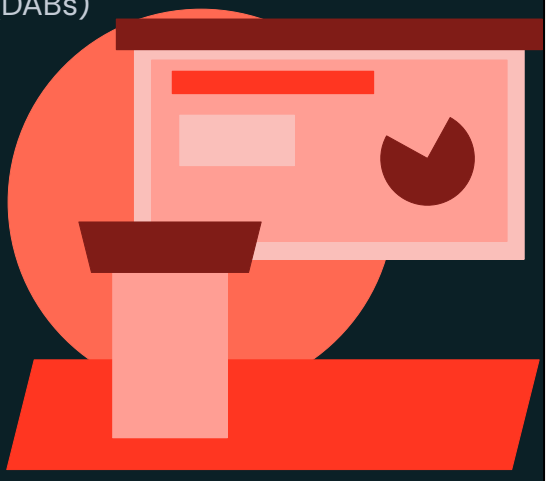
databricks

Deployment with Databricks Asset Bundles (DABs)

LECTURE

DAB Project Templates Overview



© Databricks 2025. All rights reserved. Apache, Apache Spark, Spark, the Spark Logo, Apache Iceberg, Iceberg, and the Apache Iceberg logo are trademarks of the Apache Software Foundation.

In this lecture, we provide an overview of Databricks Asset Bundle project templates.

Databricks Asset Bundle Project Templates

Overview

Default Bundle Template

Use a Databricks **default** bundle template to create your bundle

- Templates:
  - **default-python**
  - default-sql
  - dbt-sql
  - Mlops-stacks

```
databricks bundle init default-python
```

Custom Bundle Template

© Databricks 2025. All rights reserved. Apache, Apache Spark, Spark, the Spark Logo, Apache Iceberg, Iceberg, and the Apache Iceberg logo are trademarks of the Apache Software Foundation.

Now, let's take a quick look at Databricks Asset Bundle templates.
You have two main options here: you can either use the pre-configured templates provided by Databricks, or you can create your own custom templates to meet your specific needs.
Databricks offers several default bundle templates, including:

- default-python for Python-based projects
- default-sql for SQL-based projects
- dbt-sql for projects involving DBT (Data Build Tool) workflows
- mlops-stacks for machine learning operations stacks

To get started with one of these templates, you can use the Databricks CLI command databricks bundle init <template_name>. This command will automatically initialize your asset bundle based on the selected template, saving you time in setting up the configuration from scratch.
One note, when using a bundle template within a Databricks notebooks, you get the entire template. If you are running the CLI command outside of a notebook, a series of prompts enable you to select specific portions of the template.

Databricks Asset Bundle Project Templates

Overview

Default Bundle Template

Use a Databricks default bundle template to create your bundle

- Templates:
  - **default-python**
  - default-sql
  - dbt-sql
  - Mlops-stacks

Custom Bundle Template

- At a minimum, it must have **databricks_template_schema.json** and **databricks.yml.tmpl**
- You can add user prompts, specific folder structure and more
- To use a custom bundle template, pass its local path or remote URL to the Databricks CLI bundle init command.

```
databricks bundle init
/projects/templates/test-template
```

© Databricks 2025. All rights reserved. Apache, Apache Spark, Spark, the Spark Logo, Apache Iceberg, Iceberg, and the Apache Iceberg logo are trademarks of the Apache Software Foundation.

In addition to Databricks' default templates, you can create custom bundle templates specific to your organization.

Although we won't go into the details of creating custom templates in this course, here are a few key points:

- Custom templates require at least databricks_template_schema.json and databricks.yml.tmpl.
- You can include user prompts, define folder structures, and customize settings.

To use a custom template, simply pass the local path or remote URL of the template to the Databricks CLI bundle init command.

Custom templates help organizations create and manage bundles in a consistent and repeatable way, establishing folder structures, tasks, and DevOps infrastructure-as-code (IaC) for development and deployment.

© Databricks 2025. All rights reserved. Apache, Apache Spark, Spark, the Spark Logo, Apache Iceberg, Iceberg, and the Apache Iceberg logo are trademarks of the Apache Software Foundation.

Thank you for completing this lesson and continuing your journey to develop your skills with us.