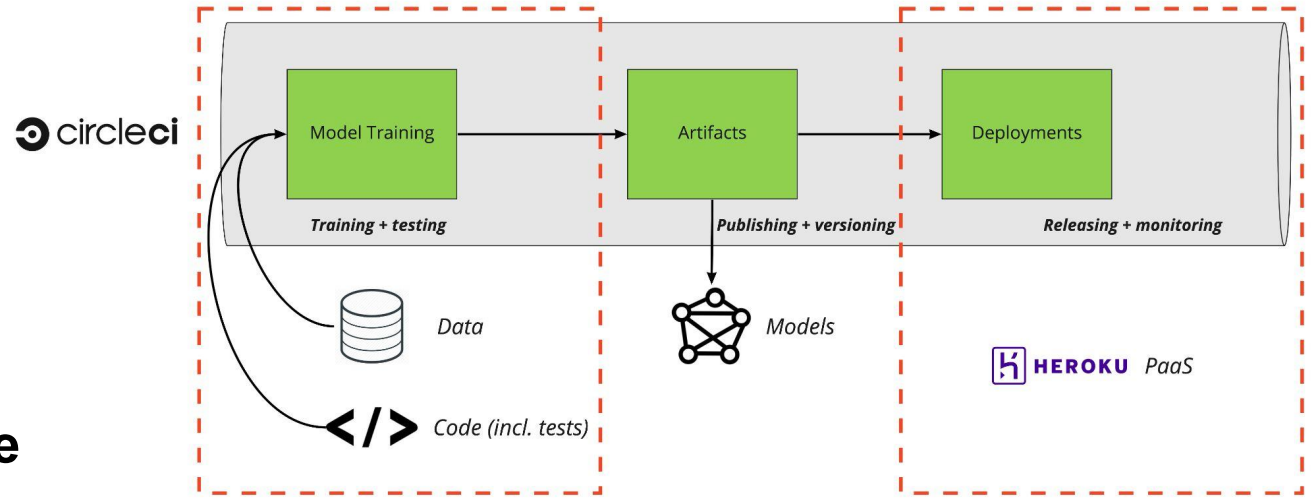


# Understanding CI/CD Automation 1

## Step 1: Automate the training, testing and deployment



*Before (section 5)*

```
tox -e train
```

```
tox
```

*Before (section 6)*

```
git push heroku main
```

# CircleCI Environment Variables Required

HEROKU\_API\_KEY  
HEROKU\_APP\_NAME  
PIP\_EXTRA\_INDEX\_URL (explained later in the section)

The screenshot shows the CircleCI Project Settings page for a project named 'deploying-machine-learning-models'. The browser address bar shows the URL: `app.circleci.com/settings/project/github/ChristopherGSTest/deploying-machine-learning-mod...`. The page has a sidebar on the left with navigation links: Overview, Advanced, Environment Variables (highlighted), SSH Keys, API Permissions, and Jira Integration. The main content area is titled 'Environment Variables' and contains the following text: 'Environment variables let you add sensitive data (e.g. API keys) to your jobs rather than placing them in the repository. The value of the variables cannot be read or edited in the app once they are set.' Below this text is a note: 'If you're looking to share environment variables across projects, try [Contexts](#).' At the bottom of the main content area, there is a table with two columns: 'Name' and 'Value'. To the right of the table are two buttons: 'Add Environment Variable' (in blue) and 'Import Variables' (in grey).

Project Settings  
deploying-machine-learning-models

Organization Settings

Overview  
Advanced  
**Environment Variables**  
SSH Keys  
API Permissions  
Jira Integration

## Environment Variables

Environment variables let you add sensitive data (e.g. API keys) to your jobs rather than placing them in the repository. The value of the variables cannot be read or edited in the app once they are set.

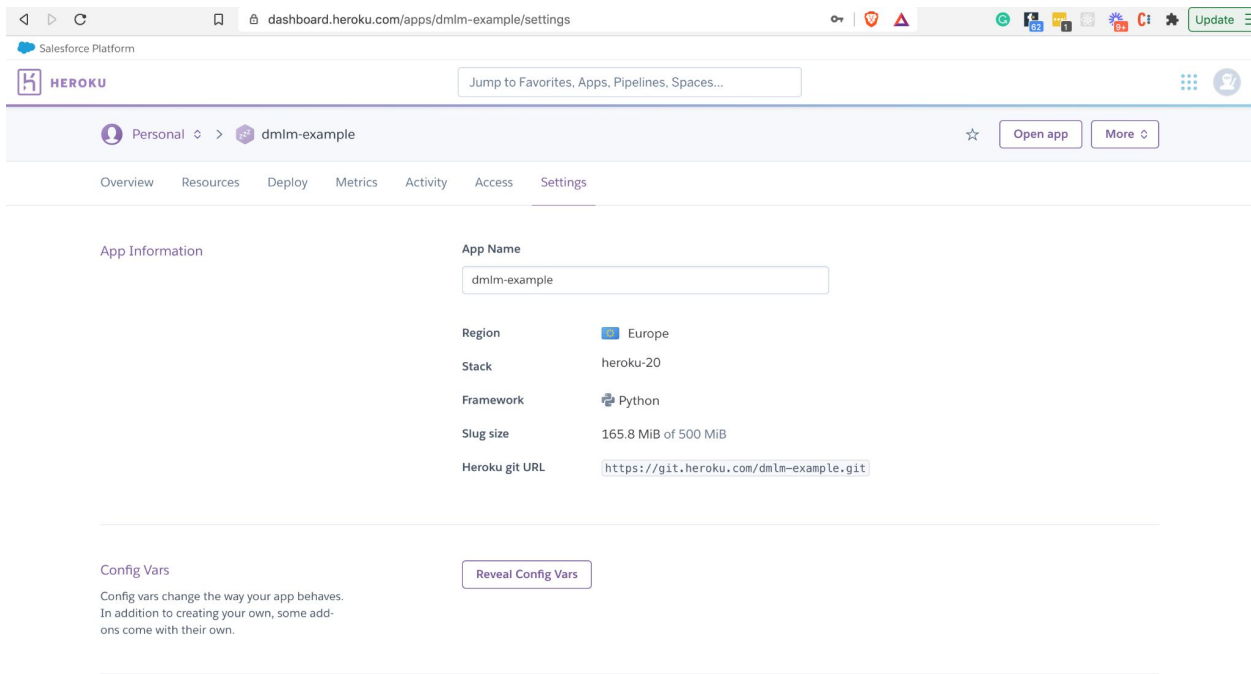
If you're looking to share environment variables across projects, try [Contexts](#).

Name	Value
------	-------

Add Environment Variable Import Variables

# Heroku Environment Variables Required

PIP\_EXTRA\_INDEX\_URL (explained later in the section)



The screenshot shows the Heroku dashboard for the 'dmlm-example' app. The browser address bar indicates the URL is `dashboard.heroku.com/apps/dmlm-example/settings`. The Heroku logo and a search bar are at the top. The app name 'dmlm-example' is displayed with 'Open app' and 'More' buttons. The 'Settings' tab is selected in the navigation menu. Under 'App Information', the following details are shown:

App Name	
dmlm-example	
Region	Europe
Stack	heroku-20
Framework	Python
Slug size	165.8 MiB of 500 MiB
Heroku git URL	<code>https://git.heroku.com/dmlm-example.git</code>

Below the app information, the 'Config Vars' section is visible, featuring a 'Reveal Config Vars' button. A note states: 'Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.'