# DevOps and Deployment Documentation

## Section 1: Git Branching and Merging

In Git, a branch is a separate line of development that allows you to work on different features or fix bugs without affecting the main codebase. This is particularly useful for collaborating on projects where multiple developers might be working on different aspects of the application simultaneously.

To create a new branch, use the following command:  
`git checkout -b <branch-name>`  
This command creates a new branch and switches to it immediately.

Once changes are made, you can commit them to the branch using:  
`git commit -m 'Your commit message'`  
After committing changes, you can merge the branch back into the main branch using:  
`git checkout main`  
`git merge <branch-name>`  
This merges the changes from the specified branch into the main branch.

In case of a merge conflict, Git will notify you, and you'll need to manually resolve the conflicts in the files indicated. After resolving conflicts, you need to stage the changes and commit again to complete the merge:  
`git add .`  
`git commit -m 'Resolved merge conflicts'`

## Section 2: Deployment

To deploy the application built in Sections 1 and 2, you can choose a cloud platform such as AWS, Heroku, or Vercel. Below is a brief outline of the deployment process on Heroku, which is straightforward for Node.js applications:

### Deployment Steps:

1. \*\*Create a Heroku Account:\*\* If you don’t have an account, sign up at Heroku.  
2. \*\*Install the Heroku CLI:\*\* Download and install the Heroku Command Line Interface (CLI).  
3. \*\*Log in to Heroku:\*\* Use the command `heroku login` to authenticate your account in the CLI.  
4. \*\*Create a new Heroku App:\*\* Use `heroku create <app-name>` to create a new application.  
5. \*\*Set Up Environment Variables:\*\* Use `heroku config:set VARIABLE\_NAME=value` to manage environment variables.  
6. \*\*Deploy the Application:\*\* You can push your code to Heroku using:  
`git push heroku main`  
7. \*\*Open the Application:\*\* After deployment, open the application using `heroku open`.

### Scaling the Application:

Heroku allows you to scale your application easily. You can adjust the number of dynos (containers) to handle increased traffic. Use the command:  
`heroku ps:scale web=<number-of-dynos>`  
to scale the number of web dynos according to your needs.