANUAL: You are using version 21.3; however, version 21.2 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
gitpod /workspace/obelisk_jewellery $ pip3 freeze > requirements.txt
gitpod /workspace/obelisk_jewellery $
```

The status bar at the bottom indicates the workspace is running Python 3.8.11 64-bit (3.8.11:pyenv) and is Shared.

## Add ‘storages’ to installed apps in settings.py

disk\_jewellery/settings.py at main · sanson0/obelisk\_jewellery — Mozilla Firefox

Search github sign in, Norton Safe Se obelisk\_jewellery/settings.py at X +

→ ⌂ ⌄ ⌅ https://github.com/sanson0/obelisk\_jewellery/blob/main/obelisk\_jew...

```
35     'django.contrib.admin',
36     'django.contrib.auth',
37     'django.contrib.contenttypes',
38     'django.contrib.sessions',
39     'django.contrib.messages',
40     'django.contrib.staticfiles',
41     'django.contrib.sites',
42     'allauth',
43     'allauth.account',
44     'allauth.socialaccount',
45     'home',
46     'products',
47     'bag',
48     'checkout',
49     'django_social_share',
50     'profiles',
51
52     # Other
53     'crispy_forms',
54     'storages',
55 ]
56
57 MIDDLEWARE = [
```

To connect Django to s3 we need to add some settings in `settings.py` to tell it which bucket it should be communicating with but only on Heroku. So add an if statement to check if there's an environment variable called `USE_AWS` in the environment. If so define the `AWS_STORAGE_BUCKET_NAME` The `AWS_S3_REGION_NAME` And our access key, and secret access key, which we'll get from the environment.

## An example

```
174 MEDIA_URL = '/media/'  
175 MEDIA_ROOT = os.path.join(BASE_DIR, 'media')  
176  
177  
178 if 'USE_AWS' in os.environ:  
179     AWS_STORAGE_BUCKET_NAME = 'ckz8780-boutique-ado'  
180     AWS_S3_REGION_NAME = 'us-east-1'  
181     AWS_ACCESS_KEY_ID = os.environ.get('AWS_ACCESS_KEY_ID')  
182     AWS_SECRET_ACCESS_KEY = os.environ.get('AWS_SECRET_ACCESS_KEY')  
183  
184 # Stripe  
185 FREE_DELIVERY_THRESHOLD = 50  
186 STANDARD_DELIVERY_PERCENTAGE = 10  
187 STRIPE_CURRENCY = 'usd'  
188 STRIPE_PUBLIC_KEY = os.getenv('STRIPE_PUBLIC_KEY', '')  
189 STRIPE_SECRET_KEY = os.getenv('STRIPE_SECRET_KEY', '')  
190 STRIPE_WH_SECRET = os.getenv('STRIPE_WH_SECRET', '')  
191 DEFAULT_FROM_EMAIL = 'boutique@do@example.com'  
192
```

Problems 1...gitpod/workspace/boutique\_ado\_v1 ✘ (0 Open Parts)

(none - dirty, 0h)

# Next, go to Heroku

The screenshot shows the Heroku dashboard for the app "obelisk-jewellery".

**GitHub repo:** sanson0/obelisk\_jewellery

**Heroku git URL:** https://git.heroku.com/obelisk-jewellery.git

**Config Vars** (Left sidebar)

Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.

**Config Vars (Main Area)**

KEY	VALUE	Actions
DATABASE_URL	postgres://	Hidden
DISABLE_COLLECTSTATIC	1	
SECRET_KEY		Hidden
KEY	VALUE	Add

**Buildpacks** (Bottom sidebar)

Add buildpack

The screenshot shows the Heroku dashboard for the app "obelisk-jewellery". At the top, it displays the GitHub repo information: "GitHub repo" is set to [sanson0/obelisk\\_jewellery](#), and "Heroku git URL" is <https://git.heroku.com/obelisk-jewellery.git>. Below this, the "Config Vars" section is shown. It includes a note: "Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own." On the right, there is a "Hide Config Vars" button. The config vars listed are:

Config Var	Value	Action
DATABASE_URL	postgres://	Hidden
DISABLE_COLLECTSTATIC	1	
SECRET_KEY		Hidden
AWS_ACCESS_KEY_ID		Hidden
AWS_SECRET_ACCESS_KEY		Hidden
USE_AWS	True	
KV	Valid	Add

As well as adding that key called USE\_AWS which I'll set to true. So that our settings file knows to use the AWS configuration when we deploy to Heroku.

## Delete the DISABLE\_COLLECTSTATIC variable

The screenshot shows the Heroku dashboard for the app "obelisk-jewellery". A modal dialog box is centered, titled "Delete (DISABLE\_COLLECTSTATIC)". It contains the question "Are you sure you want to remove this config variable?". Below the question are two buttons: "Cancel" (white background) and "Delete Config Var" (red background). The background of the dashboard shows several configuration variables listed in a table:

Config Vars		
Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.		
DATABASE_URL	postgres://[REDACTED]	/x
DISABLE_COLLECTSTATIC	1	/x
SECRET_KEY	[REDACTED]	/x
AWS_ACCESS_KEY_ID	[REDACTED]	/x
AWS_SECRET_ACCESS_KEY	[REDACTED]	/x
USE_AWS	True	/x

When deploying to Heroku this time django will collectstatic files automatically and upload them to s3.

We need to tell django where our static files will be coming from in production. Which is going to be our bucket name.s3.amazonaws.com  
And notice I'm formatting this as an F string so my bucket name from above will be interpreted and added to generate the appropriate URL.

The screenshot shows a Gitpod interface with the following details:

- File Explorer:** Shows the project structure under "OBELISK\_JEWELLERY".
- Code Editor:** The "settings.py" file is open, showing the following code snippet:

```
    172
    173
    174: # Static files (CSS, JavaScript, Images)
    175: # https://docs.djangoproject.com/en/3.2/howto/static-files/
    176
    177: STATIC_URL = '/static/'
    178: STATICFILES_DIRS = (os.path.join(BASE_DIR, 'static'),)
    179
    180: MEDIA_URL = '/media/'
    181: MEDIA_ROOT = os.path.join(BASE_DIR, 'media')
    182
    183: if 'USE_AWS' in os.environ:
    184:     # Bucket Config
    185:     AWS_STORAGE_BUCKET_NAME = 'obelisk-jewellery'
    186:     AWS_S3_REGION_NAME = 'eu-west-2'
    187:     AWS_ACCESS_KEY_ID = os.environ.get('AWS_ACCESS_KEY_ID')
    188:     AWS_SECRET_ACCESS_KEY = os.environ.get('AWS_SECRET_ACCESS_KEY')
    189:     AWS_S3_CUSTOM_DOMAIN = f'{AWS_STORAGE_BUCKET_NAME}.s3.amazonaws.com'
    190
```

A red oval highlights the line `AWS\_S3\_CUSTOM\_DOMAIN = f'{AWS\_STORAGE\_BUCKET\_NAME}.s3.amazonaws.com'`.

- Terminal:** Shows the command `gitpod /workspace/obelisk\_jewellery \$ pip3 freeze > requirements.txt` being run.
- Status Bar:** Shows the Python version (Python 3.8.11 64-bit (3.8.11: pyenv)), line (Ln 189), and column (Col 72).

The next step is to tell django that, in production,  
It must use s3 to store static files whenever someone runs  
collectstatic.

And that any uploaded product images should go there also.

Create new file custom\_storages.py and add code below

10 lines (6 sloc) | 252 Bytes

```
1 from django.conf import settings
2 from storages.backends.s3boto3 import S3Boto3Storage
3
4
5 class StaticStorage(S3Boto3Storage):
6     location = settings.STATICFILES_LOCATION
7
8
9 class MediaStorage(S3Boto3Storage):
10    location = settings.MEDIAFILES_LOCATION
```

## Settings.py

```
177  
178 if 'USE_AWS' in os.environ:  
179     # Bucket Config  
180     AWS_STORAGE_BUCKET_NAME = '123456789-boutique-adc'  
181     AWS_S3_REGION_NAME = 'us-east-1'  
182     AWS_ACCESS_KEY_ID = os.environ.get('AWS_ACCESS_KEY_ID')  
183     AWS_SECRET_ACCESS_KEY = os.environ.get('AWS_SECRET_ACCESS_KEY')  
184     AWS_S3_CUSTOM_DOMAIN = f'{AWS_STORAGE_BUCKET_NAME}.s3.amazonaws.com'  
185  
186     # Static and media files  
187     STATICFILES_STORAGE = 'custom_storages.StaticStorage'  
188     STATICFILES_LOCATION = 'static'  
189     DEFAULT_FILE_STORAGE = 'custom_storages.MediaStorage'  
190     MEDIAFILES_LOCATION = 'media'  
191  
192     # Override static and media URLs in production  
193     STATIC_URL = f'https://{{AWS_S3_CUSTOM_DOMAIN}}/{{STATICFILES_LOCATION}}/  
194     MEDIA_URL = f'https://{{AWS_S3_CUSTOM_DOMAIN}}/{{MEDIAFILES_LOCATION}}/'  
195  
196
```

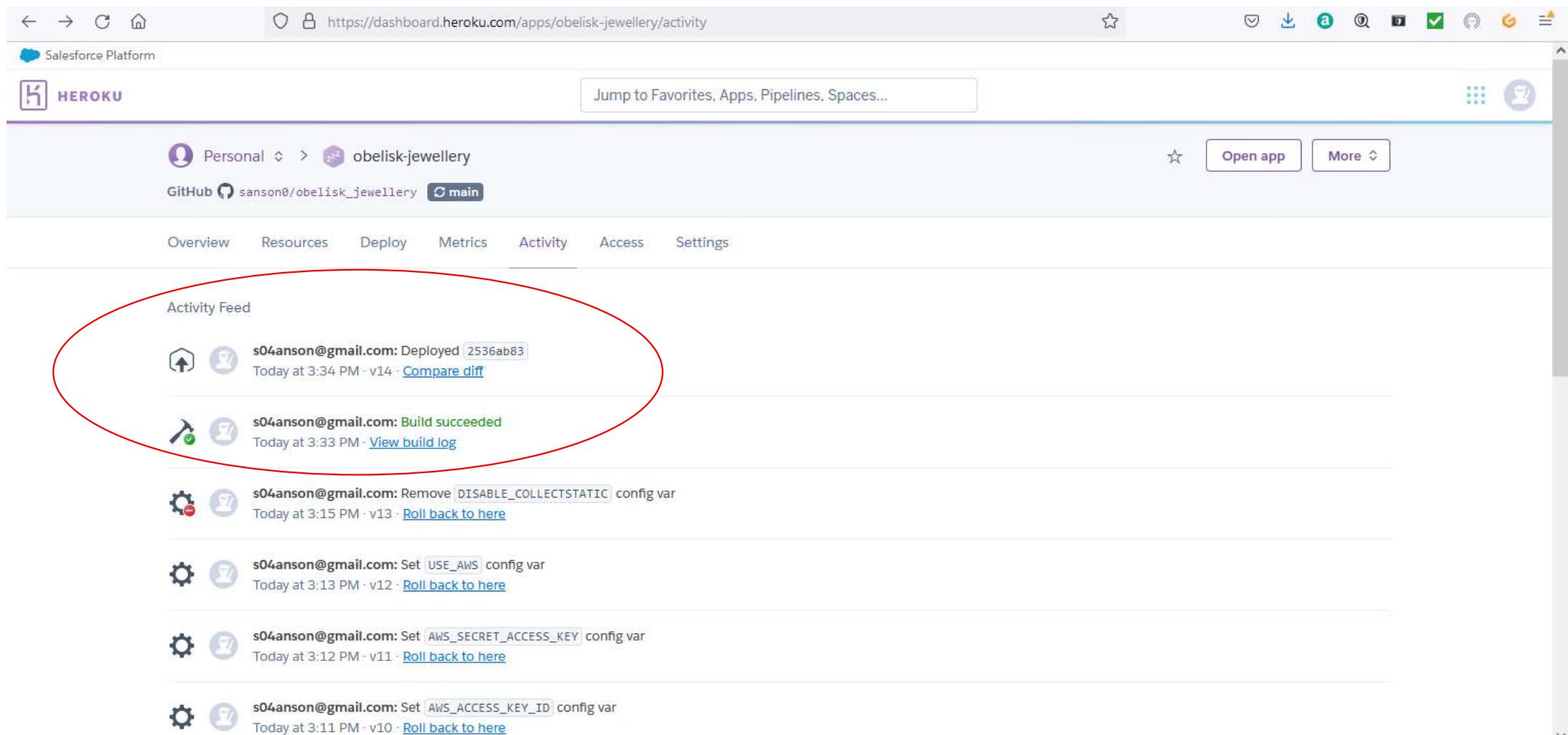
When the project is deployed to Heroku, Heroku will run `python3 manage.py collectstatic` during the build process, which will search through all apps and project folders looking for static files.

It will use the s3 custom domain setting here in conjunction with our custom storage classes that tell it the location at that URL, where we'd like to save things.

When the `USE_AWS` setting is true, whenever `collectstatic` is run, static files will be collected into a static folder in our s3 bucket. The beauty of this is that it's all automatic.

To make sure it works, add all these changes and commit them.

# Automatic deploys is set up so committing and pushing changes to Github causes Heroku to build and deploy



The screenshot shows the Heroku dashboard for the app 'obelisk-jewellery'. The activity feed is displayed, with the first entry circled in red. The entry shows a deployment from GitHub at 3:34 PM today, with a link to compare the diff.

Activity Feed

-  s04anson@gmail.com: Deployed 2536ab83  
Today at 3:34 PM · v14 · [Compare diff](#)
-  s04anson@gmail.com: Build succeeded  
Today at 3:33 PM · [View build log](#)
-  s04anson@gmail.com: Remove DISABLE\_COLLECTSTATIC config var  
Today at 3:15 PM · v13 · [Roll back to here](#)
-  s04anson@gmail.com: Set USE\_AWS config var  
Today at 3:13 PM · v12 · [Roll back to here](#)
-  s04anson@gmail.com: Set AWS\_SECRET\_ACCESS\_KEY config var  
Today at 3:12 PM · v11 · [Roll back to here](#)
-  s04anson@gmail.com: Set AWS\_ACCESS\_KEY\_ID config var  
Today at 3:11 PM · v10 · [Roll back to here](#)

# Build log

The screenshot shows the Heroku dashboard interface. At the top, there's a navigation bar with icons for back, forward, refresh, and search, followed by the URL <https://dashboard.heroku.com/apps/obelisk-jewellery/activity/builds/102e3612-8bb5-4c53-a18e-d11e10c8d8be>. To the right of the URL are several small icons for sharing, saving, and other actions. Below the URL is a "Salesforce Platform" logo.

The main header features the "HEROKU" logo on the left and a "Jump to Favorites, Apps, Pipelines, Spaces..." search bar on the right. Underneath the header, the app name "obelisk-jewellery" is displayed along with a GitHub icon and the repository URL "GitHub sanson0/obelisk\_jewellery". A "main" tab is selected, indicated by a dark blue background.

The navigation menu below includes links for Overview, Resources, Deploy, Metrics, Activity, Access, and Settings. The "Activity" link is currently active, highlighted in blue.

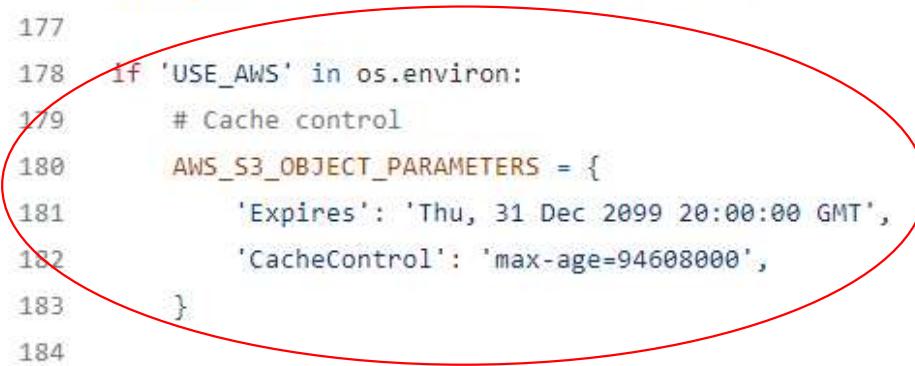
The main content area is titled "Activity Feed > Build Log" and shows a log ID "ID 102e3612-8bb5-4c53-a18e-d11e10c8d8be". The log itself contains the following text:

```
Installing collected packages: asgiref, jmespath, six, python-dateutil, urllib3, botocore, s3transfer, boto3, dj-database-url, pytz, sqlparse, Django, defusedxml, python3-openid, oauthlib, charset-normalizer, certifi, idna, requests, requests-oauthlib, django-allauth, django-countries, django-crispy-forms, django-social-share, django-storages, gunicorn, Pillow, psycopg2-binary, stripe
Successfully installed Django-3.2.7 Pillow-8.3.2 asgiref-3.4.1 boto3-1.18.63 botocore-1.21.63 certifi-2021.10.8 charset-normalizer-2.0.7 defusedxml-0.7.1 dj-database-url-0.5.0 django-allauth-0.41.0 django-countries-7.2.1 django-crispy-forms-1.12.0 django-social-share-2.2.1 django-storages-1.12.2 gunicorn-20.1.0 idna-3.3 jmespath-0.10.0 oauthlib-3.1.1 psycopg2-binary-2.9.1 python-dateutil-2.8.2 python3-openid-3.2.0 pytz-2021.1 requests-2.26.0 requests-oauthlib-1.3.0 s3transfer-0.5.0 six-1.16.0 sqlparse-0.4.1 stripe-2.60.0 urllib3-1.26.7
----> $ python manage.py collectstatic --noinput
      134 static files copied.
----> Discovering process types
      Procfile declares types -> web
----> Compressing...
      Done: 82.6M
----> Launching...
      Released v14
      https://obelisk-jewellery.herokuapp.com/ deployed to Heroku
```

At the bottom of the log area, it says "Build finished".

## Part 6

### Add cache control to improve performance



```
173 STATICFILES_DIRS = (os.path.join(BASE_DIR, 'static'),)
174
175 MEDIA_URL = '/media/'
176 MEDIA_ROOT = os.path.join(BASE_DIR, 'media')
177
178 if 'USE_AWS' in os.environ:
179     # Cache control
180     AWS_S3_OBJECT_PARAMETERS = {
181         'Expires': 'Thu, 31 Dec 2099 20:00:00 GMT',
182         'CacheControl': 'max-age=94608000',
183     }
184
185     # Bucket Config
186     AWS_STORAGE_BUCKET_NAME = 'ckz8780-boutique-ado'
187     AWS_S3_REGION_NAME = 'us-east-1'
188     AWS_ACCESS_KEY_ID = os.environ.get('AWS_ACCESS_KEY_ID')
189     AWS_SECRET_ACCESS_KEY = os.environ.get('AWS_SECRET_ACCESS_KEY')
```

This will tell the browser that it's okay to cache static files for a long time since they don't change very often, and this will improve performance for our users.

Commit and push changes to Github

Activity Feed

-  s04anson@gmail.com: Build in progress  
Today at 5:07 PM · [View build progress](#)
-  s04anson@gmail.com: Deployed 2536ab83  
Today at 3:34 PM · v14 · [Compare diff](#)
-  s04anson@gmail.com: Build succeeded  
Today at 3:33 PM · [View build log](#)
-  s04anson@gmail.com: Remove DISABLE\_COLLECTSTATIC config var  
Today at 3:15 PM · v13 · [Roll back to here](#)
-  s04anson@gmail.com: Set USE\_AWS config var  
Today at 3:13 PM · v12 · [Roll back to here](#)
-  s04anson@gmail.com: Set AWS\_SECRET\_ACCESS\_KEY config var  
Today at 3:12 PM · v11 · [Roll back to here](#)

# Build in progress

Activity Feed

-  s04anson@gmail.com: Deployed 0568fb4d  
Just now · v15 · [Compare diff](#)
-  s04anson@gmail.com: Build succeeded  
Today at 5:07 PM · [View build log](#)
-  s04anson@gmail.com: Deployed 2536ab83  
Today at 3:34 PM · v14 · [Roll back to here](#) · [Compare diff](#)
-  s04anson@gmail.com: Build succeeded  
Today at 3:33 PM · [View build log](#)
-  s04anson@gmail.com: Remove DISABLE\_COLLECTSTATIC config var  
Today at 3:15 PM · v13 · [Roll back to here](#)
-  s04anson@gmail.com: Set USE\_AWS config var  
Today at 3:13 PM · v12 · [Roll back to here](#)

Build succeeded and deployed

# Go back to s3 in AWS, to create a new folder called ‘media’.

The screenshot shows the AWS S3 console interface. The URL in the browser is <https://s3.console.aws.amazon.com/s3/buckets/obelisk-jewellery?region=eu-west-2&tab=objects>. The left sidebar has a 'Buckets' section with links to Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and Access analyzer for S3. Below that is a 'Block Public Access settings for this account' section. Under 'Storage Lens', there are links to Dashboards and AWS Organizations settings. A 'Feature spotlight' section shows 3 items. At the bottom, there are links for Feedback, English (US), Privacy Policy, Terms of Use, and Cookie preferences.

Amazon S3 > obelisk-jewellery

**obelisk-jewellery** Info

Publicly accessible

Objects Properties Permissions Metrics Management Access Points

**Objects (1)**

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	static/	Folder	-	-	-

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Call the folder 'media'. Click 'Create folder'

The screenshot shows the AWS S3 console interface. On the left, there's a sidebar with various navigation options like Buckets, Storage Lens, and Feature spotlight. The main area is titled 'Folder' and contains a form for creating a new folder. The 'Folder name' field has 'media' typed into it. Below the field, a note says 'Folder names can't contain "/'. See rules for naming.' In the 'Server-side encryption' section, there's a note that the settings apply only to the new folder object. Two radio buttons are shown: 'Disable' (which is selected) and 'Enable'. At the bottom right of the form, there are 'Cancel' and 'Create folder' buttons. The 'Create folder' button is highlighted with a red oval. A warning message box is also visible, stating that 'Your bucket policy might block folder creation' and providing a link to 'upload configuration'.

Your bucket policy might block folder creation  
If your bucket policy prevents uploading objects without specific tags, metadata, or access control list (ACL) grantees, you will not be able to create a folder using this configuration. Instead, you can use the [upload configuration](#) to upload an empty folder and specify the appropriate settings.

**Folder**

Folder name

media /

Folder names can't contain "/". See rules for naming.

**Server-side encryption**

The following settings apply only to the new folder object and not to the objects contained within it.

Server-side encryption

Disable

Enable

Create folder

Feedback English (US) ▾

© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Privacy Policy Terms of Use Cookie preferences

# Click on 'media' folder

The screenshot shows the AWS S3 console interface. On the left, the navigation pane includes 'Buckets', 'Access Points', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', 'Access analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens' (with 'Dashboards' and 'AWS Organizations settings'), 'Feature spotlight' (with a blue notification badge), and 'AWS Marketplace for S3'. The main content area displays a green success message: 'Successfully created folder "media" Operation successfully completed.' Below this, the bucket name 'obelisk-jewellery' is shown with a 'Info' link, and it is labeled as 'Publicly accessible'. The 'Objects' tab is selected, showing 'Objects (2)'. The table lists two items: 'media/' (Folder) and 'static/' (Folder). The 'media/' item is circled in red. A toolbar above the table includes 'Upload' (orange button), 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', and 'Create folder'. A search bar at the top of the table says 'Find objects by prefix'. The footer contains links for 'Feedback', 'English (US)', 'Privacy Policy', 'Terms of Use', and 'Cookie preferences'.

Name	Type	Last modified	Size	Storage class
media/	Folder	-	-	-
static/	Folder	-	-	-

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

# Click on 'Upload'

The screenshot shows the AWS S3 console interface. The URL in the browser is <https://s3.console.aws.amazon.com/s3/buckets/obelisk-jewellery?region=eu-west-2&prefix=media/&showversions=false>. The left sidebar has sections for Buckets, Storage Lens, and Feature spotlight. The main content area shows the 'media/' folder under 'Amazon S3 > obelisk-jewellery > media/'. The 'Objects' tab is selected. The 'Actions' bar at the top includes buttons for Copy S3 URI, Copy URL, Download, Open, Delete, Create folder, and Upload. The 'Upload' button is highlighted with a red oval. A search bar says 'Find objects by prefix'. Below it is a table header with columns: Name, Type, Last modified, Size, and Storage class. The table body displays 'No objects' and 'You don't have any objects in this folder.' A large 'Upload' button is located at the bottom of the table area.

Either click 'Add files' or drag and drop all the product images.

The screenshot shows the AWS S3 console at the URL <https://s3.console.aws.amazon.com/s3/upload/obelisk-jewellery?region=eu-west-2&prefix=media/>. The page title is "Upload". The breadcrumb navigation shows: Amazon S3 > obelisk-jewellery > media/ > Upload. The main area is titled "Upload" with a "Info" link. It contains instructions: "Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)". Below this is a dashed box with the text "Drag and drop files and folders you want to upload here, or choose Add files, or Add folders.". A table titled "Files and folders (0)" shows a single row with a "Remove" button. The table has columns: Name, Folder, Type, and Size. A search bar "Find by name" is above the table. The message "No files or folders" is displayed below the table, followed by the sub-instruction "You have not chosen any files or folders to upload." At the bottom, there is a "Destination" section which is currently collapsed. The footer includes links for Feedback, English (US), © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved., Privacy Policy, Terms of Use, and Cookie preferences.

Feedback English (US) ▾ https://s3.console.aws.amazon.com/s3/upload/obelisk-jewellery?region=eu-west-2&prefix=media/ marbles76 ▾ Global ▾ Support ▾

Services ▾ Search for services, features, marketplace products, and docs [Alt+S]

Amazon S3 > obelisk-jewellery > media/ > Upload

## Upload Info

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add files**, or **Add folders**.

Files and folders (99 Total, 5.4 MB)						
	Name	Folder	Type	Size		
<input type="checkbox"/>	a-bracelet-5003797_640.png	-	image/png	53.3 KB		
<input type="checkbox"/>	a-heart-3410020_640.jpg	-	image/jpeg	19.5 KB		
<input type="checkbox"/>	accessories-5644343_640.jpg	-	image/jpeg	15.7 KB		
<input type="checkbox"/>	accessory-219346_640.jpg	-	image/jpeg	11.9 KB		

All files and folders in this table will be uploaded.

Find by name < 1 2 3 4 5 6 7 ... 10 >

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Now click Next and under manage public permissions  
Select grant public read access to these objects.

The screenshot shows the AWS S3 console interface for uploading files to a bucket named 'obelisk-jewellery'. The left sidebar lists several files being uploaded:

File	Type	Format	Size
amazonite-665747_640.jpg	-	image/jpeg	39.0 KB
amber-669473_640.jpg	-	image/jpeg	23.9 KB
amethyst-2186842_640.jpg	-	image/jpeg	23.2 KB
amonite-665339_640.jpg	-	image/jpeg	48.1 KB
ankh-665720_640.jpg	-	image/jpeg	15.6 KB

In the center, the 'Destination' section shows the target bucket and path: `s3://obelisk-jewellery/media/`. Below this, the 'Destination details' section provides information about bucket settings for new objects.

The 'Permissions' section is circled in red and contains the following text:

Grant public access and access to other AWS accounts.

Below the 'Permissions' section are the 'Properties' section and the 'Upload' button.

At the bottom of the page, there are links for Feedback, English (US), Copyright notice (© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.), Privacy Policy, Terms of Use, and Cookie preferences.

# Grant public read access and tick box

The screenshot shows the AWS S3 console interface for granting public access to an object. A red oval highlights the 'Grant public-read access' section, which is the focus of the question.

**Access control list (ACL)**

Grant basic read/write permissions to other AWS accounts. [Learn more](#)

**AWS recommends using S3 bucket policies or IAM policies for access control.** [Learn more](#)

**Access control list (ACL)**

Choose from predefined ACLs  
 Specify individual ACL permissions

**Predefined ACLs**

Private (recommended)  
Only the object owner will have read and write access.

Grant public-read access  
Anyone in the world will be able to access the specified objects. The object owner will have read and write access. [Learn more](#)

**Granting public-read access is not recommended**  
Anyone in the world will be able to access the specified objects. [Learn more](#)

I understand the risk of granting public-read access to the specified objects.

**Properties**  
Specify storage class, encryption settings, tags, and more.

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Click next through to the end here. And then click upload

The screenshot shows the AWS S3 console interface. At the top, there's a navigation bar with icons for back, forward, refresh, and home, followed by the URL <https://s3.console.aws.amazon.com/s3/upload/obelisk-jewellery?region=eu-west-2&prefix=media/>. To the right of the URL are several browser extensions and user profile information for 'marbles76'.

The main content area has a dark blue header with the text 'Uploading' and a progress bar indicating 14% completion. Below the header, status information is displayed: 'Total remaining: 75 files: 4.7 MB(85.79%)', 'Estimated time remaining: a minute', and 'Transfer rate: 96.0 KB/s'. On the far right of this header is a 'Cancel' button.

A modal window titled 'Upload: status' is open in the center. It contains a message: 'The information below will no longer be available after you navigate away from this page.' Below this, there's a 'Summary' section with three categories: 'Destination' (s3://obelisk-jewellery/media/), 'Succeeded' (23 files, 791.0 KB (14.21%)), and 'Failed' (0 files, 0 B (0%)).

At the bottom of the modal, there are tabs for 'Files and folders' (which is selected) and 'Configuration'. Below these tabs, a large table is partially visible with the heading 'Files and folders (99 Total, 5.4 MB)'.

At the very bottom of the page, there are links for 'Feedback', 'English (US)', 'Privacy Policy', 'Terms of Use', and 'Cookie preferences'.



Check the email address for the superuser on the Postgres database, verified and primary should be ticked. Checked through the admin screen. User won't be able to log in to the store without this.

The screenshot shows the Django administration interface for changing an email address. The URL is <https://obelisk-jewellery.herokuapp.com/admin/account/emailaddress/1/change/>. The page title is "Django administration". The top navigation bar includes "WELCOME, SLA7604. VIEW SITE / CHANGE PASSWORD / LOG OUT". The left sidebar has sections for ACCOUNTS (Email addresses, Groups, Users), AUTHENTICATION AND AUTHORIZATION (Groups, Users), CHECKOUT (Orders), PRODUCTS (Categories, Products), and SITES (Sites). The main content area is titled "Change email address" for user "s04anson@gmail.com". It shows a search field with "User: 1" and "s04anson@gmail.com" in the "E-mail address:" field. Below these fields are two checkboxes: "Verified" (checked) and "Primary" (checked). A red circle highlights both checkboxes. At the bottom are buttons for "Delete", "Save and add another", "Save and continue editing", and "SAVE".

← → 🔍 https://obelisk-jewellery.herokuapp.com/admin/account/emailaddress/ ⭐ ? 🌐 ⭐ 🗑️ 🚙

# Django administration

WELCOME, **SLA7604**. [VIEW SITE / CHANGE PASSWORD / LOG OUT](#)

Home > Accounts > Email addresses

**ACCOUNTS**

Email addresses [+ Add](#)

**AUTHENTICATION AND AUTHORIZATION**

Groups [+ Add](#)

Users [+ Add](#)

**CHECKOUT**

Orders [+ Add](#)

« [PRODUCTS](#)

Categories [+ Add](#)

Products [+ Add](#)

**SITES**

Sites [+ Add](#)

The email address "s04anson@gmail.com" was changed successfully.

Select email address to change [ADD EMAIL ADDRESS +](#)

Action:   0 of 2 selected

<input type="checkbox"/> E-MAIL ADDRESS	USER	PRIMARY	VERIFIED
<input type="checkbox"/> test@test.com	testytest	✓	✓
<input type="checkbox"/> s04anson@gmail.com	sla7604	✓	✓

2 email addresses

**FILTER**

By primary

All  
Yes  
No

By verified

All  
Yes  
No

Next, add stripe keys to the Heroku config variables. Go to stripe, login and click on 'Developers'

The screenshot shows the Stripe Developers dashboard at <https://dashboard.stripe.com/test/developers>. A red circle highlights the 'Developers' button in the top right corner of the header.

**TEST DATA**

**Developers**

**Overview**

- API keys
- Webhooks
- Events
- Logs
- Extensions

**Your integration**

**API requests**

Successful	Failed
18	0

11 Oct Today

**API error distribution**

GET	POST	DELETE
0	0	0

11 Oct Today

**Webhooks**

Successful	Failed
0	30

11 Oct Today

**Webhooks response time**

Min	Avg	Max
1.1 s	7.3 s	11 s

11 Oct Today

**Integration status**  
**Activate your account** Step 2 / 4

Use these test cards to test your integration. Once you are ready, [activate your account](#) to access live API keys.

Successful payment ...4242

Failed payment ...9995

Requires authentication ...3155

[Learn about testing](#)

**Recent errors** 2

# Click on API keys, and collect the public and secret key from stripe

The screenshot shows the Stripe dashboard at <https://dashboard.stripe.com/test/apikeys>. The top navigation bar includes links for Home, Payments, Balances, Customers, Products, Reports, Connect, More, Developers (which is selected), and Test mode (which is turned on). A red oval highlights the 'API keys' tab in the left sidebar, which is currently active. The main content area is titled 'API keys' and displays two types of keys: 'Standard keys' and 'Restricted keys'. Under 'Standard keys', there are two entries: 'Publishable key' (TOKEN: Hidden) and 'Secret key' (TOKEN: [REDACTED], with a 'Reveal test key' button). A large red oval encircles both these key entries. The 'Restricted keys' section is partially visible below.

NAME	TOKEN	LAST USED	CREATED	...
Publishable key	Hidden	15 Oct	24 Sep	...
Secret key	[REDACTED]	15 Oct	24 Sep	...

**Standard keys**  
These keys will allow you to authenticate API requests. [Learn more](#)

NAME	TOKEN	LAST USED	CREATED	...
Publishable key	Hidden	15 Oct	24 Sep	...
Secret key	[REDACTED]	15 Oct	24 Sep	...

**Restricted keys**  
For greater security, you can create restricted API keys that limit access and permissions for different areas of your account data. [Learn more](#)

NAME	TOKEN	LAST USED	CREATED	...
------	-------	-----------	---------	-----

Collect these two keys and added them to config variables in Heroku

# Go back to Heroku and add the keys into config variables

The screenshot shows the Heroku dashboard for the app "obelisk-jewellery". The top navigation bar includes links for Overview, Resources, Deploy, Metrics, Activity, Access, and Settings. The "Settings" tab is currently selected. The main content area displays "App Information" with details such as App Name (obelisk-jewellery), Region (Europe), Stack (heroku-20), Framework (Python), Slug size (74.1 MiB of 500 MiB), GitHub repo (sanson0/obelisk\_jewellery), and Heroku git URL (<https://git.heroku.com/obelisk-jewellery.git>). Below this, the "Config Vars" section contains a note about config vars and a "Reveal Config Vars" button, which is highlighted with a red oval. The URL in the browser address bar is <https://dashboard.heroku.com/apps/obelisk-jewellery/settings>.

# Add keys into Heroku

The screenshot shows the Heroku dashboard for the app "obelisk-jewellery". The URL in the browser is <https://dashboard.heroku.com/apps/obelisk-jewellery/settings>. The page displays configuration variables (Config Vars) and buildpacks.

**Config Vars**

Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.

KEY	VALUE	
AWS_ACCESS_KEY_ID	Hidden	<input type="button" value="Edit"/> <input type="button" value="X"/>
AWS_SECRET_ACCESS_KEY	Hidden	<input type="button" value="Edit"/> <input type="button" value="X"/>
DATABASE_URL	Hidden	<input type="button" value="Edit"/> <input type="button" value="X"/>
SECRET_KEY	Hidden	<input type="button" value="Edit"/> <input type="button" value="X"/>
USE_AWS	True	<input type="button" value="Edit"/> <input type="button" value="X"/>
STRIPE_PUBLIC_KEY	Hidden	<input type="button" value="Edit"/> <input type="button" value="X"/>
KEY	VALUE	<input type="button" value="Add"/>

**Buildpacks**

Buildpacks are scripts that are run when your

The screenshot shows the Heroku Settings page for the 'obelisk-jewellery' app. The left sidebar has a 'Salesforce Platform' entry. The main area displays configuration variables (Config Vars) with their values and visibility status (Hidden). A 'Hide Config Vars' button is visible at the top right of the list.

KEY	VALUE	HIDDEN
AWS_ACCESS_KEY_ID		Hidden
AWS_SECRET_ACCESS_KEY		Hidden
DATABASE_URL	postgres://	Hidden
SECRET_KEY		Hidden
USE_AWS	True	
STRIPE_PUBLIC_KEY		Hidden
STRIPE_SECRET_KEY		Hidden

**Config Vars**  
Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.

**Config Vars**

**Hide Config Vars**

Go back to Stripe and click on ‘Webhooks’ in the Developers menu. Next, click on ‘Add endpoint’. At the moment, webhooks are going to the Gitpod workspace.

The screenshot shows the Stripe Developers dashboard with the 'Webhooks' section selected. A red circle highlights the 'Webhooks' link in the sidebar. Another red circle highlights the '+ Add endpoint' button in the 'Hosted endpoints' section. The 'TEST DATA' tab is also visible at the top of the page.

**Developers**

**Webhooks**

**Hosted endpoints**

URL	TYPE	LAST 7 DAYS	ERROR RATE	STATUS
https://8000-green-hummingbird-2gjepeu6.ws-eu18.gitpod.io/checkout/wh/	Direct		100%	Active
https://8000-green-hummingbird-2gjepeu6.ws-eu18.gitpod.io/checkout/wh/	Direct		100%	Active

**Local listeners**

+ Add local listener

Listen to live Stripe events and forward them to your local device using the Stripe CLI.

# Type in the url for the Heroku app where the project is deployed

The screenshot shows a web browser window with the URL <https://dashboard.stripe.com/test/webhooks/create>. The page title is "Listen to Stripe events". On the left, there's a form to add a webhook endpoint. The "Endpoint URL" field contains "https://", which is circled in red. Below it is a "Description" field with placeholder text "An optional description of what this webhook endpoint is used for...". There's also a checkbox for "Listen to events on Connected accounts" and a "Select events" button. At the bottom are "Add endpoint" and "Cancel" buttons. On the right, there's a "Sample endpoint" section containing a Ruby code snippet:

```
1 # server.rb
2 #
3 # Use this sample code to handle webhook events in your integration.
4 #
5 # 1) Paste this code into a new file (server.rb)
6 #
7 # 2) Install dependencies
8 #   gem install sinatra
9 #   gem install stripe
10 #
11 # 3) Run the server on http://localhost:4242
12 #   ruby server.rb
13
14 require 'json'
15 require 'sinatra'
16 require 'stripe'
17
18 # This is your Stripe CLI webhook secret for testing your endpoint locally.
19 endpoint_secret = 'whsec_auUmZ0Yi9ItrHkwXQihizMN24iv4DdpB'
20
21 set :port, 4242
22
23 post '/webhook' do
24   payload = request.body.read
25   sig_header = request.env['HTTP_STRIPE_SIGNATURE']
26   event = nil
27
28   begin
```

# Click 'Select events'

The screenshot shows the Stripe dashboard at <https://dashboard.stripe.com/test/webhooks/create>. The page title is "Listen to Stripe events". It includes sections for "Add an endpoint" (button), "Test in a local environment" (button), "Endpoint URL" (input field containing <https://obelisk-jewellery.herokuapp.com/checkout/wh/>), "Description" (input field placeholder "An optional description of what this webhook endpoint is used for..."), "Listen to events on Connected accounts" (checkbox), and "Select events to listen to" (button circled in red). Below these are "Add endpoint" and "Cancel" buttons.

Sample endpoint   Received events   Ruby

```
1 # server.rb
2 #
3 # Use this sample code to handle webhook events in your integration.
4 #
5 # 1) Paste this code into a new file (server.rb)
6 #
7 # 2) Install dependencies
8 #   gem install sinatra
9 #   gem install stripe
10 #
11 # 3) Run the server on http://localhost:4242
12 #   ruby server.rb
13
14 require 'json'
15 require 'sinatra'
16 require 'stripe'
17
18 # This is your Stripe CLI webhook secret for testing your endpoint locally.
19 endpoint_secret = 'whsec_auUmZ0Yi9ItrHkwXQihizMN24iv4DdpB'
20
21 set :port, 4242
22
23 post '/webhook' do
24   payload = request.body.read
25   sig_header = request.env['HTTP_STRIPE_SIGNATURE']
26   event = nil
27
28   begin
```

# Tick box for 'Select all events'

The screenshot shows the Stripe dashboard at <https://dashboard.stripe.com/test/webhooks/create>. The user is on the 'Select events to send' step. A red oval highlights the 'Select all events' checkbox, which is located above a list of event categories. Below the categories are two buttons: 'Add events' (blue) and 'Cancel'.

Sample endpoint   Received events   Ruby

```
1 # server.rb
2 #
3 # Use this sample code to handle webhook events in your integration.
4 #
5 # 1) Paste this code into a new file (server.rb)
6 #
7 # 2) Install dependencies
8 #   gem install sinatra
9 #   gem install stripe
10 #
11 # 3) Run the server on http://localhost:4242
12 #   ruby server.rb
13
14 require 'json'
15 require 'sinatra'
16 require 'stripe'
17
18 # This is your Stripe CLI webhook secret for testing your endpoint locally.
19 endpoint_secret = 'whsec_aUUmZ0Yi9ItrHkwXQihizMN24iv4DdpB'
20
21 set :port, 4242
22
23 post '/webhook' do
24   payload = request.body.read
25   sig_header = request.env['HTTP_STRIPE_SIGNATURE']
26   event = nil
27
28   begin
```

# Click 'Add endpoint'

The screenshot shows the Stripe dashboard at <https://dashboard.stripe.com/test/webhooks/create>. On the left, there's a sidebar with a list of event types: subscription\_schedule.updated, tax\_rate.created, tax\_rate.updated, topup.canceled, topup.created, topup.failed, topup.reversed, topup.succeeded, transfer.created, transfer.failed, transfer.paid, transfer.reversed, and transfer.updated. Below this list are two buttons: 'Change events' and 'Add endpoint'. The 'Add endpoint' button is circled in red. On the right, there's a code editor window titled 'Ruby' showing a sample endpoint for handling various Stripe events. The code uses case statements to handle different event types like account.updated, charge.captured, and charge.refunded.

```
Sample endpoint Received events Ruby
43     case event.type
44     when 'account.updated'
45         account = event.data.object
46     when 'account.external_account.created'
47         external_account = event.data.object
48     when 'account.external_account.deleted'
49         external_account = event.data.object
50     when 'account.external_account.updated'
51         external_account = event.data.object
52     when 'balance.available'
53         balance = event.data.object
54     when 'billing_portal.configuration.created'
55         configuration = event.data.object
56     when 'billing_portal.configuration.updated'
57         configuration = event.data.object
58     when 'capability.updated'
59         capability = event.data.object
60     when 'charge.captured'
61         charge = event.data.object
62     when 'charge.expired'
63         charge = event.data.object
64     when 'charge.failed'
65         charge = event.data.object
66     when 'charge.pending'
67         charge = event.data.object
68     when 'charge.refunded'
69         charge = event.data.object
70     when 'charge.succeeded'
```

# Reveal webhook signing secret

The screenshot shows the Stripe Dashboard at [https://dashboard.stripe.com/test/webhooks/we\\_1Jm1Q7Gl8FO5pWvHf0TmQpgE](https://dashboard.stripe.com/test/webhooks/we_1Jm1Q7Gl8FO5pWvHf0TmQpgE). The top navigation bar includes links for Home, Payments, Balances, Customers, Products, Reports, Connect, More, Developers (selected), and Test mode (on). The main content area is titled "WEBHOOK" and shows a webhook endpoint: <https://obelisk-jewellery.herokuapp.com/checkout/wh/>. The webhook status is "Enabled". The "Status" section shows "Listening for 180 events" and "API version 2020-08-27". The "Signing secret" section has a red oval around the "Reveal" button. Below the endpoint, it says "Waiting for events..." and "Send a test event to test your webhook endpoint." A blue button labeled "Send a test event" is visible.

New Business [Activate account →](#) Search... Create ⚙️ Help 📡 Settings 🚙

Home Payments Balances Customers Products Reports Connect More [Developers](#) Test mode

WEBHOOK

<https://obelisk-jewellery.herokuapp.com/checkout/wh/>

Status Listening for 180 events API version 2020-08-27 [Reveal](#)

Events [View logs](#)

Extensions

**Waiting for events...**  
Send a test event to test your webhook endpoint.

[Send a test event](#)

# Webhook secret key copied and pasted into Heroku

The screenshot shows the Stripe Dashboard at [https://dashboard.stripe.com/test/webhooks/we\\_1Jm1Q7Gl8FO5pWvHf0TmQpgE](https://dashboard.stripe.com/test/webhooks/we_1Jm1Q7Gl8FO5pWvHf0TmQpgE). The top navigation bar includes links for New Business, Activate account, Search, Create, Help, and account settings. Below the navigation is a menu with Home, Payments, Balances, Customers, Products, Reports, Connect, More, Developers (which is selected), and Test mode (which is off). A red button labeled TEST DATA is visible.

The main area is titled "Developers" and "WEBHOOK". It displays a webhook endpoint URL: <https://obelisk-jewellery.herokuapp.com/checkout/wh/>. To the right of the URL is a "we\_1Jm1Q7Gl8FO5pWvHf0TmQpgE" ID and a "Send test event" button. Below the URL, there are sections for Status (Enabled), Listening for 180 events, API version 2020-08-27, and Signing secret (which is circled in red and labeled "Hidden"). There are also Configuration and View logs links.

Below the webhook details, a section titled "Waiting for events..." with the sub-instruction "Send a test event to test your webhook endpoint." contains a "Send a test event" button.

Config Vars

Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.

Config Vars

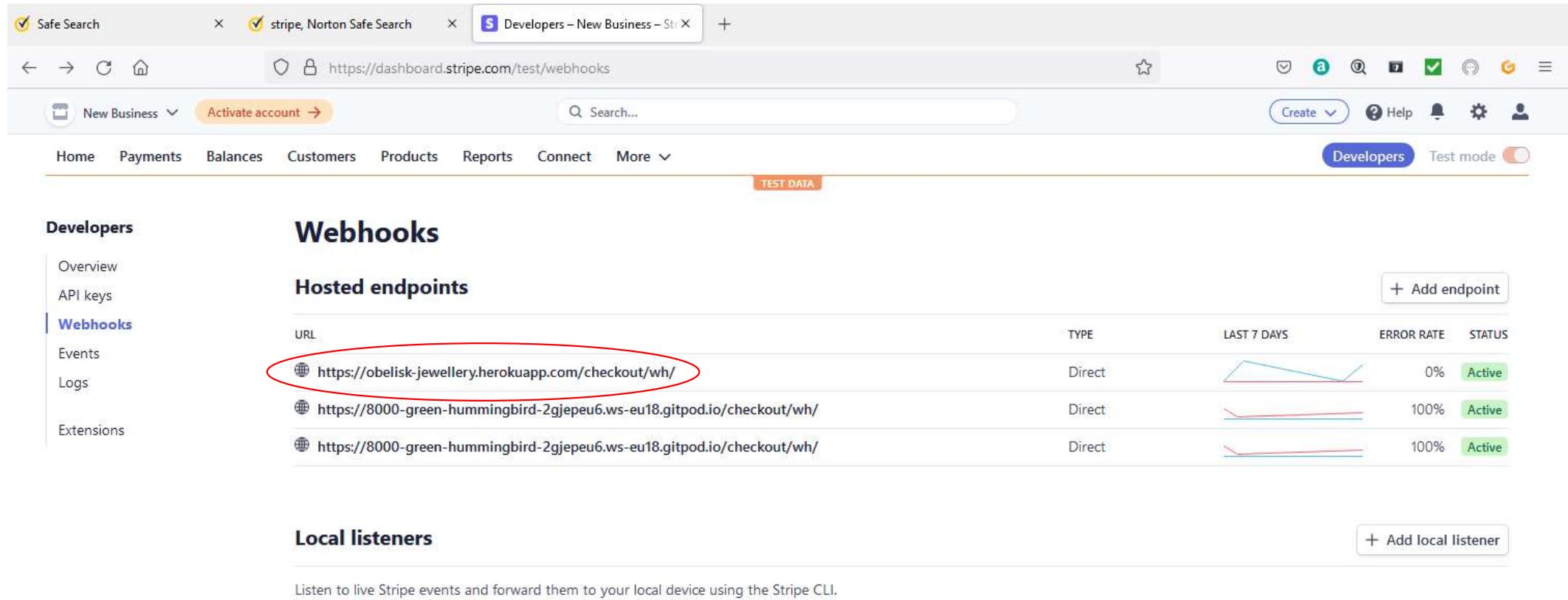
AWS\_ACCESS\_KEY\_ID      Hidden  
AWS\_SECRET\_ACCESS\_KEY      Hidden  
DATABASE\_URL      postgres://      Hidden  
SECRET\_KEY      Hidden  
USE\_AWS      True  
STRIPE\_PUBLIC\_KEY      Hidden  
STRIPE\_SECRET\_KEY      Hidden  
STRIPE\_WH\_SECRET      Hidden

KEY      VALUE      Add



Check that the variable names match those in the settings.py file of the project

# Click on url



The screenshot shows the Stripe Developers dashboard under the 'Webhooks' tab. The left sidebar includes links for Overview, API keys, Webhooks (which is selected and highlighted in blue), Events, Logs, and Extensions. The main content area is titled 'Webhooks' and contains two sections: 'Hosted endpoints' and 'Local listeners'. The 'Hosted endpoints' section lists three entries, each with a URL, type (Direct), and status (Active). The first URL, <https://obelisk-jewellery.herokuapp.com/checkout/wh/>, is circled in red.

URL	Type	Last 7 Days	Error Rate	Status
<a href="https://obelisk-jewellery.herokuapp.com/checkout/wh/">https://obelisk-jewellery.herokuapp.com/checkout/wh/</a>	Direct		0%	Active
<a href="https://8000-green-hummingbird-2gjepeu6.ws-eu18.gitpod.io/checkout/wh/">https://8000-green-hummingbird-2gjepeu6.ws-eu18.gitpod.io/checkout/wh/</a>	Direct		100%	Active
<a href="https://8000-green-hummingbird-2gjepeu6.ws-eu18.gitpod.io/checkout/wh/">https://8000-green-hummingbird-2gjepeu6.ws-eu18.gitpod.io/checkout/wh/</a>	Direct		100%	Active

**Hosted endpoints**

+ Add endpoint

**Local listeners**

+ Add local listener

# Click on 'Send test event'

The screenshot shows the Stripe Developers dashboard under the 'New Business' tab. A specific webhook configuration is displayed for the URL <https://obelisk-jewellery.herokuapp.com/checkout/wh/>. The 'Webhooks' section is active, showing the status as 'Enabled', listening for 180 events since 2020-08-27, and a signing secret available for reveal. The 'TEST DATA' tab is selected. On the right, a detailed view of a received webhook event for 'reporting.report\_type.updated' is shown, with a 'Response' of '200 OK' and a log entry indicating an unhandled webhook. A red circle highlights the 'Send test event' button next to the webhook URL.

Safe Search stripe, Norton Safe Search Developers – New Business – St

Home Payments Balances Customers Products Reports Connect More TEST DATA

Developers Test mode

WEBHOOK

<https://obelisk-jewellery.herokuapp.com/checkout/wh/>

Status Enabled Listening for 180 events API version 2020-08-27 Signing secret Reveal Configuration View logs

All Succeeded Failed

TODAY

- ✓ reporting.report\_type.updated 5:02:05 AM
- ✓ reporting.report\_type.updated 5:02:05 AM
- ✓ reporting.report\_type.updated 5:02:04 AM

reporting.report\_type.updated

Response

✓ 200 OK

1 Unhandled webhook received: reporting.report\_type.updated

Request

```
1 {  
2   "id": "evt_1Jm1Q7GI8FO5pWvHf0TmQpgE",  
3   "type": "reporting.report_type.updated",  
4   "time": 1597501000000,  
5   "livemode": false,  
6   "secret": "sk_test_4eC39F03F794F9AD2F549B72",  
7   "data": {  
8     "object": "report_type",  
9     "id": "rt_1Jm1Q7GI8FO5pWvHf0TmQpgE",  
10    "name": "Report Type 1",  
11    "description": "A report type for testing.",  
12    "status": "active",  
13    "created": 1597501000000,  
14    "updated": 1597501000000  
15  }  
}
```

# Send a test webhook from stripe to check that the listener is working

The screenshot shows the Stripe dashboard interface. The URL in the browser is [https://dashboard.stripe.com/test/webhooks/we\\_1Jm1Q7GI8FO5pWvHf0TmQpgE](https://dashboard.stripe.com/test/webhooks/we_1Jm1Q7GI8FO5pWvHf0TmQpgE). The left sidebar is titled "Developers" and includes links for "Overview", "API keys", "Webhooks" (which is selected), "Events", "Logs", and "Extensions". The main content area has a title "Send a test event to a webhook endpoint" and displays a JSON configuration for a test webhook. The JSON code is as follows:

```
14     "business_profile": {  
15         "mcc": null,  
16         "name": null,  
17         "product_description": null,  
18         "support_address": null,  
19         "support_email": null,  
20         "support_phone": null,  
21         "support_url": null,  
22         "url": null  
23     },  
24     "business_type": null,  
25     "capabilities": {
```

Below the configuration, there is a "Response" section with the message "Test webhook sent successfully". At the bottom, it says "Unhandled webhook received: account.updated". At the very bottom right are two buttons: "Cancel" and "Send test webhook".