CELEBAL SUMMER INTERNSHIP – PROJECT

ACI CICD using Azure DevOps

Problem Statement - Automate Azure Container Instance deployment with Azure DevOps CI/CD pipelines for streamlined containerized application delivery.

Name – Ayush Pandey

Student ID - CT_CSI_DV_2504

Phase 1: Initial Setup

- the initial environment setup such as Azure resources, resource groups, and container registry.

```
C:\Windows\System32>az group create --name rg-finance-app --location eastus
{
  "id": "/subscriptions/4af928af-144b-44ea-a5ad-9fe8d622bd21/resourceGroups/rg-finance-app",
  "location": "eastus",
  "managedBy": null,
  "name": "rg-finance-app",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
```

```
C:\Windows\System32>az acr create --resource-group rg-finance-app --name financeappacr --sku Basic --admin-enabled true
{
  "adminUserEnabled": true,
  "anonymousPullEnabled": false,
  "autoGeneratedDomainNameLabelScope": "Unsecure",
  "creationDate": "2025-07-09T09:57:42.111484+00:00",
  "dataEndpointEnabled": false,
  "dataEndpointHostNames": [],
  "encryption": {
    "keyVaultProperties": null,
    "status": "disabled"
  },
  "id": "/subscriptions/4af928af-144b-44ea-a5ad-9fe8d622bd21/resourceGroups/rg-finance-app/providers/Microsoft.ContainerRegistry/registries/financeappacr",
  "identity": null,
  "location": "eastus",
  "loginServer": "financeappacr.azurecr.io",
  "metadataSearch": "Disabled",
  "name": "financeappacr",
  "networkRuleBypassOptions": "AzureServices",
  "networkRuleSet": null,
  "policies": {
    "azureAdAuthenticationAsArmPolicy": {
      "status": "enabled"
    },
    "exportPolicy": {
      "status": "enabled"
    },
    "quarantinePolicy": {
      "status": "disabled"
    },
```

```json
      "lastUpdatedTime": "2025-07-09T09:57:54.222499+00:00",
      "retentionDays": 7,
      "status": "disabled"
    },
    "trustPolicy": {
      "status": "disabled",
      "type": "Notary"
    }
  },
  "privateEndpointConnections": [],
  "provisioningState": "Succeeded",
  "publicNetworkAccess": "Enabled",
  "resourceGroup": "rg-finance-app",
  "roleAssignmentMode": "LegacyRegistryPermissions",
  "sku": {
    "name": "Basic",
    "tier": "Basic"
  },
  "status": null,
  "systemData": {
    "createdAt": "2025-07-09T09:57:42.111484+00:00",
    "createdBy": "22BDO10038@cuchd.in",
    "createdByType": "User",
    "lastModifiedAt": "2025-07-09T09:57:42.111484+00:00",
    "lastModifiedBy": "22BDO10038@cuchd.in",
    "lastModifiedByType": "User"
  },
  "tags": {},
  "type": "Microsoft.ContainerRegistry/registries",
  "zoneRedundancy": "Disabled"
}
```

```
C:\Windows\System32>az acr show --name financeappacr --query loginServer --output table
Result
----------------------
financeappacr.azurecr.io

C:\Windows\System32>az acr credential show --name financeappacr
{
  "passwords": [
    {
      "name": "password",
      "value": "ibXzSxkaDGP8vlosf98I8cMcZSELq25LzzAyq13yHM+ACRCbzvJu"
    },
    {
      "name": "password2",
      "value": "6OLHPv3Of5YILeqU/XZ2Tp3oYjnGYTdLUa4W+yXUf1+ACRBq3BtS"
    }
  ],
  "username": "financeappacr"
}
```

```
C:\Windows\System32>az ad sp create-for-rbac --name "finance-app-sp" --role contributor --scope /subscriptions/4af928af-144b-4
4ea-a5ad-9fe8d622bd21/resourceGroups/rg-finance-app --sdk-auth
Option '--sdk-auth' has been deprecated and will be removed in a future release.
Found an existing application instance: (id) e5d35dfb-990e-4283-ad9e-12fcbe0aae10. We will patch it.
Creating 'contributor' role assignment under scope '/subscriptions/4af928af-144b-44ea-a5ad-9fe8d622bd21/resourceGroups/rg-fina
nce-app'
The output includes credentials that you must protect. Be sure that you do not include these credentials in your code or check
 the credentials into your source control. For more information, see https://aka.ms/azadsp-cli
{
  "clientId": "d86d2dcb-b423-4b84-8fb4-a2ffc5bda36a",
  "clientSecret": "kXU8Q~qFYPQEhmTUbUNgglihpui5anyaLezCHa8W",
  "subscriptionId": "4af928af-144b-44ea-a5ad-9fe8d622bd21",
  "tenantId": "34bd8bed-2ac1-41ae-9f08-4e0a3f11706c",
  "activeDirectoryEndpointUrl": "https://login.microsoftonline.com",
  "resourceManagerEndpointUrl": "https://management.azure.com/",
  "activeDirectoryGraphResourceId": "https://graph.windows.net/",
  "sqlManagementEndpointUrl": "https://management.core.windows.net:8443/",
  "galleryEndpointUrl": "https://gallery.azure.com/",
  "managementEndpointUrl": "https://management.core.windows.net/"
}

C:\Windows\System32>
```

Phase 2: Application Development

- Describes the development of the containerized application including code, Dockerfile, and local testing.
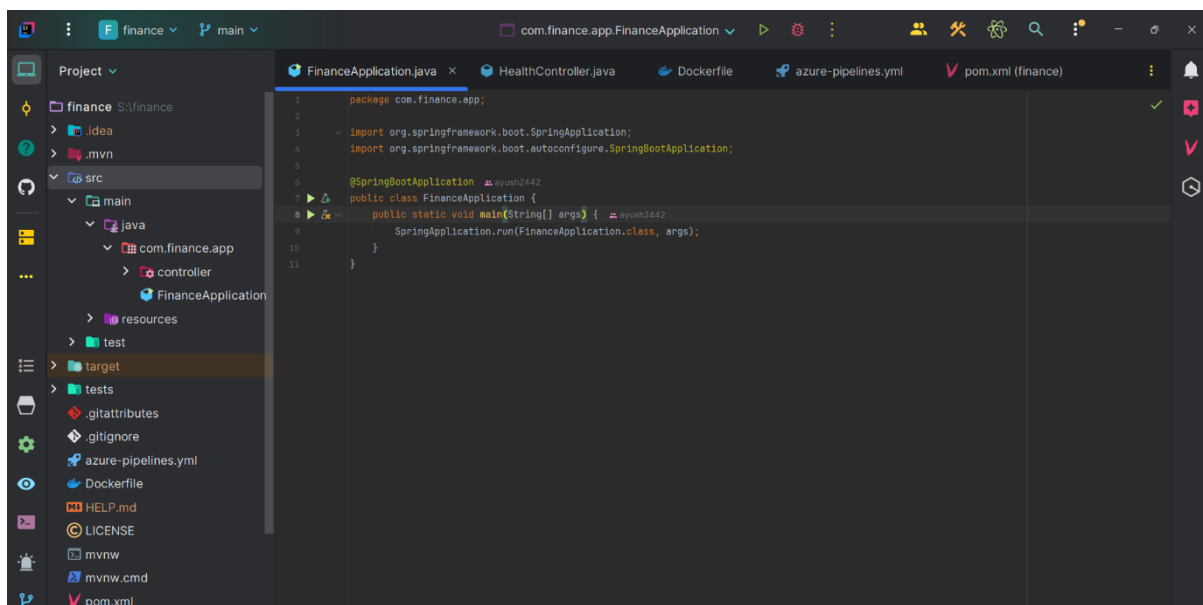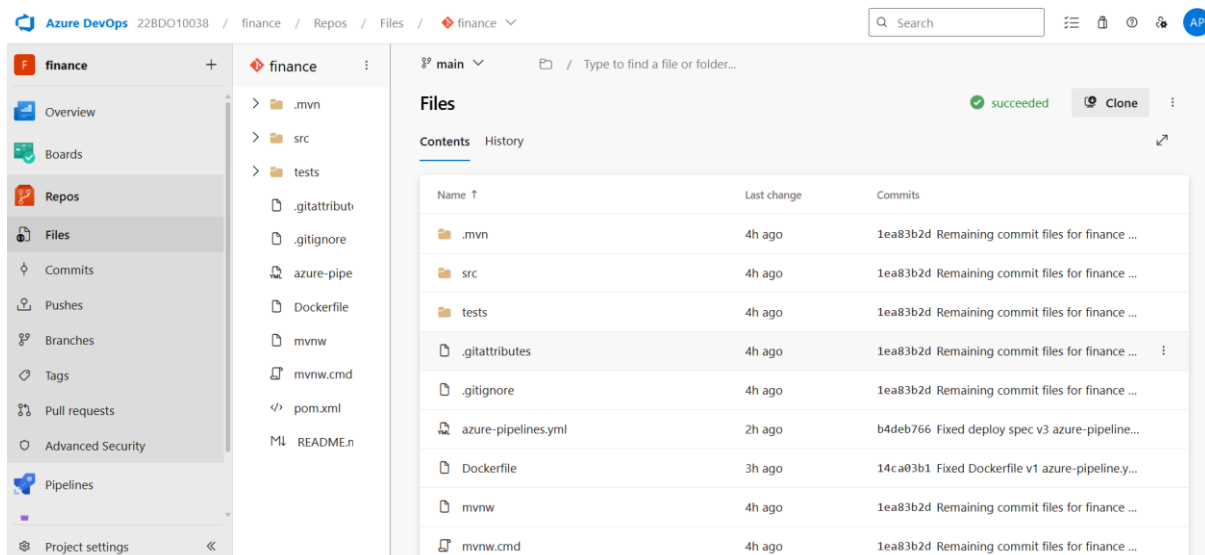
Phase 3: CI/CD Pipeline

- Focuses on the CI/CD pipeline setup in Azure DevOps, including pipeline YAML and build/deploy steps.

FinanceApplication.java | HealthController.java | Dockerfile | azure-pipelines.yml × | pom.xml (finance)

```yaml
trigger:
  branches:
    include:
      - main
      - develop
  paths:
    exclude:
      - README.md
      - docs/*

variables:
  acrServiceConnection: 'financeappacr-connection'
  containerRegistryUrl: 'financeappacr.azurecr.io'
  imageRepository: 'finance-app'
  dockerfilePath: '$(Build.SourcesDirectory)/Dockerfile'
  tag: '$(Build.BuildId)'

  azureSubscription: 'azure-finance-connection'
  resourceGroup: 'rg-finance-app'
  containerInstanceName: 'finance-app-aci'

  vmImageName: 'ubuntu-latest'

stages:
  - stage: Build
    displayName: Build and Push Image
    jobs:
      - job: Build
        displayName: Build
        pool:
          vmImage: $(vmImageName)
        steps:
```
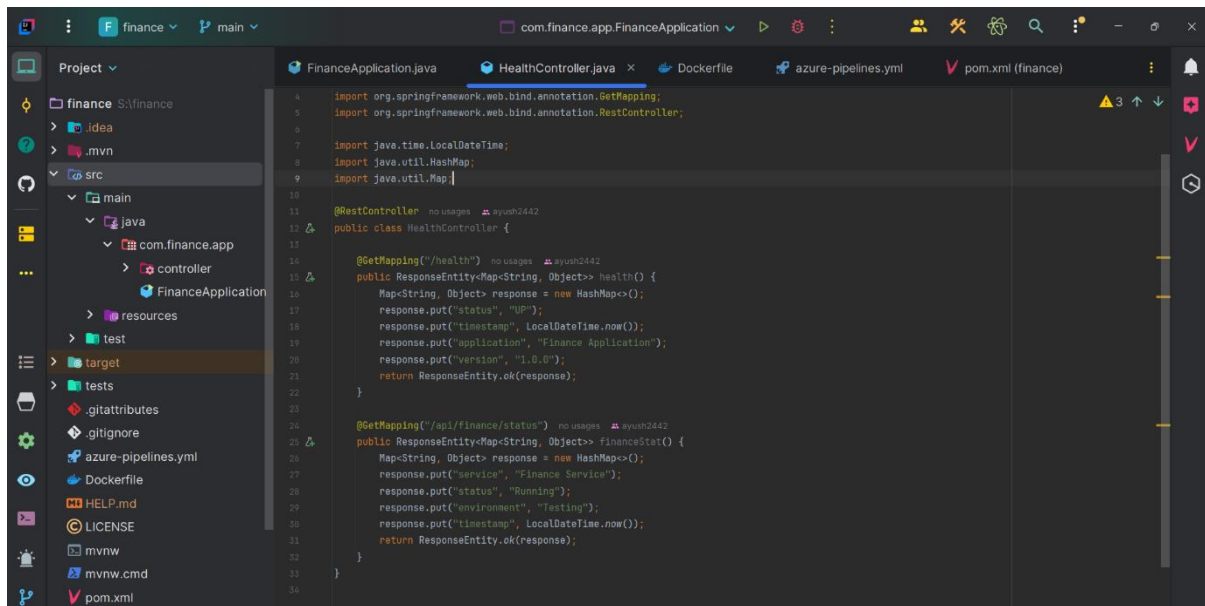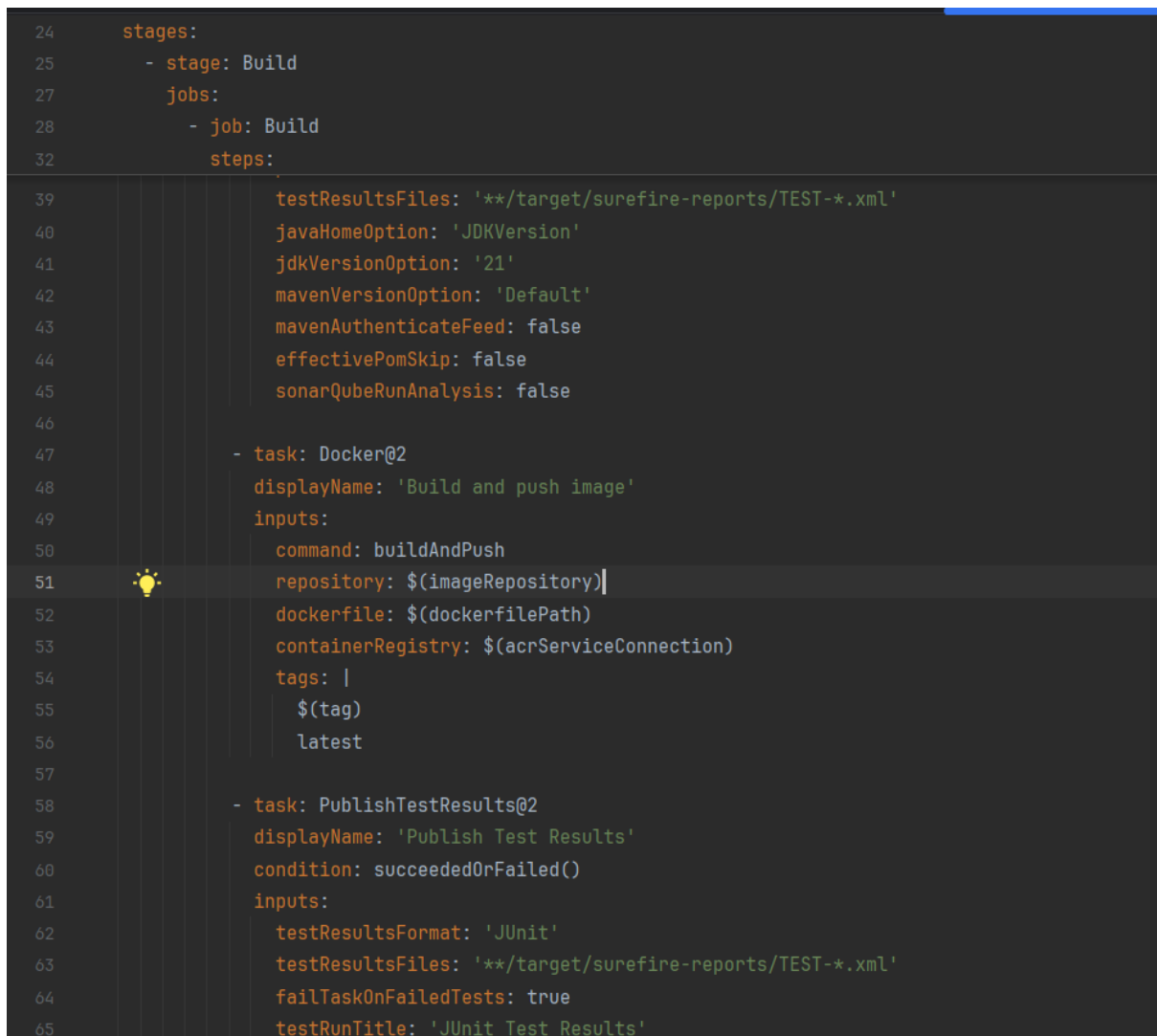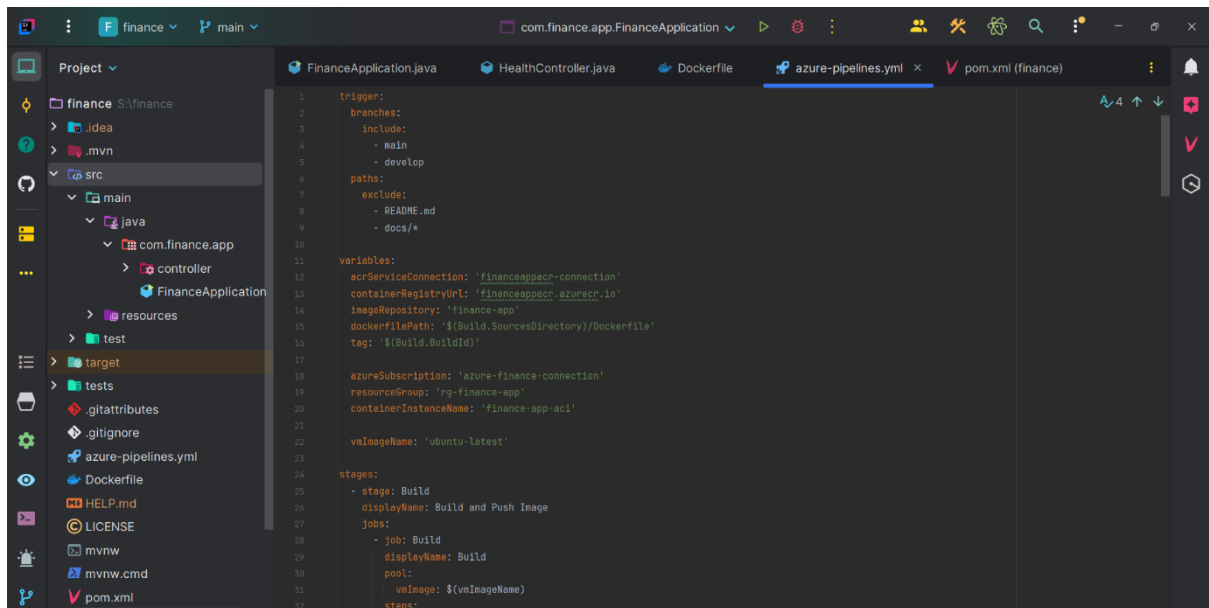
```yaml
stages:
  - stage: Build
    jobs:
      - job: Build
        steps:

            testResultsFiles: '**/target/surefire-reports/TEST-*.xml'
            javaHomeOption: 'JDKVersion'
            jdkVersionOption: '21'
            mavenVersionOption: 'Default'
            mavenAuthenticateFeed: false
            effectivePomSkip: false
            sonarQubeRunAnalysis: false

        - task: Docker@2
          displayName: 'Build and push image'
          inputs:
            command: buildAndPush
            repository: $(imageRepository)
            dockerfile: $(dockerfilePath)
            containerRegistry: $(acrServiceConnection)
            tags: |
              $(tag)
              latest

        - task: PublishTestResults@2
          displayName: 'Publish Test Results'
          condition: succeededOrFailed()
          inputs:
            testResultsFormat: 'JUnit'
            testResultsFiles: '**/target/surefire-reports/TEST-*.xml'
            failTaskOnFailedTests: true
            testRunTitle: 'JUnit Test Results'
```

```yaml
stages:
  - stage: Deploy
    jobs:
      - deployment: Deploy
        strategy:
```

```
                --name $(containerInstanceName) \
                --image $(containerRegistryUrl)/$(imageRepository):$(tag) \
                --os-type Linux \
                --dns-name-label finance-app-$(Build.BuildId) \
                --ports 8080 \
                --cpu 1 \
                --memory 2 \
                --registry-login-server $(containerRegistryUrl) \
                --registry-username $(acrUsername) \
                --registry-password $(acrPassword) \
                --environment-variables \
                  SPRING_PROFILES_ACTIVE=testing \
                  SERVER_PORT=8080 \
                --restart-policy OnFailure


                echo "Waiting for container to be ready..."
                sleep 30

                az container show \
                  --resource-group $(resourceGroup) \
                  --name $(containerInstanceName) \
                  --query "{FQDN:ipAddress.fqdn,ProvisioningState:provisioningState}" \
                  --output table
            env:
              acrUsername: $(acrUsername)
              acrPassword: $(acrPassword)
```


Azure DevOps 22BDO10038 / finance / Pipelines

finance   +

← finance

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Library

Test Plans

Artifacts

Project settings «

main ⌄   ◆ finance / azure-pipelines.yml

```
 1  trigger:
 2    branches:
 3      include:
 4        - main
 5        - develop
 6    paths:
 7      exclude:
 8        - README.md
 9        - docs/*
10
11  variables:
12    acrServiceConnection: 'financeappacr-connection'
13    containerRegistryUrl: 'financeappacr.azurecr.io'
14    imageRepository: 'finance-app'
15    dockerfilePath: '$(Build.SourcesDirectory)/Dockerfile'
16    tag: '$(Build.BuildId)'
17
18    azureSubscription: 'azure-finance-connection'
19    resourceGroup: 'rg-finance-app'
20    containerInstanceName: 'finance-app-aci'
21
22    vmImageName: 'ubuntu-latest'
23
```

Variables   ✕

🔍 Search variables   +

𝑓x **acrPassword**
    = 6OLHPv3Of5YILeqU/XZ2Tp3oYjnGYTdLUa4W+yXUf1+AC...

𝑓x **acrUsername**
    = financeappacr

𝑓x **containerRegistry**
    = financeappacr.azurecr.io

Learn about variables   Close   Save

Phase 4: Testing and Monitoring

- Details the testing procedures and implementation of monitoring tools like Azure Monitor or Application Insights.

```yaml
stages:
  - stage: Test
    jobs:
      - job: IntegrationTests
        steps:
          - task: NodeTool@0
            displayName: 'Install Node.js'
            inputs:
              versionSpec: '18.x'

          - task: AzureCLI@2
            displayName: 'Run Integration Tests'
            inputs:
              azureSubscription: $(azureSubscription)
              scriptType: 'bash'
              scriptLocation: 'inlineScript'
              inlineScript: |
                FQDN=$(az container show \
                  --resource-group $(resourceGroup) \
                  --name $(containerInstanceName) \
                  --query "ipAddress.fqdn" \
                  --output tsv)

                echo "Testing application at: http://$FQDN:8080"

                npm install -g newman

                curl -f http://$FQDN:8080/health || exit 1

                curl -f http://$FQDN:8080/api/finance/status || exit 1

                echo "All tests passed successfully!"
```

```yaml
stages:
  - stage: Monitor
    displayName: Setup Monitoring
    dependsOn: Test
    condition: succeeded()
    jobs:
      - job: SetupMonitoring
        displayName: 'Setup Application Monitoring'
        pool:
          vmImage: $(vmImageName)
        steps:
          - task: AzureCLI@2
            displayName: 'Configure Application Insights'
            inputs:
              azureSubscription: $(azureSubscription)
              scriptType: 'bash'
              scriptLocation: 'inlineScript'
              inlineScript: |
                az monitor app-insights component create \
                  --app finance-app-insights \
                  --location eastus \
                  --resource-group $(resourceGroup) \
                  --kind web \
                  --application-type web

                INSTRUMENTATION_KEY=$(az monitor app-insights component show \
                  --app finance-app-insights \
                  --resource-group $(resourceGroup) \
                  --query "instrumentationKey" \
                  --output tsv)

                echo "Application Insights configured with key: $INSTRUMENTATION_KEY"
```

```
C:\Windows\System32>az monitor app-insights component create --app finance-app-insights --location eastus --resource-group rg-finance-app --kind w
eb --application-type web
Preview version of extension is disabled by default for extension installation, enabled for modules without stable versions.
Please run 'az config set extension.dynamic_install_allow_preview=true or false' to config it specifically.
The command requires the extension application-insights. Do you want to install it now? The command will continue to run after the extension is in
stalled. (Y/n): y
Run 'az config set extension.use_dynamic_install=yes_without_prompt' to allow installing extensions without prompt.
Extension 'application-insights' has a later preview version to install, add `--allow-preview True` to try preview version.
{- Installing ..
  "appId": "cd376eb6-46d0-4077-a3c3-448f38dae1f1",
  "applicationId": "finance-app-insights",
  "applicationType": "web",
  "connectionString": "InstrumentationKey=d705e409-39a6-4377-acdb-1e86aa1d7813;IngestionEndpoint=https://eastus-8.in.applicationinsights.azure.com
/;LiveEndpoint=https://eastus.livediagnostics.monitor.azure.com/;ApplicationId=cd376eb6-46d0-4077-a3c3-448f38dae1f1",
  "creationDate": "2025-07-10T09:40:47.855150+00:00",
  "disableIpMasking": null,
  "etag": "\"8c003aac-0000-0100-0000-686f8aaa0000\"",
  "flowType": "Bluefield",
  "hockeyAppId": null,
  "hockeyAppToken": null,
  "id": "/subscriptions/4af928af-144b-44ea-a5ad-9fe8d622bd21/resourceGroups/rg-finance-app/providers/microsoft.insights/components/finance-app-ins
ights",
  "immediatePurgeDataOn30Days": null,
  "ingestionMode": "LogAnalytics",
  "instrumentationKey": "d705e409-39a6-4377-acdb-1e86aa1d7813",
  "kind": "web",
  "location": "eastus",
  "name": "finance-app-insights",
  "privateLinkScopedResources": null,
  "provisioningState": "Succeeded",
  "publicNetworkAccessForIngestion": "Enabled",
  "publicNetworkAccessForQuery": "Enabled",
  "requestSource": "rest",
  "resourceGroup": "rg-finance-app",
  "retentionInDays": 90,
```

```
C:\Windows\System32>az container show --name finance-app-aci --resource-group rg-finance-app
{
  "confidentialComputeProperties": null,
  "containerGroupProfile": null,
  "containers": [
    {
      "command": null,
      "configMap": {
        "keyValuePairs": {}
      },
      "environmentVariables": [
        {
          "name": "SPRING_PROFILES_ACTIVE",
          "secureValue": null,
          "value": "testing"
        },
        {
          "name": "SERVER_PORT",
          "secureValue": null,
          "value": "8080"
        }
      ],
      "image": "financeappacr.azurecr.io/finance-app:48",
      "instanceView": {
        "currentState": {
          "detailStatus": "",
          "exitCode": null,
          "finishTime": null,
          "startTime": "2025-07-10T14:06:44.023000+00:00",
          "state": "Running"
        },
        "events": [
          {
            "count": 1,
            "firstTimestamp": "2025-07-10T14:06:21+00:00",
```

```json
  ],
  "initContainers": [],
  "instanceView": {
    "events": [],
    "state": "Running"
  },
  "ipAddress": {
    "autoGeneratedDomainNameLabelScope": "Unsecure",
    "dnsNameLabel": "finance-app-48",
    "fqdn": "finance-app-48.eastus.azurecontainer.io",
    "ip": "4.236.219.94",
    "ports": [
      {
        "port": 8080,
        "protocol": "TCP"
      }
    ],
    "type": "Public"
  },
  "isCreatedFromStandbyPool": false,
  "location": "eastus",
  "name": "finance-app-aci",
  "osType": "Linux",
  "priority": null,
  "provisioningState": "Succeeded",
  "resourceGroup": "rg-finance-app",
  "restartPolicy": "OnFailure",
  "sku": "Standard",
  "standbyPoolProfile": null,
  "subnetIds": null,
  "tags": {},
  "type": "Microsoft.ContainerInstance/containerGroups",
  "volumes": null,
  "zones": null
}
```

RESULT

```
C:\Windows\System32>az container logs --name finance-app-aci --resource-group rg-finance-app


   .   ____          _            __ _ _
  /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
 ( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
  \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
   '  |____| .__|_| |_|_| |_\__, | / / / /
  =========|_|==============|___/=/_/_/_/

 :: Spring Boot ::                (v3.5.3)

2025-07-10T14:06:50.280Z  INFO 19 --- [finance] [           main] com.finance.app.FinanceApplication      : Starting FinanceApplication v0.0.1-SN
APSHOT using Java 21.0.7 with PID 19 (/app/app.jar started by financeapp in /app)
2025-07-10T14:06:50.295Z  INFO 19 --- [finance] [           main] com.finance.app.FinanceApplication      : The following 1 profile is active: "t
esting"
2025-07-10T14:06:58.863Z  INFO 19 --- [finance] [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 8080 (ht
tp)
2025-07-10T14:06:58.984Z  INFO 19 --- [finance] [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2025-07-10T14:06:58.986Z  INFO 19 --- [finance] [           main] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache Tomc
at/10.1.42]
2025-07-10T14:06:59.701Z  INFO 19 --- [finance] [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebAppli
cationContext
2025-07-10T14:06:59.716Z  INFO 19 --- [finance] [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initializ
ation completed in 9114 ms
2025-07-10T14:07:05.094Z  INFO 19 --- [finance] [           main] o.s.b.a.e.web.EndpointLinksResolver      : Exposing 1 endpoint beneath base path
 '/actuator'
2025-07-10T14:07:05.570Z  INFO 19 --- [finance] [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port 8080 (http) wi
th context path '/'
2025-07-10T14:07:05.722Z  INFO 19 --- [finance] [           main] com.finance.app.FinanceApplication      : Started FinanceApplication in 18.298
seconds (process running for 21.592)
2025-07-10T14:08:43.838Z  INFO 19 --- [finance] [nio-8080-exec-4] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring DispatcherServlet
 'dispatcherServlet'
2025-07-10T14:08:43.848Z  INFO 19 --- [finance] [nio-8080-exec-4] o.s.web.servlet.DispatcherServlet       : Initializing Servlet 'dispatcherServl
et'
2025-07-10T14:08:43.851Z  INFO 19 --- [finance] [nio-8080-exec-4] o.s.web.servlet.DispatcherServlet       : Completed initialization in 1 ms
```