Using conditions to control job execution

Prevent a job from running unless your conditions are met.

Overview 🔗

> **Note:** A job that is skipped will report its status as "Success". It will not prevent a pull request from merging, even if it is a required check.

You can use the `jobs.<job_id>.if` conditional to prevent a job from running unless a condition is met. You can use any supported context and expression to create a conditional. For more information on which contexts are supported in this key, see "[Contexts](#)."

> **Note:** The `jobs.<job_id>.if` condition is evaluated before `jobs.<job_id>.strategy.matrix` is applied.

When you use expressions in an `if` conditional, you can, optionally, omit the `${{ }}` expression syntax because GitHub Actions automatically evaluates the `if` conditional as an expression. However, this exception does not apply everywhere.

You must always use the `${{ }}` expression syntax or escape with `''`, `""`, or `()` when the expression starts with `!`, since `!` is reserved notation in YAML format. For example:

```
if: ${{ ! startsWith(github.ref, 'refs/tags/') }}
```

For more information, see "[Expressions](#)."

Example: Only run job for specific repository 🔗

This example uses `if` to control when the `production-deploy` job can run. It will only run if the repository is named `octo-repo-prod` and is within the `octo-org` organization. Otherwise, the job will be marked as *skipped*.

```yaml
name: example-workflow
on: [push]
jobs:
  production-deploy:
    if: github.repository == 'octo-org/octo-repo-prod'
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
      - uses: actions/setup-node@v3
        with:
          node-version: '14'
      - run: npm install -g bats
```

On a skipped job, you should see "This check was skipped."

> **Note:** In some parts of the workflow you cannot use environment variables. Instead you can use contexts to access the value of an environment variable. For more information, see "[Variables](#)."

Legal