

# Tutorial Pipeline

## 1. Features

- Use **blocks** for building **pipelines**
- Easily copy and paste **blocks**
- Drag and drop elements for intuitive organization
- Switch between the results view and the generated YAML
- Reset to the default **pipeline** with a single click
- Connect directly to your code repository
- Execute the **pipeline** seamlessly
- Get real-time **pipeline** validation
- Load existing **pipelines** or save new ones
- Zoom in and zoom out

The screenshot displays the 'Visual Environment for CI/CD' interface. At the top, there's a status bar with 'Success' and 'All systems operational' indicators, along with 'Repository Connected' and a 'Run Pipeline' button. The main workspace is divided into two panels. The left panel, titled 'Visual Pipeline Definition', shows a drag-and-drop interface with various blocks like 'Pipeline', 'Job', 'Tool', 'Command', 'Parameter', 'Environment', 'Permission', 'Input', 'Output', 'Depends on job', and 'Command Name'. The right panel, titled 'Results', shows the execution log of the pipeline. The log includes steps like 'Initializing pipeline...', 'Fetching repository information...', 'Using default branch: main', 'Getting latest commit information...', 'Latest commit SHA: 9629207', 'Creating workflow file...', 'Creating Blockly workspace file...', 'Setting up directory structure...', 'Committing files to repository...', 'Updating main branch reference...', 'Workflow file 'tutorial\_pipeline-15674.yml' committed successfully to main', 'Waiting for GitHub to register the workflow...', 'Fetching workflow information...', 'Triggering workflow run for "tutorial\_pipeline"...', 'Workflow triggered successfully', 'Waiting for workflow run to start...', 'Workflow run completed with conclusion: success', and 'Fetching job results...'. A 'Copy' button is visible next to the log text.

```
Visual Pipeline Definition
```

Save Load Clear

Results View Run View on GitHub

Results

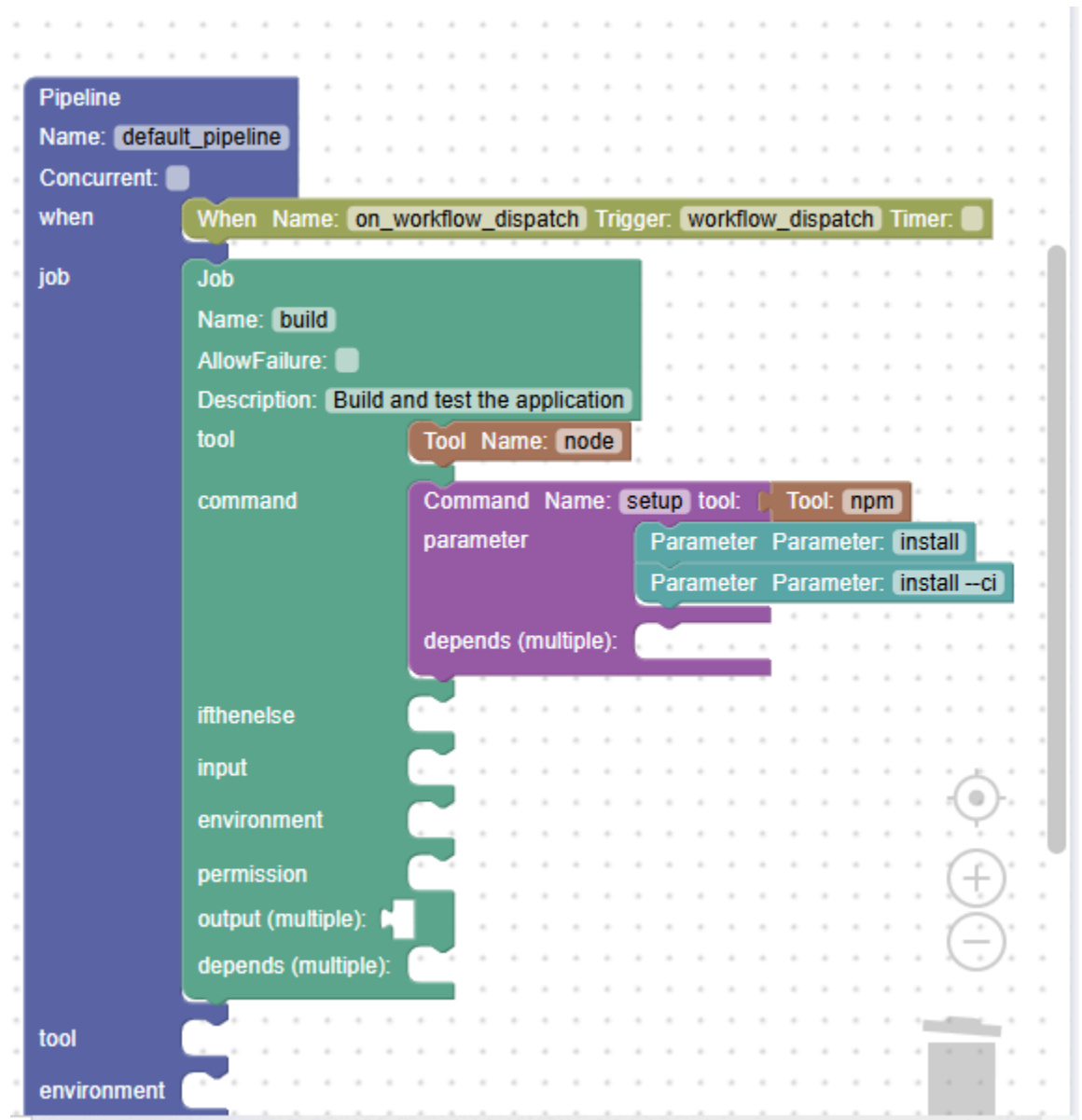
YAML

Copy

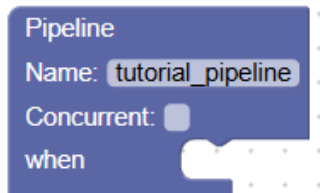
```
Initializing pipeline...
Fetching repository information...
Using default branch: main
Getting latest commit information...
Latest commit SHA: 9629207
Creating workflow file...
Creating Blockly workspace file...
Setting up directory structure...
Committing files to repository...
Updating main branch reference...
Workflow file 'tutorial_pipeline-15674.yml'
committed successfully to main
Waiting for GitHub to register the workflow...
Fetching workflow information...
Triggering workflow run for
"tutorial_pipeline"...
Workflow triggered successfully
Waiting for workflow run to start...
Workflow run completed with conclusion: success
Fetching job results...
Workflow run completed with conclusion: success
```

## 2. The default Pipeline

A simple **pipeline** containing one **when** condition, one **job**, one **tool**, and a single **command** that uses the tool with two **parameters**. This **pipeline** installs npm.



### 3. The Pipeline Block:



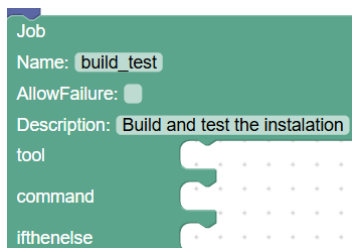
- Update the **pipeline block** and name it "tutorial\_pipeline".

#### 4. When Block:



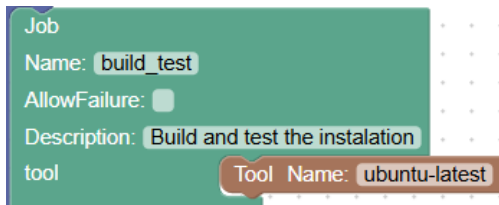
- Update the **when block** in the **pipeline**.
- Name it "on\_workflow\_dispatch".
- Set the trigger to "workflow\_dispatch".

#### 5. Job Block:



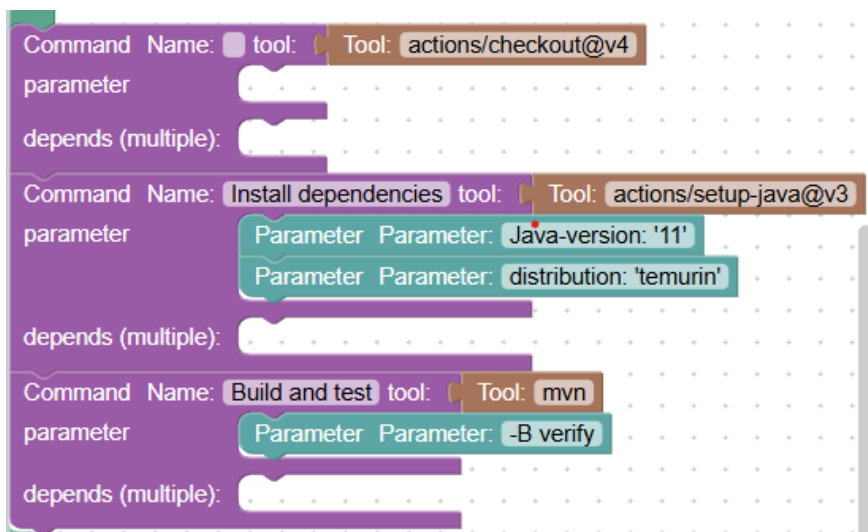
- Update the **job block** to the **pipeline**.
- Name it "build\_test".

#### 6. Update the Tool Block inside the Job:



- Inside the "build\_test" **job**, add a **tool** block.
- Set the tool **parameter** to "ubuntu-latest".

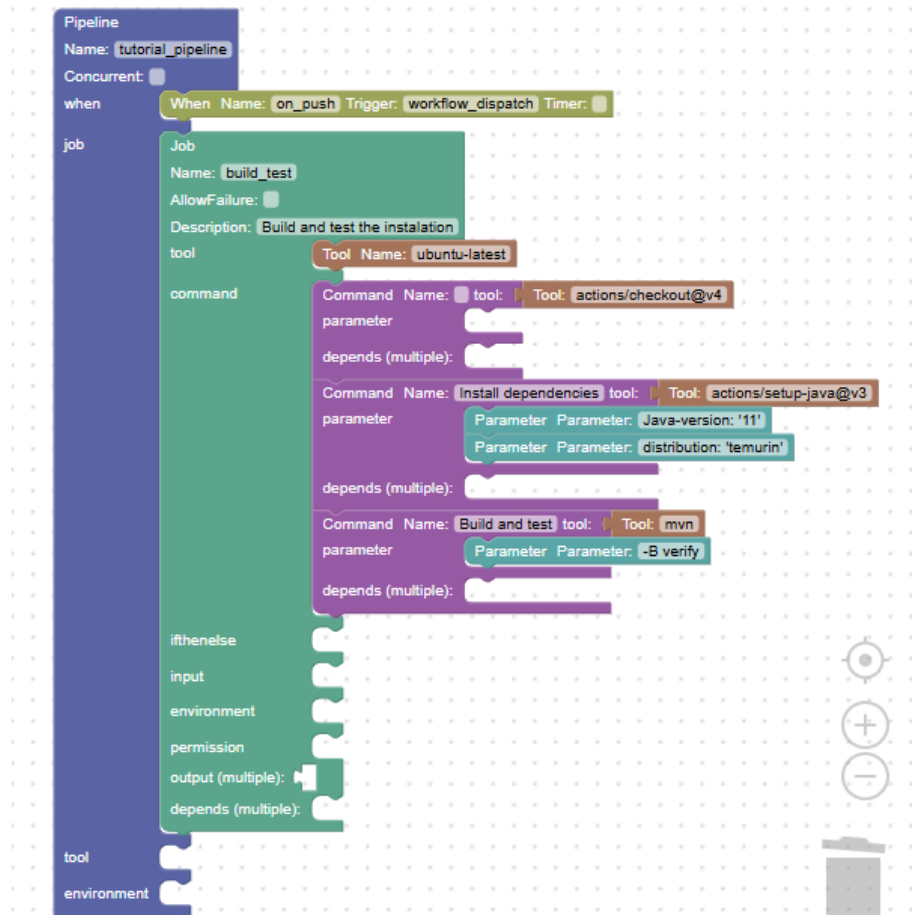
## 7. Add Commands and Parameters:



- Insert **command blocks** within a **job**, positioning them before or after existing **blocks**
- Each default **command block** includes one **tool** and one **parameter**
- Edit, delete, or rearrange **blocks** to match the output shown in the image
- Below is the list of **commands** contained in the **job** block
  - Add a **command** with the **tool** "actions/checkout@v4".
  - Add a **command** with the **tool** "actions/setup-java@v3".
    1. Add a **parameter** with the value "Java-version: '11'".
    2. Add a **parameter** with the value "distribution: 'temurin'".
  - Add a **command** with the **tool** "mvn".

1. Add a **parameter** with the value *"-B verify"*.

## 8. The complete Pipeline



## 9. Connect to the repository

Visual Environment for CI/CD Ready

[Connect to Repo](#) [Run Pipeline](#)

All systems operational. Pipeline configuration is valid.

Visual Pipeline Definition

Save Load Clear

Pipeline Name:  Concurrent: ☐ when: ☐ job: ☐ tool: ☐ environment: ☐ permission: ☐

Job Name:  AllowFailure: ☐ Description:  tool: ☐

Command Name:  Tool Name:  Tool:

parameter

depends (multiple):

Command Name:  Tool:  Tool:

parameter

Parameter:  Parameter:

depends (multiple):

Command Name:  Tool:  Tool:

parameter

Parameter:  Parameter:

depends (multiple):

Results

Results YAML

```
# Generated CI/CD Pipeline YAML
# This YAML is compatible with GitHub Actions

# Pipeline: tutorial_pipeline
name: tutorial_pipeline

jobs:
  build_test:
    # Build and test the installation
    runs-on: ubuntu-latest
    steps:
      - name: command_87733
        uses: actions/checkout@v4
      - name: Install dependencies
        uses: actions/setup-java@v3
        with:
          java-version: '11'
          distribution: 'temurin'
```

Connect to Repository

Enter your repository details and API key to connect your pipeline.

Repository

API Key  [👁](#)

Connect

## 10. Run the Pipeline

Visual Environment for CI/CD Ready Connect to Repo Run Pipeline

All systems operational. Pipeline configuration is valid.

Visual Pipeline Definition

Save Load Clear

Pipeline Name: Concurrent: when: job: tool: environment: permission:

Job Name: AllowFailure: Description: tool: command:

Description: Build and test the installation

Tool Name: ubuntu-latest

Command Name: tool: actions/checkout@v4

parameter

depends (multiple):

Command Name: Install dependencies tool: actions/setup-java@v3

parameter

Parameter Parameter: Java-version: '11'

Parameter Parameter: distribution: 'temurin'

depends (multiple):

Command Name: Build and test tool: mvn

parameter

Parameter Parameter: -B verify

Parameter Parameter: || true

depends (multiple):

Results

Results YAML

```
# Generated CI/CD Pipeline YAML
# This YAML is compatible with GitHub Actions

# Pipeline: tutorial_pipeline
name: tutorial_pipeline

jobs:
  build_test:
    # Build and test the installation
    runs-on: ubuntu-latest
    steps:
      - name: command_87733
        uses: actions/checkout@v4
      - name: Install dependencies
        uses: actions/setup-java@v3
        with:
```

## 11. Results

The outcome should be successful.

Results [View Run](#)

[View on GitHub](#)

> Results

YAML

```
Creating Blockly workspace file...
Setting up directory structure...
Committing files to repository...
Updating main branch reference...
✅ Workflow file 'tutorial_pipeline-82128.yml'
committed successfully to main
Waiting for GitHub to register the workflow...
Fetching workflow information...
Triggering workflow run for
"tutorial_pipeline"...
✅ Workflow triggered successfully
Waiting for workflow run to start...
Workflow run completed with conclusion: success
Fetching job results...
Workflow run completed with conclusion: success
Job: build_test
Status: completed, Conclusion: success
Fetching logs for job "build_test"...
Could not retrieve logs for this job.
✅ View complete run at:
https://github.com/h4g0/testrepository/actions/runs
```