# Deployment Documentation

## 1. Overview

This document explains how to deploy code changes using **GitHub Workflows**.  
The process involves:

1. Pushing your branch to the remote repository
2. Identifying the commit SHA to deploy
3. Using GitHub Workflow to deploy the changes to the desired environment

## 2. Prerequisites

Before deploying, ensure you have:

* Access to the repository with required permissions
* The GitHub Actions workflows already set up in .github/workflows/
* Correct environment configuration (e.g., staging, production)
* Your branch pushed to GitHub

## 3. Step-by-Step Deployment Guide

### Get the Commit SHA

Go to your branch in GitHub and find the commit you want to deploy.  
Copy the **full commit SHA** from the commit history.

**Screenshot Placeholder:**

[Insert screenshot here: GitHub commit history page]

Description: Shows where to find and copy the commit SHA from GitHub.

### Trigger GitHub Workflow

1. Navigate to the **"Actions"** tab in your GitHub repository.
2. Select the deployment workflow (e.g., Deploy to Environment).
3. Click **"Run workflow"**.
4. Provide the required inputs:
   * **Commit SHA** — Paste the SHA you copied earlier
   * **Environment** — Select from available options (e.g., staging, production)

**Screenshot Placeholder:**

[Insert screenshot here: GitHub workflow trigger screen]

Description: Shows where to enter commit SHA and environment in the GitHub Actions "Run workflow" panel.

### Monitor the Deployment

Once the workflow starts:

* Monitor progress in the **Actions** tab
* Check for build, test, and deployment steps
* Wait for the status to change to **Success** or **Failure**

**Screenshot Placeholder:**

[Insert screenshot here: GitHub Actions run progress screen]

Description: Shows the deployment job logs in GitHub Actions.

### Post Deployment Verification

After a successful run:

* Test the deployed environment to ensure changes are live and functional
* If deployment fails, review logs, fix the issue, and retry

## 4. Notes & Best Practices

* Always deploy from a clean, tested branch
* Never deploy unreviewed code to production
* Verify commit SHA before triggering workflow
* Use staging environments before production deployments