

Module 6: Functions and Storage

<https://github.com/...>

• Screenshots showing the configuration of the OrderItemsReserver Azure Functions.

The screenshot displays the 'Overview' page for the 'ISorderItemsReserver' Function App in the Microsoft Azure portal. The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Microsoft Defender for Cloud, Events (preview), Functions, App keys, App files, Profiles, Deployment, Deployment slots, Deployment Center, Settings, and Configuration. The main content area shows the 'Essentials' section with details about the resource group, location, subscription, and tags. Below this, the 'Functions' tab is active, displaying a table with one function named 'reserveOrder'.

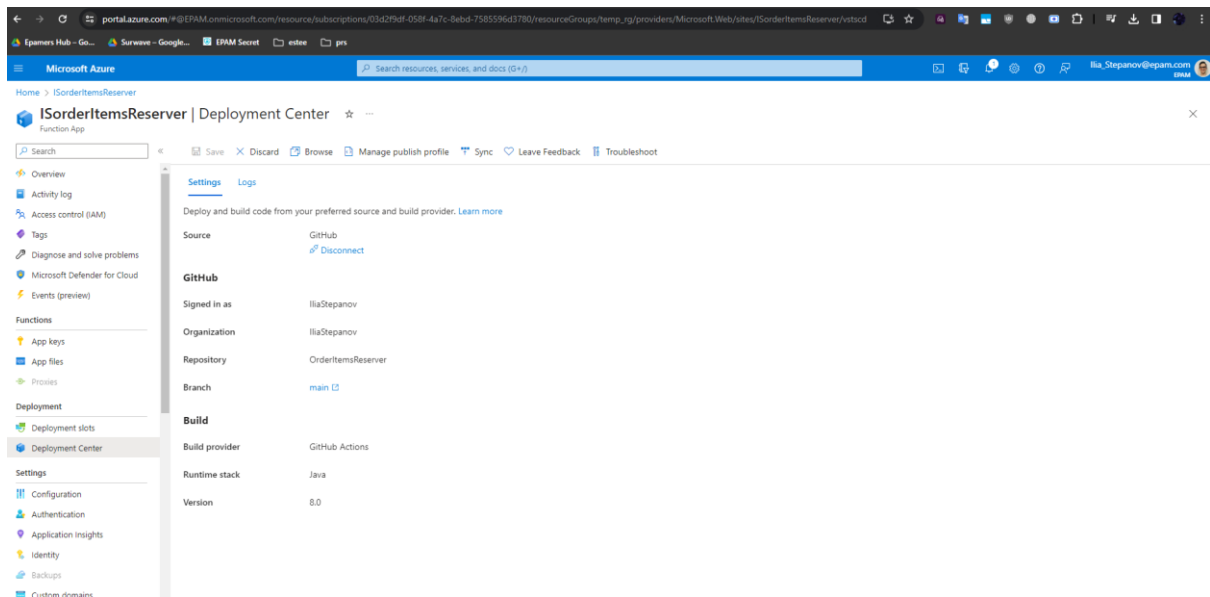
Name	Trigger	Status	Monitor
reserveOrder	HTTP	Enabled	Invocations and more

The screenshot displays the 'Configuration' page for the 'ISorderItemsReserver' Function App. The left sidebar is the same as the previous screenshot. The main content area shows the 'Application settings' tab, which includes a table of application settings and a section for connection strings.

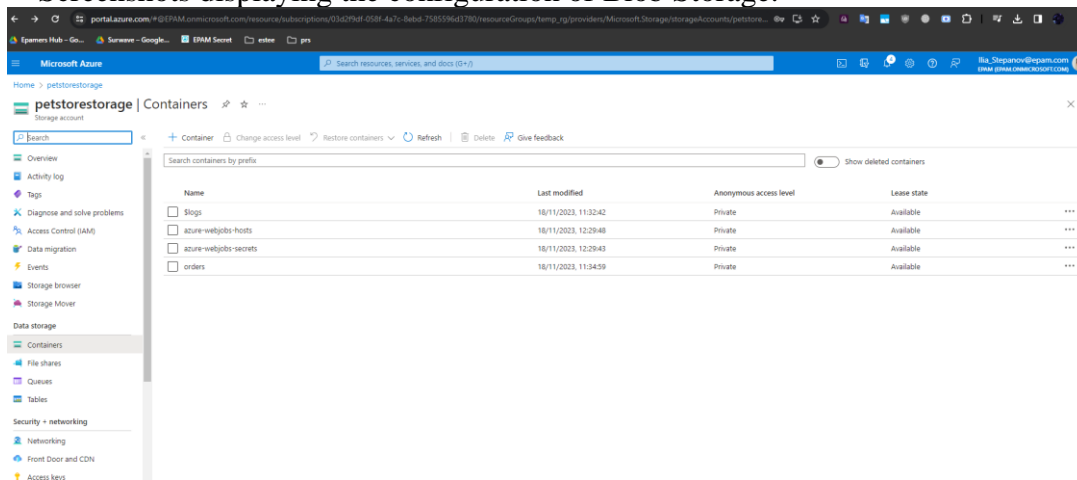
Name	Value	Source	Deployment slot setting	Delete	Edit
APPINSIGHTS_INSTRUMENTATIONKEY	Hidden value. Click to show value	App Service	✓		
APPLICATIONINSIGHTS_CONNECTION_STRING	Hidden value. Click to show value	App Service			
APPLICATIONINSIGHTS_ENABLE_AGENT	Hidden value. Click to show value	App Service			
AzureWebJobsStorage	Hidden value. Click to show value	App Service			
FUNCTIONS_EXTENSION_VERSION	Hidden value. Click to show value	App Service			
FUNCTIONS_WORKER_RUNTIME	Hidden value. Click to show value	App Service			
storageKey	Hidden value. Click to show value	App Service			
WEBSITE_CONTENTAZUREFILECONNECTIONSTRING	Hidden value. Click to show value	App Service			
WEBSITE_CONTENTSHARE	Hidden value. Click to show value	App Service			

Connection strings

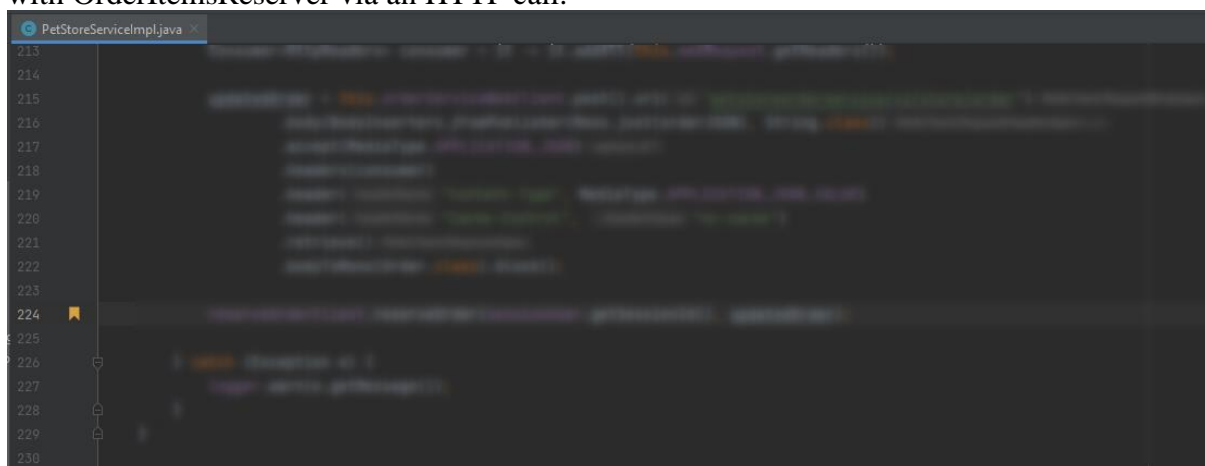
Connection strings are encrypted at rest and transmitted over an encrypted channel. Connection strings should only be used with a function app if you are using entity framework. For other scenarios use App Settings. [Learn more](#)



- Screenshots displaying the configuration of Blob Storage.



- Screenshots of the code within the PetStore application that establishes communication with OrderItemsReserver via an HTTP call.



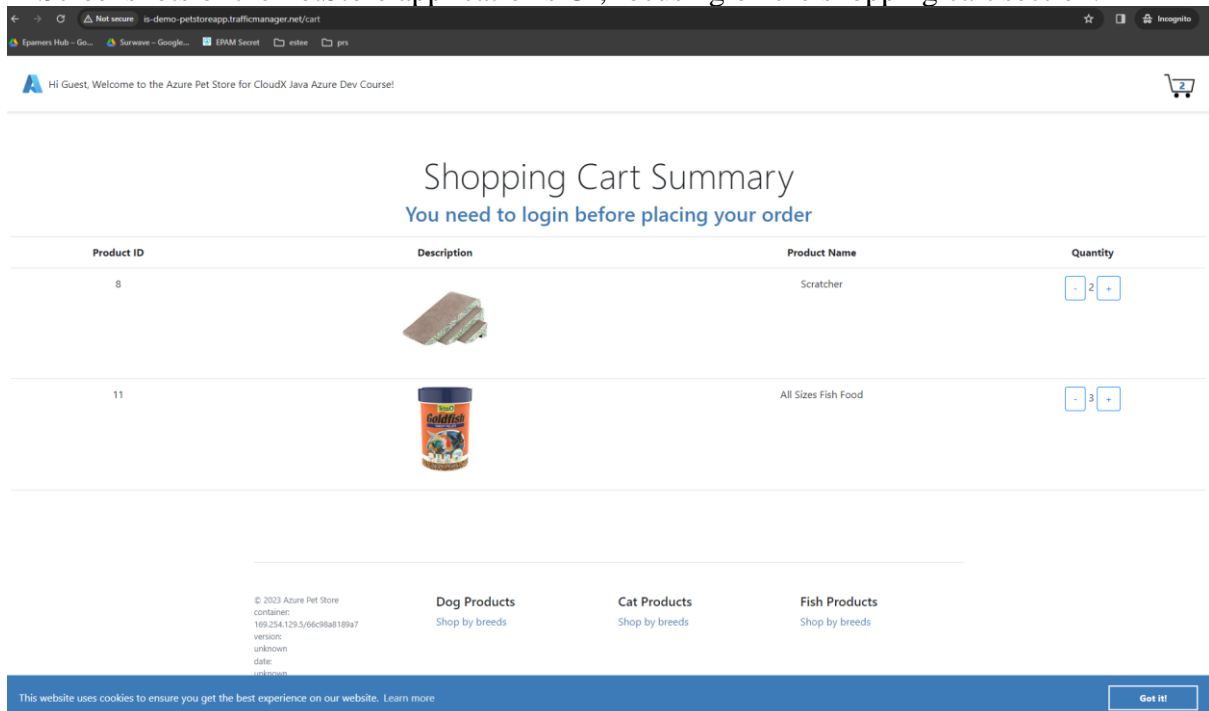
- Screenshots highlighting the code section responsible for uploading JSON files to Blob Storage.

```

1  import com.azure.functions.annotation.*;
2  import com.azure.storage.blob.*;
3  import com.azure.storage.blob.models.*;
4  import com.azure.storage.common.*;
5  import com.azure.core.util.BinaryData;
6  import java.io.IOException;
7  import java.io.InputStream;
8  import java.io.OutputStream;
9  import java.net.URI;
10 import java.net.URISyntaxException;
11
12 public class HttpTriggerFunction {
13     private BlobServiceClient blobServiceClient;
14     private String storageConnectionString;
15
16     public HttpTriggerFunction(BlobServiceClient blobServiceClient, String storageConnectionString) {
17         this.blobServiceClient = blobServiceClient;
18         this.storageConnectionString = storageConnectionString;
19     }
20
21     @FunctionName("HttpTrigger")
22     public HttpResponseMessage run(
23         @HttpTrigger(name = "req") HttpRequestMessage request,
24         boolean async,
25         Context ctx) {
26         try {
27             // Get the request body as a stream
28             InputStream inputStream = request.getBody().getInputStream();
29             // Read the stream into a byte array
30             byte[] requestBody = inputStream.readAllBytes();
31             // Convert the byte array to a string
32             String requestBodyString = new String(requestBody, "UTF-8");
33             // Parse the JSON body
34             JsonObject jsonObject = JsonParser.parseString(requestBodyString).getAsJsonObject();
35             // Get the file name from the JSON body
36             String fileName = jsonObject.get("fileName").getAsString();
37             // Get the file content from the JSON body
38             String fileContent = jsonObject.get("fileContent").getAsString();
39             // Upload the file to Blob Storage
40             uploadFile(fileName, fileContent);
41             // Return a 200 OK response
42             return request.createResponse(200);
43         } catch (IOException | URISyntaxException e) {
44             // Return a 500 Internal Server Error response
45             return request.createResponse(500);
46         }
47     }
48
49     private void uploadFile(String fileName, String fileContent) {
50         // Create a BlobClient
51         BlobClient blobClient = blobServiceClient.getBlobClient("my-container/" + fileName);
52         // Upload the file content to the blob
53         blobClient.upload(BinaryData.fromString(fileContent));
54     }
55 }

```

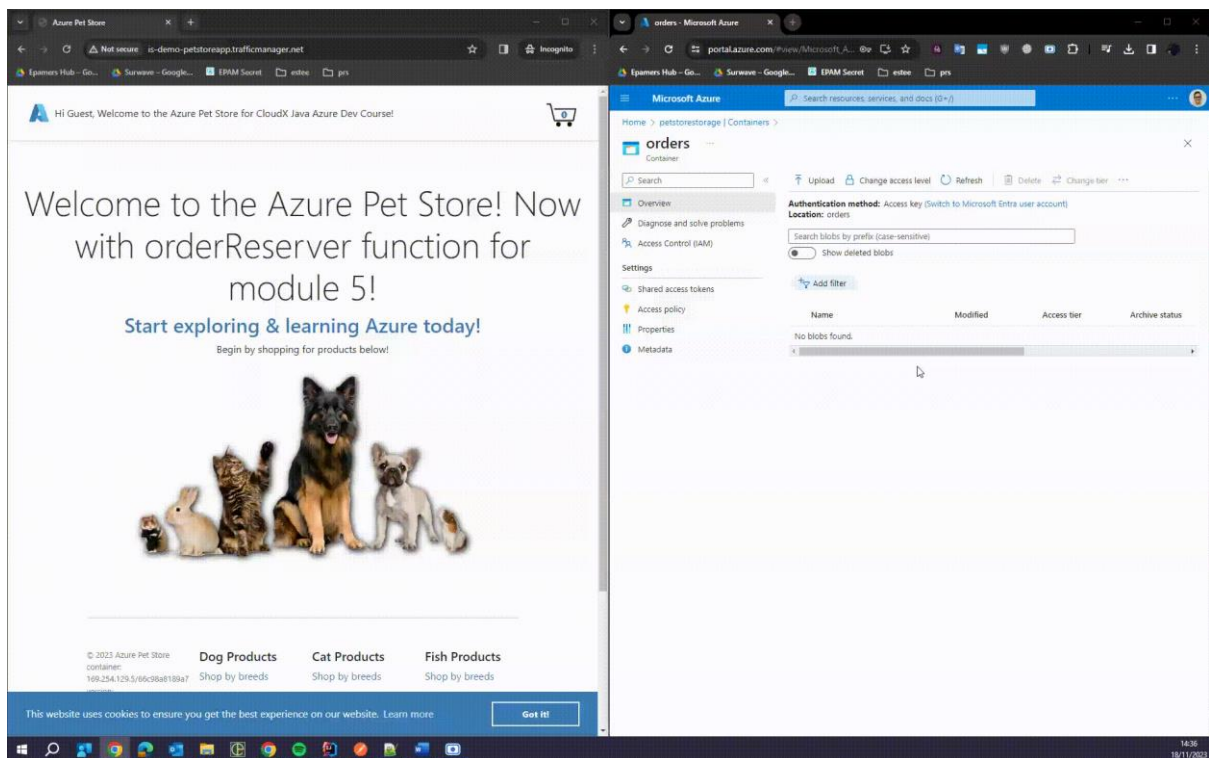
- Screenshots of the PetStore application's UI, focusing on the shopping cart section.



- Screenshots illustrating the content and structure of a generated JSON file in Blob Storage.

```
{
  "id": "E7DBA502A3B795953DDE76EA965EBBF3",
  "products": [
    {
      "id": 8,
      "name": "Scratcher",
      "photoURL": "https://raw.githubusercontent.com/chtrembl/staticcontent/master/cat-toys/scratcher.jpg?raw=true",
      "quantity": 10
    },
    {
      "id": 11,
      "name": "All Sizes Fish Food",
      "photoURL": "https://raw.githubusercontent.com/chtrembl/staticcontent/master/fish-food/fish.jpg?raw=true",
      "quantity": 3
    }
  ],
  "complete": false
}
```

- Screenshots confirming that the JSON file is overwritten with each user session update.



- A screenshot showing the list of Azure resources that correspond to the diagram.

The screenshot shows the Microsoft Azure portal interface for the resource group 'temp_rg'. The left sidebar contains navigation options like Overview, Activity log, Access control (IAM), Tags, Resource visualizer, Events, Settings, Deployments, Security, Deployment stacks, Policies, Properties, Locks, Cost Management, Monitoring, Insights (preview), Alerts, Metrics, Diagnostic settings, Logs, Advisor recommendations, and Workbooks.

The main content area displays the 'Essentials' section for the resource group, showing the subscription (EPAM), subscription ID, location (East US), and tags. Below this, the 'Resources' section lists 13 resources, showing 1 to 13 of 13 records. The resources are listed in a table with columns for Name, Type, and Location.

Name	Type	Location
ASP-temp-rg-1c1e	App Service plan	East US
asp_petstore_api_eastus	App Service plan	East US
asp_petstore_web_eastus	App Service plan	East US
asp_petstore_web_westeurope	App Service plan	West Europe
is-demo-petstoreapp	Traffic Manager profile	Global
is-petstoreapp-eastus	App Service	East US
is-petstoreapp-westeurope	App Service	West Europe
is-petstoreorderservice-eastus	App Service	East US
is-petstorepetservice-eastus	App Service	East US
is-petstoreproductservice-eastus	App Service	East US
isorderitemsReserver	Function App	East US
isorderitemsReserver	Application Insights	East US
petstorestorage	Storage account	East US

The page also includes a search bar, filters for Name, Type, and Location, and a 'Show hidden types' checkbox. The bottom of the page shows pagination controls (Page 1 of 1) and a 'Give feedback' link.