### **Task 1 Postgresql - Reflection**

1. **What is the need for Add Ons in Heroku?**

Heroku add-ons are components that support your application, such as data storage, monitoring, analytics, data processing, and more.

These are fully maintained for you by either a third-party provider or by Heroku. Add-ons exist so that developers can focus on their own application logic, and not the additional complexity of keeping supporting services running at full production capacity.

Heroku Add-ons are integrated into your Heroku platform experience, making it easy to install new services and manage billing, credentials, or configurations directly from your Heroku Dashboard or CLI.

1. **What exactly happens when you click on provision while configuring the Postgres addon?**

When we click on Provision while configuring Postgres addon, the addon will be installed and you will be having a working Postgres database. It provides us with 10000 rows which are free to use. We can get credentials from the settings and these are used to login on Adminer.

1. **What is the use of Adminer? How does it work?**

Adminer is a tool similar to phpMyAdmin. It is used to run SQL commands on a GUI interface. We can access the databases via Adminer by logging into it using credentials from the Heroku Credentials page.

### **Task 2 Python and Flask - Reflection**

1. **How do I manage to use python 3.6 if I already have python 2.7?**

We can use both the python versions in your system. After downloading python 3 add it to the path in Environment variables. Then navigate to Python2 and Python3 folders and change the .exe files to Python2 and Python3. Then open the command prompt and type python2 to use Python 2.7 and python3 to use Python 3.6.

1. **What is the role of pip and how does it work?**

pip is a standard package-management system used to install and manage software packages written in Python. Many packages can be found in the default source for packages and their dependencies — Python Package Index (PyPI).

You use pip with an install command followed by the name of the package you want to install. pip looks for the package in PyPI, calculates its dependencies, and installs them to insure requests will work.

1. **What is the role of requirements.txt and how does it work with pip?**

When we want to install all the dependencies required for the program using pip. All of the required dependencies list should be in the requirements.txt file. Then by running

**pip install -r requirements.txt**

we get all the installation done rather than installing each at a time.

1. **Which packages are installed and why are they required?**

The packages that are installed are Flask, Flask-Session, psycopg2-binary, SQLAlchemy.

1. **Which environment variables set for Flask to work? What is the purpose of each variable?**

We use the export command to set environment variables.

We set the FLASK\_APP variable to tell the start point of the application i.e what file to run after the run command is used.

We set the FLASK\_DEBUG to 1 to automatically apply a change to the web app when the file is changed or modified.

Set up the DATABASE\_URL to establish the communication between the flask application and database.

1. **What happens when the Flask run command is issued on the terminal?**

When the flask run command is issued on the terminal it runs the web server and gives you on which URL it is running on. Generally it is http://127.0.0.1:5000/.

1. **On which port is Flask running and can it be changed?**

Flask is running on port number 5000. It can be changed using

$ export FLASK\_RUN\_PORT=8000

$ flask run

\* Running on http://127.0.0.1:8000/

1. **How is Flask different from the tiny web server?**

Flask is a micro web application framework, that means it is basically a set of tools and libraries that make it easier to build web applications in Python.

Flask does however include a web server that can be used for testing and development. But when you're ready to host your app or put it into production, you should choose a different web server to use with it. Flask also enables you to choose the URI, whereas in webserver you have to parse the URI.

### **Task 3 Goodreads API - Reflection**

1. **What are the various categories of web APIs available on good reads?**

Some of the categories of web APIs available on good reads are

* Get Goodreads book IDs given ISBNs
* Get Goodreads work IDs given Goodreads book IDs
* Get review statistics given a list of ISBNs
* Get the reviews for a book given a Goodreads book id
* Get the reviews for a book given an ISBN

1. **Is there a limit on the use of the web API? What are the limits?**

Yes there is a limit on the use of web API. One such limit is to not request any method more than once a second. Goodreads tracks all requests made by developers.

Detailed limits and terms and conditions are provided in the link below.

<https://www.goodreads.com/api/terms>