

ASSIGNMENT 2

Title

To demonstrate the use of PaaS tools.

Aim

To demonstrate the use of Heroku PaaS tools.

Objective

- To study & implement use of different PaaS tools.
- To implement & test programs on real cloud env.

Theory

Heroku

- cloud platform as a service provider.
- Supports Ruby, Java, Node, Scala, Clojure, PHP, Python.
- To build, run & scale applications.
- Applications have unique domain used to ~~connect~~ ^{route} HTTP calls to the container (dyno) running the program.
- Each dyno is spread across dyno grid which consists of several servers.
- All Heroku's services are hosted on Amazon's EC2 platform.

Heroku Platform

- Runs customer's apps in virtual containers executing on reliable runtime environment.
- Scaling instantly by increasing number of dynos or changing dyno type.

Heroku Postgres

- cloud database (DBaaS) service for Heroku using PostgreSQL.
- Features like conn^ous protection, roll backs & high availability.

Steps to deploy app on Heroku :

- Install the Heroku CLI w/ the following command (for linux)

→ `curl https://cli-assets.heroku.com/install.sh | sh`

- Run

→ `heroku login`

& login from the opened browser.

- Use the following command to create a new app

→ `heroku apps:create app-name`

- change Directory to the source code, & make sure it is initialised as a git repo.

- create a file named 'Procfile' & declare the process to be run in the format

process-type : command_to_execute

- Setup remote tracking to heroku with

→ `heroku git:remote -a app-name`

- Push code to heroku & deploy w/

→ `git push heroku master`

- The application is accessible at `https://app-name.herokuapp.com`

<https://app-name.herokuapp.com>.

Input

Routes

Output

Rendered pages as site.

Software

- Editor / IDE : PyCharm
- Python (Django framework)
- Heroku CLI

Conclusion

Thus we have successfully completed & studied PaaS (Heroku) & implemented on real cloud environment.

FAQs

Q. What are the different environments supported by your PaaS ?

A. Heroku supports the following languages natively :

- Node.js
- PHP
- Scala
- Ruby
- Python
- Clojure
- Java
- Go

Other languages can be run w/ third party buildpacks.
You can also use Docker images to directly deploy a containerised application.

Q. What are the features supported by your PaaS ?

- A. - Heroku Postgres : postgresql as a service
- Heroku Redis : In-memory key-value store as a service.
 - Scaling : Both vertical & horizontal.
 - Add-ons : Pre-migrated services
 - Data clips : data insights from project.
 - CI/CD : Heroku Flow, Pipelines
 - GitHub Integration : connect to a repo & autodeploy commits pushed to a certain branch.
 - Extensibility : using community built add on build packs.