

Flask - Database - Heroku

(code with Harry)

```

# https://flask.palletsprojects.com/en/2.1.x/quickstart/
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello_world():
    return "<p>Hello, World!</p>"

if __name__ == "__main__":
    app.run(debug=True) # debug=True --- this shows the error message in the browser itself (if any)

```

can also run from the command line using
python app.py

default port for Flask = 5000

```

# https://flask.palletsprojects.com/en/2.1.x/quickstart/
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello_world():
    return "<p>Hello, World!</p>"

if __name__ == "__main__":
    app.run(debug=True, port=8000) # debug=True

```

Rewritten code: changed the port

Error: because we gave the port = 8000

This site can't be reached
127.0.0.1 refused to connect.
Try:
• Checking the connection
• Checking the proxy and the firewall
ERR_CONNECTION_REFUSED

```

# https://flask.palletsprojects.com/en/2.1.x/quickstart/
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello_world():
    return "Hello, World!"

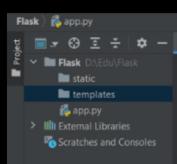
@app.route("/about")
def about():
    return "This is the about page"

if __name__ == "__main__":
    app.run(debug=True) # debug=True --- this shows the error message in the browser itself (if any)

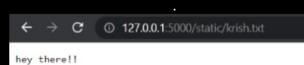
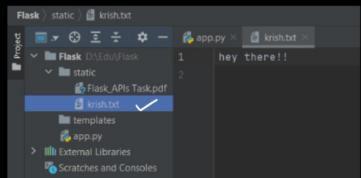
```

'/' means the homepage

→ Create folders: static, templates.

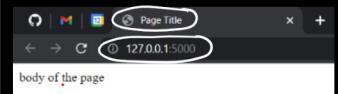
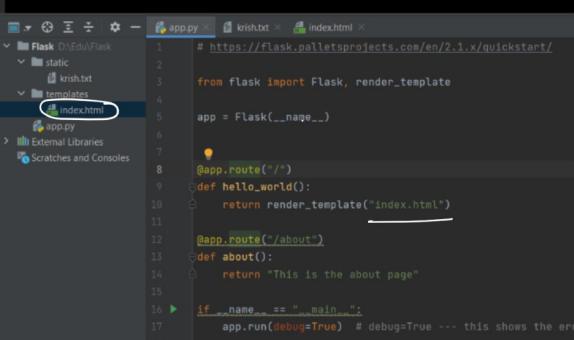
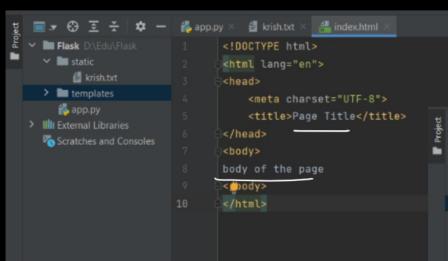


→ whatever is written in the files present in 'static', their content will be displayed in the browser.



→ In 'templates' folder, we keep the templates (HTML files)

⇒ we can render the HTML files & display the required text in the browser.



⇒ getbootstrap.com > get started.

↳ In the index.html, delete everything & paste in index.html

COPY getbootstrap.com > get started > Starter template → paste in index.html

Starter template

Be sure to have your pages set up with the latest design and development standards. That means using an HTML5 doctype and including a viewport meta tag for proper responsive behaviors. Put it all together and your pages should look like this:

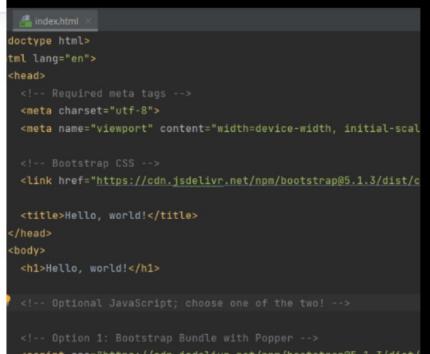
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

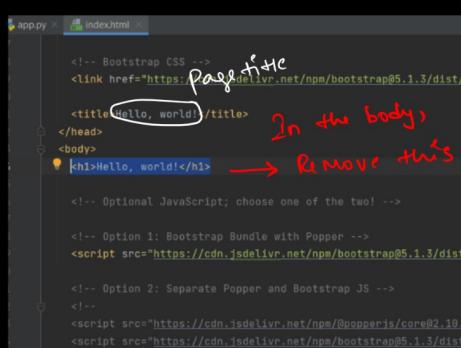
    <title>Hello, world!</title>
  </head>
  <body>
    <h1>Hello, world!</h1>

    <!-- Optional JavaScript; choose one of the two! -->

    <!-- Option 1: Bootstrap Bundle with Popper -->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-E7wVXoCzT6lW86ZsdEj7Wx8n+L9y91yOppeKuqCkqOjy+oQGgA==" crossorigin="an
```



2, Adding Navbar → Copy the content of Navbar (present in Components)



In the body
→ Remove this & paste Navbar Content.



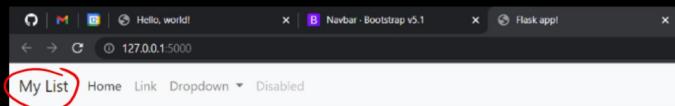
3, Change the Name Navbar to myList

```
<!-- Bootstrap CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

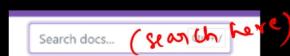
<title>Flask app!</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <div class="container-fluid">
      <a class="navbar-brand" href="#">Navbar</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent">
        <span class="navbar-toggler-icon"></span>
      </button>
    </div>
```

```
<!-- Bootstrap CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

<title>Flask app!</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <div class="container-fluid">
      <a class="navbar-brand" href="#">My List</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent">
        <span class="navbar-toggler-icon"></span>
      </button>
    </div>
```



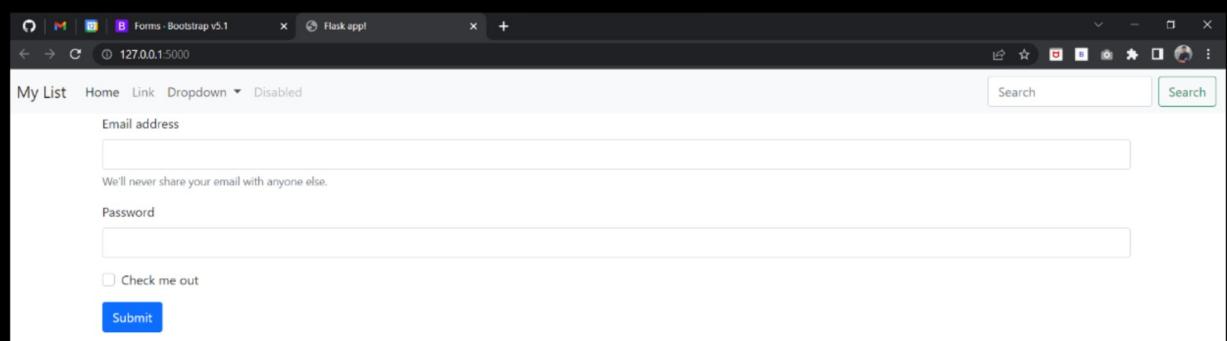
4, Adding the form → Copy forms overview



- > Getting started
- > Customize
- > Layout

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
  <title>Flask app!</title>
  </head>
  <body>
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
      <div class="container">
```

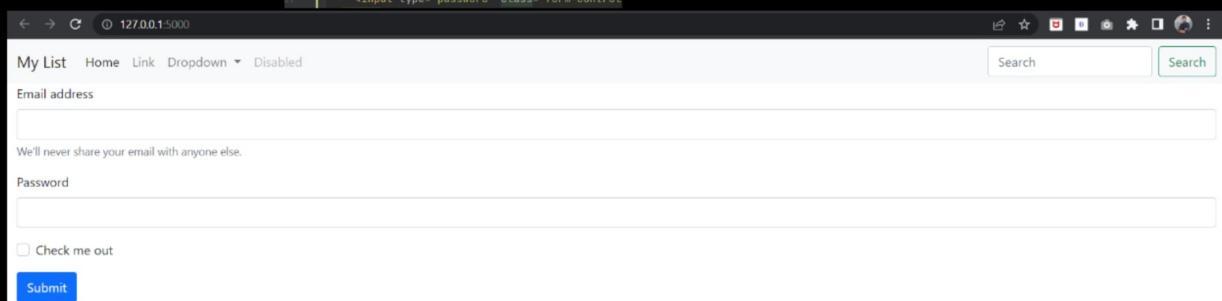
collapsed navbar
content
Create container class & paste the 'forms overview'
Content in it.



→ Instead of container, if I use 'container-fluid'

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
  <title>Flask app!</title>
  </head>
  <body>
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
      <div class="container-fluid">
```

the fields width
will be the entire
page's width.



5. Adding Table - Copy Content of 'table overview'

The image shows a code editor window with `index.html` open and a browser window displaying a Flask application.

Code Editor (index.html):

```
7 <!-- Bootstrap CSS -->
8 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/
9 ...
10 ...
11 ...
12 ...
13 ...
14 ...
15 ...
16 ...
17 ...
18 ...
19 ...
20 ...
21 ...
22 ...
23 ...
24 ...
25 ...
26 ...
27 ...
28 ...
29 ...
30 ...
31 ...
32 ...
33 ...
34 ...
35 ...
36 ...
37 ...
38 ...
39 ...
40 ...
41 ...
42 ...
43 ...
44 ...
45 ...
46 ...
47 ...
48 ...
49 ...
50 ...
51 ...
52 ...
53 ...
54 ...
55 ...
56 ...
57 ...
58 ...
59 ...
60 ...
61 ...
62 ...
63 ...
64 ...
65 ...
66 ...
67 ...
68 ...
69 ...
70 ...
71 ...
72 ...
73 ...
74 ...
75 ...
76 ...
77 ...
78 ...
79 ...
80 ...
81 ...
82 ...
83 ...
84 ...
85 ...
86 ...
87 ...
88 ...
89 ...
90 ...
91 ...
92 ...
93 ...
94 ...
95 ...
96 ...
97 ...
98 ...
```

Annotations:

- An arrow points from the handwritten note "create container" to the opening `<div class='container'>` tag.
- An arrow points from the handwritten note "parse the table's content." to the opening `<table>` tag.

Browser Window (Flask app):

My List Home Link Dropdown Disabled

Email address
[Input field]

We'll never share your email with anyone else.

Password
[Input field]

Check me out

Submit

#	First	Last	Handle
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

— X —

* Using Database in Flask:

↳ Create a database / table class — we need to inform flask that what we will store in the table → then flask will create the db/table for us.

→ to use database in flask, we need to install 'flask-sqlalchemy' package.

⇒ pip install flask-sqlalchemy

```
Terminal Local Command Prompt + v
(Flask) D:\Edu\Flask>pip install flask-sqlalchemy
```

```
Terminal Local Command Prompt + v
Requirement already satisfied: dataclasses in c:\users\krishnaleti\anaconda3\envs\flask
Requirement already satisfied: zipp>=0.5 in c:\users\krishnaleti\anaconda3\envs\flask
Requirement already satisfied: typing-extensions>=3.6.4 in c:\users\krishnaleti\anaconda3\envs\flask (4.1.1)
Installing collected packages: greenlet, SQLAlchemy, flask-sqlalchemy
Successfully installed SQLAlchemy-1.4.35 flask-sqlalchemy-2.5.1 greenlet-1.1.2
```

* SQLAlchemy is an ORM mapper → This facilitates to make changes in the database from python

Configuration Keys

A list of configuration keys currently understood by the extension:

SQLALCHEMY_DATABASE_URI	The database URI that should be used for the connection. Examples:
• sqlite:///tmp/test.db	
• mysql://username:password@server/db	

we get few code snippets from this url

```
app.py index.html
1 # https://flask.palletsprojects.com/en/2.1.x/quickstart/
2
3 from flask import Flask, render_template
4 from flask_sqlalchemy import SQLAlchemy # importing sqlalchemy
5 app = Flask(__name__)
6
7 # configuring the database
8 app.config['SQLALCHEMY_DATABASE_URI'] = "mysql://test:password@localhost/flask_tododb"
9 # mysql://username:password@server/db --- defining the mysql database
10 db = SQLAlchemy(app) # initializing the database
11
12 @app.route("/")
13 def hello_world():
14     return render_template("index.html")
```

* Create a database in Sql workbench and give that database name while defining the db.

```
app
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
C:\Users\KrishnaLeti\anaconda3\envs\Flasklib\site-packages\flask_sqlalchemy\_init_.py:873: FSDeprecationWarning: SQLALCHEMY_TRACK_MODIFICATIONS adds significant overhead and
'SQLALCHEMY_TRACK_MODIFICATIONS' adds significant overhead and '
SQLALCHEMY_TRACK_MODIFICATIONS' adds significant overhead and '
SQLALCHEMY_TRACK_MODIFICATIONS' adds significant overhead and '
* Debugger is active!
* Debugger PIN: 196-016-742
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

What is SQLAlchemy_track_modifications?

SQLALCHEMY_TRACK_MODIFICATIONS. If set to True , Flask-SQLAlchemy will track modifications of objects and emit signals. The default is None , which enables tracking but issues a warning that it will be disabled by default in the future. This requires extra memory and should be disabled if not needed.

```
app.py index.html
1 # https://flask.palletsprojects.com/en/2.1.x/quickstart/
2
3 from flask import Flask, render_template
4 from flask_sqlalchemy import SQLAlchemy # importing sqlalchemy
5 app = Flask(__name__)
6
7 # configuring the database
8 app.config['SQLALCHEMY_DATABASE_URI'] = "mysql://test:password@localhost/flask_tododb"
9 app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
10 # mysql://username:password@server/db --- defining the mysql database
11 db = SQLAlchemy(app) # initializing the database
12
13 @app.route("/")
14 def hello_world():
15     return render_template("index.html")
```

(MySQL) for database usage with Flask

To install:
1) pip install flask-sqlalchemy
2) pip install pymysql

Give the database name which is created in sql workbench.

```

2
3     from flask import Flask, render_template
4     from flask_sqlalchemy import SQLAlchemy # importing sqlalchemy
5     from datetime import datetime
6
7     app = Flask(__name__)
8
9     # configuring the database
10    # https://flask-sqlalchemy.palletsprojects.com/en/2.x/config/
11    # app.config['SQLALCHEMY_DATABASE_URI'] = mysql://username:password@server/db - defining the mysql database
12    app.config['SQLALCHEMY_DATABASE_URI'] = "mysql://test:password@localhost/flask_tododb"
13    app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
14    db = SQLAlchemy(app) # initializing the database
15
16    class Todo(db.Model): # creating the table
17        sno = db.Column(db.Integer, primary_key=True) # defining the columns
18        title = db.Column(db.String(80), nullable=False)
19        desc = db.Column(db.String(200))
20        date_of_creation = db.Column(db.DateTime, default=datetime.utcnow)
21
22        def __repr__(self):
23            return f"Todo('{self.sno}', '{self.title}')"
24
25
26    """When the object of this class is printed, the __repr__ function will be automatically executed and it will return the values of the sno,title """
27
28
29    flow:
30    Define & create database → Initialize database
31    → create table → define columns → run the app

```

defining the database ✓

obj = Todo()
print(obj) → then the __repr__ function gets automatically executed & whatever written in return will be displayed.

database - flask_db created.

pip install PyMySQL
And adjust your connection string to use PyMySQL, from mysql:// to mysql+pymysql://.

```

(Flask) D:\Edu\Flask>python
Python 3.6.13 |Anaconda, Inc.| (default, Mar 16 2021, 11:37:27) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from app import db
>>> db.create_all()
File "C:\Users\KrishnaAleti\anaconda3\envs\Flasklib\site-packages\flask_sqlalchemy\_init_.py", line 594, in get_engine
    self._engine = rv = self._sa.create_engine(sa_url, options)
File "C:\Users\KrishnaAleti\anaconda3\envs\Flask\lib\site-packages\flask_sqlalchemy\_init_.py", line 1027, in create_engine
    return sqlalchemy.create_engine(sa_url, **engine_opts)
File "c:\string", line 2, in create_engine
File "C:\Users\KrishnaAleti\anaconda3\envs\Flask\lib\site-packages\sqlalchemy\util\deprecations.py", line 309, in warned
    return fn(*args, **kwargs)
File "C:\Users\KrishnaAleti\anaconda3\envs\Flask\lib\site-packages\sqlalchemy\engine\create.py", line 560, in create_engine
    dbapi = dialect_cls.dbapi(*_dopapi_args)
File "C:\Users\KrishnaAleti\anaconda3\envs\Flask\lib\site-packages\sqlalchemy\dialects\mysql\mysqldb.py", line 163, in dbapi
    return __import__('MySQLdb')
ModuleNotFoundError: No module named 'MySQLdb'
>>>

```

```

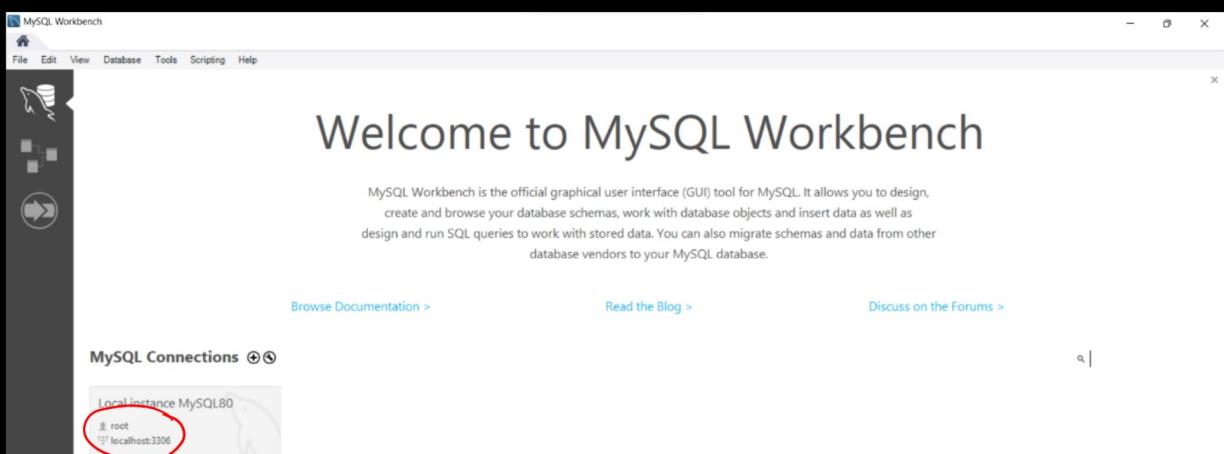
(Flask) D:\Edu\Flask>pip install PyMySQL
Collecting PyMySQL
  Downloading PyMySQL-1.0.2-py3-none-any.whl (43 kB)
    100% |██████████| 43 kB 709 kB/s
Installing collected packages: PyMySQL
Successfully installed PyMySQL-1.0.2
(Flask) D:\Edu\Flask>

```

```

Project app.py index.html
1  # https://flask-sqlalchemy.palletsprojects.com/en/2.1.x/quickstart/
2
3     from flask import Flask, render_template
4     from flask_sqlalchemy import SQLAlchemy # importing sqlalchemy
5     from datetime import datetime
6
7     import pymysql
8
9     import mysql.connector as connection
10    app = Flask(__name__)
11    # configuring the database
12    # https://flask-sqlalchemy.palletsprojects.com/en/2.1.x/config/
13    # app.config['SQLALCHEMY_DATABASE_URI'] = mysql://username:password@server/db - defining the mysql database
14    """The database which we are giving here needs to be created in the sql workbench 1st"""
15    # app.config['SQLALCHEMY_DATABASE_URI'] = "mysql://root:password@localhost/flask_tododb"
16
17    # https://stackoverflow.com/questions/4548545/no-module-named-mysqldb
18    """we need to edit the SQLALCHEMY URL schema like this: mysql+pymysql://username:password@host/database
19    as we are installed PyMySQL"""
20    app.config['SQLALCHEMY_DATABASE_URI'] = 'mysql+pymysql://root:password@127.0.0.1:3306/flaskdb'
21    app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
22    db = SQLAlchemy(app) # initializing the database
23
24    # http://task-sqlalchemy.palletsprojects.com/en/2.x/models/
25    class Todo(db.Model): # creating the table
26        sno = db.Column(db.Integer, primary_key=True) # defining the columns
27        title = db.Column(db.String(80), nullable=False)
28        desc = db.Column(db.String(200))
29        date_of_creation = db.Column(db.DateTime, default=datetime.utcnow)
30
31        def __repr__(self):
32            return f"Todo('{self.sno}', '{self.title}')"

```



```
(Flask) D:\Edu\Flask>python
Python 3.6.13 |Anaconda, Inc.| (default, Mar 16 2021, 11:37:27) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from app import db
>>> db.create_all()
→ Table will be created in the
mentioned database with the defined columns.
```

Table Created.

Tables_in_flaskdb
todo

```
app.py index.html
18 app.config['SQLALCHEMY_DATABASE_URI'] = 'mysql+pymysql://root:password@127.0.0.1:3306/flaskdb'
19 app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
20 db = SQLAlchemy(app) # initializing the database
21
22 # https://flask-sqlalchemy.palletsprojects.com/en/2.x/models/
23 class Todo(db.Model): # creating the table
24     sno = db.Column(db.Integer, primary_key=True) # defining the columns
25     title = db.Column(db.String(80), nullable=False)
26     desc = db.Column(db.String(200))
27     data_of_creation = db.Column(db.DateTime, default=datetime.utcnow)
```

Col. names

→ As we had defined sno. column as primary key, it is set to auto-increment automatically hence, even if we don't pass values into sno. col, the database fills it. (starting with 1)

Field	Type	Null	Key	Default	Extra
sno	int	NO	PRI	1	auto_increment
title	varchar(80)	NO	NO		
desc	varchar(200)	YES	NO		
data_of_creation	datetime	YES	NO		

→ Now passing/inserting values into the table.

```
app.py index.html
41 # https://flask-sqlalchemy.palletsprojects.com/en/2.x/quickstart/#a-minimal-application
42 @app.route('/')
43 def hello_world():
44     todo = Todo(title='first Todo', desc='first description') # instantiating the class - Todo and passing the values
45     db.session.add(todo) # adding the object to the database
46     db.session.commit() # committing the changes to the database
47     return render_template("index.html")
```

Field	Type	Null	Key	Default	Extra
sno	int	NO	PRI	1	auto_increment
title	varchar(80)	NO	NO		
desc	varchar(200)	YES	NO		
data_of_creation	datetime	YES	NO		

```
Flask app.py index.html
Project Flask (D:\Edu\Flask)
  > static
  > templates
    > index.html
  > app.py
  > External Libraries
  > Scratches and Consoles

28 class Todo(db.Model): # creating the table
29     sno = db.Column(db.Integer, primary_key=True) # defining the columns
30     title = db.Column(db.String(80), nullable=False)
31     desc = db.Column(db.String(200))
32     data_of_creation = db.Column(db.DateTime, default=datetime.utcnow)
33     def __repr__(self):
34         return f"Todo('{self.sno}', '{self.title}')"
35     """When the object of this class is printed, the __repr__ function will be automatically executed and
36     it will return the values of the sno,title """
37     # https://flask-sqlalchemy.palletsprojects.com/en/2.x/quickstart/#a-minimal-application
38     @app.route('/')
39     def hello_world():
40         todo = Todo(title='first Todo', desc='first description') # instantiating the class - Todo and passing the values
41         db.session.add(todo) # adding the object to the database
42         db.session.commit() # committing the changes to the database
43         return render_template("index.html")
44     @app.route("/show")
45     def about():
46         todolist = Todo.query.all() # fetching all the data from the table ✓
47         print(todolist)
48         return str(todolist) → tablename.query.all()
```

```
Run: app
127.0.0.1 - - [12/Apr/2022 22:31:21] "GET /show HTTP/1.1" 200 -
[Todo('1', 'first Todo'), Todo('2', 'first Todo'), Todo('3', 'first Todo')]
127.0.0.1 - - [12/Apr/2022 22:31:22] "GET /show HTTP/1.1" 200 -
127.0.0.1 - - [12/Apr/2022 22:31:23] "GET /show HTTP/1.1" 200 -
[Todo('1', 'first Todo'), Todo('2', 'first Todo'), Todo('3', 'first Todo')]
```

```
127.0.0.1:5000/show
[Todo('1', 'first Todo'), Todo('2', 'first Todo'), Todo('3', 'first Todo')]
```

```

app.py
38 @app.route('/')
39 def hello_world():
40     todo = Todo(title='first Todo', desc='first description') # instantiating the class - Todo and passing the values
41     db.session.add(todo) # adding the object to the database
42     db.session.commit() # committing the changes to the database
43     todolist = Todo.query.all() # fetching all the data from the table
44     return render_template("index.html", todolist=todolist)
45 @app.route("/show")
46 def about():
47     todolist = Todo.query.all() # fetching all the data from the table
48     print(todolist)
49     return str(todolist)
50
51
52 if __name__ == "__main__":
53     app.run(debug=True) # debug=True --- this shows the error message in the browser itself (if any)
54
55

```

```

index.html
14 <nav class="navbar navbar-expand-lg navbar-light bg-light"...>
15 <div class="container"...>
16 <div class="container my-3">
17     <h2> Todo list </h2>
18
19     <table class="table">
20         <thead>
21             <tr>
22                 <th scope="col">SNo</th>
23                 <th scope="col">Todo Title</th>
24                 <th scope="col">Description</th>
25                 <th scope="col">Created_DateTime</th>
26             </tr>
27         </thead>
28         <tbody>
29             {% for todo in todolist %}
30                 <tr>
31                     <th scope="row">{{todo.sno}}</th>
32                     <td>{{todo.title}}</td>
33                     <td>{{todo.desc}}</td>
34                     <td>{{todo.data_of_creation}}</td>
35                 </tr>
36             {% endfor %}
37         </tbody>
38     </table>
39 
```

My List Home Link Dropdown Disabled

Add a ToDo

Todo Title

Todo Description

Submit

Todo list

SNo	Todo Title	Description	Created DateTime
1	first Todo	first description	2022-04-12 16:49:41
1	first Todo	first description	2022-04-12 16:58:33
1	first Todo	first description	2022-04-12 17:01:13
1	first Todo	first description	2022-04-12 17:17:10

Query 1

```

1 • create database flaskdb;
2 • show databases;
3 • use flaskdb;
4 • show tables;
5 • desc todo;
6 • select * from Todo;
7

```

Result Grid | Filter Rows | Edit | Export/Import | Wrap Cell Content:

sno	title	desc	data_of_creation
1	first Todo	first description	2022-04-12 16:49:41
2	first Todo	first description	2022-04-12 16:58:33
3	first Todo	first description	2022-04-12 17:01:13
4	first Todo	first description	2022-04-12 17:17:10
5	first Todo	first description	2022-04-12 17:24:35
6	first Todo	first description	2022-04-12 17:25:09
7	first Todo	first description	2022-04-12 17:25:38

app.py

```

16 <nav class="navbar navbar-expand-lg navbar-light bg-light"...>
17 <div class="container"...>
18 <div class="container my-3">
19     <h2> Todo list </h2>
20
21     <table class="table">
22         <thead>
23             <tr>
24                 <th scope="col">SNo</th>
25                 <th scope="col">Todo Title</th>
26                 <th scope="col">Description</th>
27                 <th scope="col">Created_DateTime</th>
28             </tr>
29         </thead>
30         <tbody>
31             {% for todo in todolist %}
32                 <tr>
33                     <th scope="row">{{todo.sno}}</th>
34                     <td>{{todo.title}}</td>
35                     <td>{{todo.desc}}</td>
36                     <td>{{todo.data_of_creation}}</td>
37                 </tr>
38             {% endfor %}
39         </tbody>
40     </table>
41 
```

My List Home Link Dropdown Disabled

Add a ToDo

Todo Title

Todo Description

Submit

Todo list

SNo	Todo Title	Description	Created DateTime
1	first Todo	first description	2022-04-12 16:49:41
2	first Todo	first description	2022-04-12 16:58:33
3	first Todo	first description	2022-04-12 17:01:13
4	first Todo	first description	2022-04-12 17:17:10
5	first Todo	first description	2022-04-12 17:24:35
6	first Todo	first description	2022-04-12 17:25:09
7	first Todo	first description	2022-04-12 17:25:38

Here what happens, is suppose if we delete 6th record then I will have sno. as 1,2,3,4,5,7 we need to fix this. Use loop.index instead of todo.sno

Query 1

```

2 • show databases;
3 • use flaskdb;
4 • show tables;
5 • desc todo;
6 • select * from Todo;
7 • delete from flaskdb.todo where sno=6;
8 • select * from Todo;

```

Result Grid | Filter Rows | Edit | Export/Import | Wrap

sno	title	desc	data_of_creation
1	first Todo	first description	2022-04-12 16:49:41
2	first Todo	first description	2022-04-12 16:58:33
3	first Todo	first description	2022-04-12 17:01:13
4	first Todo	first description	2022-04-12 17:17:10
5	first Todo	first description	2022-04-12 17:24:35
6	first Todo	first description	2022-04-12 17:25:38
7	first Todo	first description	2022-04-12 17:30:26
8	first Todo	first description	2022-04-12 17:32:14

Add a ToDo

Todo Title
Todo Description
Submit

Todo list

SNo	Todo Title	Description	Created DateTime
1	first Todo	first description	2022-04-12 16:49:41
2	first Todo	first description	2022-04-12 16:58:33
3	first Todo	first description	2022-04-12 17:01:13
4	first Todo	first description	2022-04-12 17:17:10
5	first Todo	first description	2022-04-12 17:24:35
7	first Todo	first description	2022-04-12 17:25:38
8	first Todo	first description	2022-04-12 17:30:26
9	first Todo	first description	2022-04-12 17:32:14
10	first Todo	first description	2022-04-12 17:33:58

index.html

```

14 <nav class="navbar navbar-expand-lg navbar-light bg-light">
15   <div class="container">...</div>
16   <div class="container my-3">
17     <h2> Todo list </h2>
18     <!-- todo li -->
19     <table class="table">
20       <thead>
21         <tr>
22           <th scope="col">SNo</th>
23           <th scope="col">Todo Title</th>
24           <th scope="col">Description</th>
25           <th scope="col">Created DateTime</th>
26         </tr>
27       </thead>
28       <tbody>
29         {% for todo in todolist %}
30           <tr>
31             <th scope="row">{{loop.index}}</th>
32             <td>{{todo.title}}</td>
33             <td>{{todo.desc}}</td>
34             <td>{{todo.data_of_creation}}</td>
35           </tr>
36         {% endfor %}
37       </tbody>
38     </table>
39   </div>

```

Add a ToDo

Todo Title
Todo Description
Submit

Todo list

SNo	Todo Title	Description	Created DateTime
1	first Todo	first description	2022-04-12 16:49:41
2	first Todo	first description	2022-04-12 16:58:33
3	first Todo	first description	2022-04-12 17:01:13
4	first Todo	first description	2022-04-12 17:17:10
5	first Todo	first description	2022-04-12 17:24:35
6	first Todo	first description	2022-04-12 17:25:38
7	first Todo	first description	2022-04-12 17:30:26

My List

Add a ToDo

Todo Title
Todo Description
Submit

Todo list

SNo	Todo Title	Description	Created DateTime
1	first Todo	first description	2022-04-12 16:49:41
2	first Todo	first description	2022-04-12 16:58:33
3	first Todo	first description	2022-04-12 17:01:13
4	first Todo	first description	2022-04-12 17:17:10
5	first Todo	first description	2022-04-12 17:24:35
6	first Todo	first description	2022-04-12 17:25:38
7	first Todo	first description	2022-04-12 17:30:26

we get the SNo property in the browser.

(Now we are not fetching the SNo from database, we are getting it from loop.index)

Code with Harry's Code:

Used SQLite db.

My Code - Used my Sqlite db

give the db created from scratch

Table name

```

22 app.config['SQLALCHEMY_DATABASE_URI'] = 'mysql+pymysql://root:password@127.0.0.1:3306/flaskdb'
23 app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
24 db = SQLAlchemy(app) # initializing the database
25 # https://flask-sqlalchemy.palletsprojects.com/en/2.x/models/
26 class Todo(db.Model): # creating the table
27   # defining the columns
28   sno = db.Column(db.Integer, primary_key=True)
29   title = db.Column(db.String(200), nullable=False)
30   desc = db.Column(db.String(200))
31   data_of_creation = db.Column(db.DateTime, default=datetime.utcnow)
32
33   def __repr__(self):
34     return f'Todo({self.sno}, {self.title})'
35
36   """When the object of this class is printed, the __repr__ function will be automatically executed and it will return the values of the sno,title """
37   """As we had given the sno filed as int type and serial number, the database (mysql) will automatically give the integer number to that field and it will be incremented automatically
38   https://retool.com/blog/how-to-work-with-auto-incrementing-ids-in-sql/ """
39
40   # https://flask-sqlalchemy.palletsprojects.com/en/2.x/quickstart/#a-minimal-application
41   @app.route('/', methods=['GET', 'POST'])
42   def hello_world():
43     if request.method == 'POST':
44       title = request.form.get('title')
45       desc = request.form.get('desc')
46       todo = Todo(title=title, desc=desc) # instantiating the class - Todo and passing the values
47       db.session.add(todo) # adding the object to the database
48       db.session.commit() # committing the changes to the database
49       todolist = Todo.query.all() # fetching all the data from the table
50     return render_template("index.html", todolist=todolist)
51

```

```

1 from flask import Flask, render_template, request, redirect
2 from flask_sqlalchemy import SQLAlchemy
3 from datetime import datetime
4
5 app = Flask(__name__)
6 app.config['SQLALCHEMY_DATABASE_URI'] = "sqlite:///todo.db"
7 app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
8 db = SQLAlchemy(app)
9
10 class Todo(db.Model):
11   sno = db.Column(db.Integer, primary_key=True)
12   title = db.Column(db.String(200), nullable=False)
13   desc = db.Column(db.String(200), nullable=False)
14   date_created = db.Column(db.DateTime, default=datetime.utcnow)
15
16   def __repr__(self):
17     return f'{self.sno} - {self.title}'
18
19 @app.route('/', methods=['GET', 'POST'])
20 def hello_world():
21   if request.method=='POST':
22     title = request.form['title']
23     desc = request.form['desc']
24     todo = Todo(title=title, desc=desc)
25     db.session.add(todo)
26     db.session.commit()
27
28     allTodo = Todo.query.all()
29     return render_template('index.html', allTodo=allTodo)
30
31 @app.route('/show')
32 def productsO():
33   allTodo = Todo.query.all()
34   print(allTodo)
35   return 'this is products page'
36
37 @app.route('/update/<int:sno>', methods=['GET', 'POST'])
38 def update(sno):
39   if request.method=="POST":
40     title = request.form['title']
41     desc = request.form['desc']
42     todo = Todo.query.filter_by(sno=sno).first()
43     todo.title = title
44     todo.desc = desc
45     db.session.add(todo)
46     db.session.commit()
47     return redirect("/")
48
49 todo = Todo.query.filter_by(sno=sno).first()
50 return render_template('update.html', todo=todo)

```

* Deployment in Heroku:

- ↳ Create free account in heroku.com
- 2, Download & install Heroku CLI (from browser) — Restart PyCharm
- 3, install gunicorn \Rightarrow Using Gunicorn, we can serve our applications with multiple threads.

```

Terminal: Local + 
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Loading personal and system profiles took 1644ms.
(base) PS D:\Edu\Flask> conda activate flask
(Flask) PS D:\Edu\Flask> pip install gunicorn
Collecting gunicorn
  Downloading gunicorn-20.1.0-py3-none-any.whl (79 kB)
    ━━━━ | 79 kB 2.6 MB/s
Requirement already satisfied: setuptools>=3.0 in c:\users\krishnaaleti\anaconda3\envs\flask
Installing collected packages: gunicorn
Successfully installed gunicorn-20.1.0
(Flask) PS D:\Edu\Flask>
  
```

\rightarrow Gunicorn uses our 'proc' file.

- 4, Now we need to freeze the requirements \Rightarrow we say that for this application, whatever we had installed till now is fine.

The code editor shows a requirements.txt file with the following content:

```

certifi==2020.6.20
click==8.1.2
colorama @ file:///tmp/build/80754af9/colorama
Flask==2.1.1
Flask-SQLAlchemy==2.5.1
greenlet==1.1.2
gunicorn==20.1.0
itsdangerous==2.1.2
Jinja2==3.1.1
MarkupSafe==2.1.1
PyMySQL==1.0.2
SQLAlchemy==1.4.35
Werkzeug==2.1.1
wincertstore==0.2
  
```

The terminal shows the output of pip freeze:

```

ERROR: Could not find a version that satisfies the requirement gunicorn (from: No matching distribution found for gunicorn)
(Flaskapp) PS D:\Edu\Flask_db_heroku\Flask> pip install gunicorn
Collecting gunicorn
  Using cached gunicorn-20.1.0-py3-none-any.whl (79 kB)
Requirement already satisfied: setuptools>=3.0 in c:\users\krishnaaleti\anaconda3\envs\flask
Installing collected packages: gunicorn
Successfully installed gunicorn-20.1.0
(Flaskapp) PS D:\Edu\Flask_db_heroku\Flask> pip freeze > requirements.txt
  
```

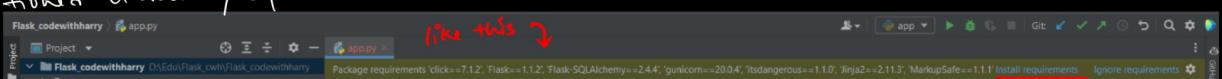
\Rightarrow pip freeze > requirements.txt

\rightarrow freezing & creating requirements.txt ✓

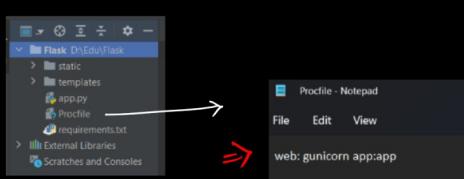
\Rightarrow Whoever wants to use this project, the requirements.txt will help in installing all the mentioned libraries/dependencies for them.

- 5, Create a Procfile
↳ used by Heroku to deploy our application.

like this ↴



All packages will be installed.



In the Procfile write \Rightarrow web: gunicorn app:app

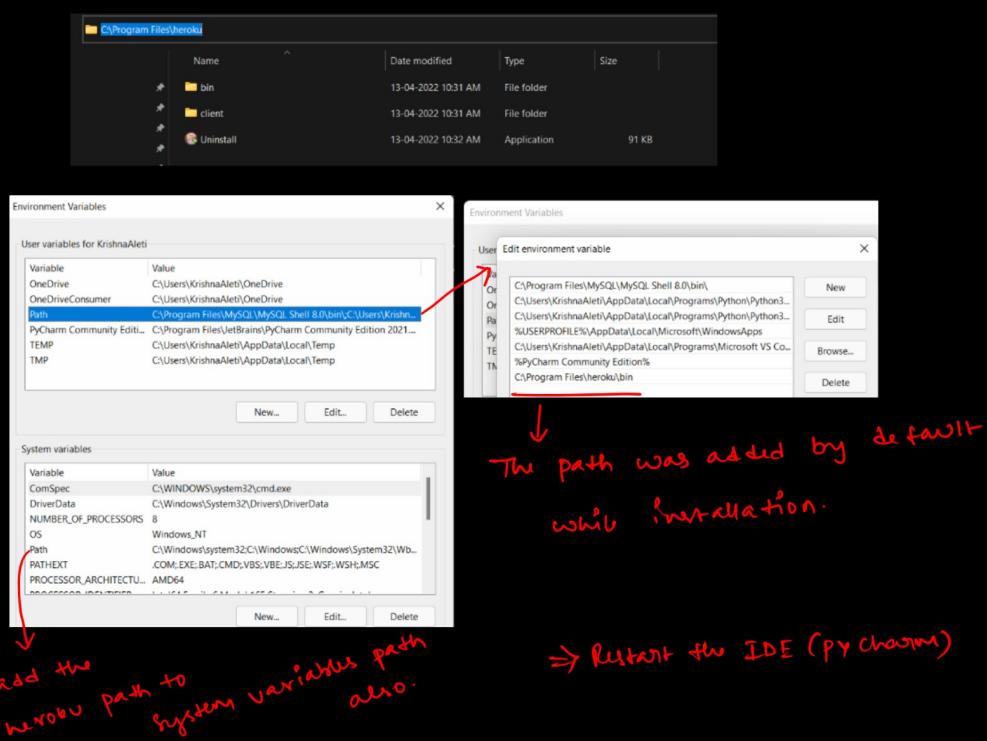
6,

```

Terminal: Local + 
(Flask) PS D:\Edu\Flask> heroku
heroku : The term 'heroku' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the name, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ heroku
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (heroku:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

(Flask) PS D:\Edu\Flask>
  
```

We get error because we had installed Heroku CLI but didn't add to the system environment path.



```
Terminal Local + v
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Loading personal and system profiles took 1992ms.
(base) PS D:\Edu\Flask> conda activate flask
(flask) PS D:\Edu\Flask> heroku
» Warning: heroku update available from 7.53.0 to 7.60.1.
CLI to interact with Heroku

VERSION
heroku/7.53.0 win32-x64 node-v12.21.0

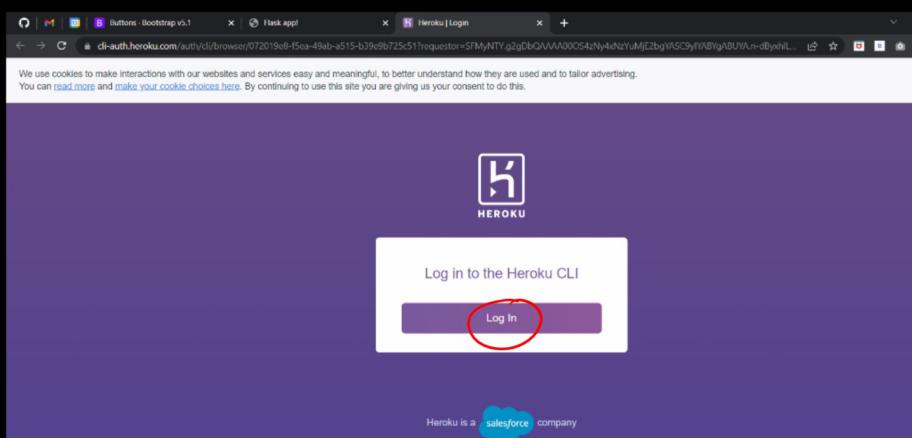
USAGE
$ heroku [COMMAND]

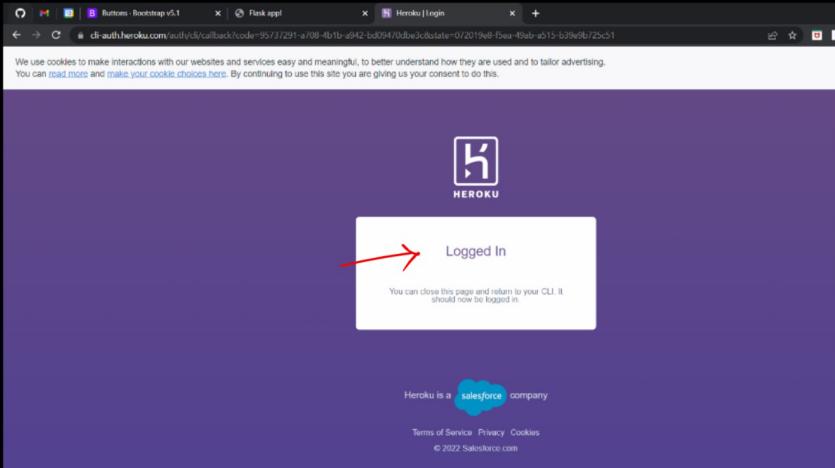
COMMANDS
access      manage user access to apps
addons     tools and services for developing, extending, and operating your app
apps       manage apps on Heroku
auth        check 2fa status
authorizations OAuth authorizations
autocomplete display autocomplete installation instructions
buildpacks  scripts used to compile apps
certs       a topic for the ssl plugin
ci          run an application test suite on Heroku
clients    OAuth clients on the platform
```

(Now it works)

? login to heroku

```
(flask) PS D:\Edu\Flask> heroku login ✓
» Warning: heroku update available from 7.53.0 to 7.60.1.
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.herokuapp.com/auth/cli/browser/e/2b19e8-f5ea-49ab-a515-b39e9b725c51?requestor=SFMyNTY.g2gDQAAAAB00054zNy4xNzYuMjE2bgYASC9yIVABYgABUYA.n-dByxhL8IzoLg5xLbfH\_8t9j1DeTnkoqQuvcx8
heroku: Waiting for login... |
```





⇒ logged-in

```
(flask) PS D:\Edu\Flask> heroku login
» Warning: heroku update available from 7.53.0 to 7.60.1.
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.herokuapp.com/auth/cli/browser/072019e8-f5ea-49ab-a515-b39e9b725c51?requestor=SFMyNTY.g2gDbQAAAAB0054zNy4xHzYuMjE2bgYASC9yIYABYgABUYA.n-dByxhL8Izot
g5xlbfM_8t9jDeTn4ogUvex8
Logging in... done
Logged in as krishnachandramouli5@gmail.com
(Flask) PS D:\Edu\Flask>
```

s, working with git
⇒ Download git from git-scm.com/downloads

```
(flask) PS D:\Edu\Flask> heroku login
» Warning: heroku update available from 7.53.0 to 7.60.1.
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.herokuapp.com/auth/cli/browser/072019e8-f5ea-49ab-a515-b39e9b725c51?requestor=SFMyNTY.g2gDbQAAAAB0054zNy4xHzYuMjE2bgYASC9yIYABYgABUYA.n-dByxhL8Izot
g5xlbfM_8t9jDeTn4ogUvex8
Logging in... done
Logged in as krishnachandramouli5@gmail.com
(Flask) PS D:\Edu\Flask> git init
Initialized empty Git repository in D:\Edu\Flask/.git/
(Flask) PS D:\Edu\Flask> git add .
warning: LF will be replaced by CRLF in .idea/inspectionProfiles/Project_Default.xml.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in .idea/inspectionProfiles/profiles_settings.xml.
The file will have its original line endings in your working directory
(Flask) PS D:\Edu\Flask> git commit -m "Initial commit"
```

```
Terminal Local + √
(flask) PS D:\Edu\Flask> git commit -m "Initial commit"
Author identity unknown
*** Please tell me who you are.

Run
git config --global user.email "you@example.com"
git config --global user.name "Your Name"
to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'KrishnaAleti@KrishnaAleti.(none)')
(Flask) PS D:\Edu\Flask> git commit -m "Initial commit"
Author identity unknown
*** Please tell me who you are.

Run
git config --global user.email "you@example.com"
git config --global user.name "Your Name"
to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'KrishnaAleti@KrishnaAleti.(none)')
```

fixing the issue by giving our github details

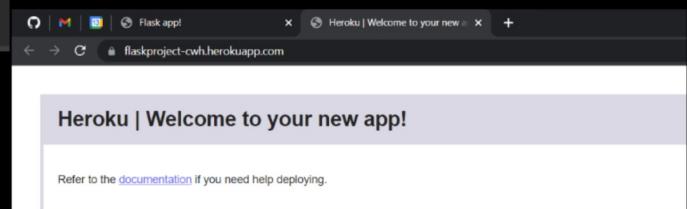
```
(flask) PS D:\Edu\Flask> git config --global user.email "you@example.com"
(Flask) PS D:\Edu\Flask> git config --global user.email "K_____5@gmail.com"
(Flask) PS D:\Edu\Flask> git config --global user.name "Your Name"
(Flask) PS D:\Edu\Flask> git config --global user.name "Krishna A."
(Flask) PS D:\Edu\Flask> git commit -m "Initial commit"
[master (root-commit) 2fcc108] Initial commit
12 files changed, 224 insertions(+)
create mode 108644 .idea/.gitignore
create mode 108644 .idea/Flask.iml
create mode 108644 .idea/inspectionProfiles/Project_Default.xml
create mode 108644 .idea/inspectionProfiles/profiles_settings.xml
create mode 108644 .idea/misc.xml
create mode 108644 .idea/modules.xml
create mode 108644 Procfile
create mode 108644 __pycache__/_app.cpython-36.pyc
create mode 108644 app.py
create mode 108644 requirements.txt
create mode 108644 static/krish.txt
create mode 108644 templates/index.html
(Flask) PS D:\Edu\Flask>
```

This is being fetched from heroku

9, creating app in heroku

⇒ heroku create appname

```
(flask) PS D:\Edu\Flask> heroku create flaskproject-cwh
» Warning: heroku update available from 7.53.0 to 7.60.1.
Creating ⬤ flaskproject-cwh... done
https://flaskproject-cwh.herokuapp.com/ | https://git.heroku.com/flaskproject-cwh.git
(Flask) PS D:\Edu\Flask>
```



```
(flask) PS D:\Edu\Flask> heroku create flaskproject-cwh
-> Warning: heroku update available from 7.55.0 to 7.60.1.
Creating ⬤ flaskproject-cwh... done
https://flaskproject-cwh.herokuapp.com/ | https://git.heroku.com/flaskproject-cwh.git
(flask) PS D:\Edu\Flask> git remote -v → git remote -v
heroku https://git.heroku.com/flaskproject-cwh.git (fetch)
heroku https://git.heroku.com/flaskproject-cwh.git (push)
(flask) PS D:\Edu\Flask>
```

fetch, push are done from heroku (not from git)
to, deploying to heroku

```
(flaskapp) PS D:\Edu\Flask_db\heroku\Flask> git push heroku master
Enumerating objects: 23, done.
Counting objects: 100% (23/23), done.
Delta compression using up to 8 threads
Compressing objects: 100% (17/17), done.
Writing objects: 100% (23/23), 5.69 KiB | 832.00 KiB/s, done.
Total 23 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Compressing source files... done.
remote: Building source...
remote:
remote: -----> Building on the Heroku-20 stack
remote: -----> Determining which buildpack to use for this app
remote: ----- Python app detected
remote: ----- No Python version was specified. Using the buildpack default: python-3.10.4
remote: To use a different version, see: https://devcenter.heroku.com/articles/python-runtimes
remote: ----- Installing python-3.10.4
remote: ----- Installing pip 21.3.1, setuptools 57.5.0 and wheel 0.37.0
remote: ----- Installing SQLite3
remote: ----- Installing requirements with pip
remote: Processing /tmp/build/80754af9/colorama_1607707115595/work
remote: ERROR: Could not install packages due to an ERROR: [Errno 2] No such file or directory: '/tmp/build/80754af9/colorama_1607707115595/mark'
remote: ! Push rejected, failed to compile Python app.
```

Requirement already satisfied: colorama in c:\users\krishnaaleti\anaconda3\envs\flaskapp\lib\site-packages (0.4.4)

⇒ In requirements.txt, Change Colorama==0.4.4

→ After changing ⇒ pip freeze
git add .
git commit -m "Comment"
heroku login
git push heroku master
(deploying Code to heroku)

```
(flaskapp) PS D:\Edu\Flask_db\heroku\Flask> git commit -m "comment"
[master b248f10] comment
 1 file changed, 0 insertions(+), 0 deletions(-)
(flaskapp) PS D:\Edu\Flask_db\heroku\Flask> heroku login
-> Warning: heroku update available from 7.55.0 to 7.60.1.
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/cli/browser/d89344e6-4177-4c0f-910b-15VKdmcUQ
Logging in... done
Logged in as Krishnachandranouli5@gmail.com
(flaskapp) PS D:\Edu\Flask_db\heroku\Flask> (flaskapp) PS D:\Edu\Flask_db\heroku\Flask> git push heroku master
Enumerating objects: 26, done.
Counting objects: 100% (26/26), done.
Delta compression using up to 8 threads
Compressing objects: 100% (26/26), done.
Total 26 (delta 5), reused 0 (delta 0), pack-reused 0
remote: -----> Compressing...
remote: Done: 64.2M
remote: -----> Launching...
remote: Released v3
remote: https://flaskapp05.herokuapp.com/ deployed to Heroku
remote: Verifying deploy... done.
-> To https://git.heroku.com/flaskapp05.git ✘ * [new branch] master -> master
(flaskapp) PS D:\Edu\Flask_db\heroku\Flask>
```

We can open this link &

see our app in heroku.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: powershell
remote: https://todo-codewithharry.herokuapp.com/ deployed to Heroku
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/todo-codewithharry.git
 d9bbeb5..a40a4db master -> master
PS D:\MyData\Business\code playground\Flask2021>
```

SNo	Title	Description	Time	Actions
1	Read about PC Parts	You have to read about pc parts	2021-03-09 06:59:44.235244	<button>Update</button> <button>Delete</button>

List of commands for deploying an app into Heroku:

heroku login

pip install gunicorn
pip freeze requirements.txt

Procfile

web: gunicorn app:app

git init
git add .
git commit -m "Initial commit"

heroku create todo-codewithharry

git remote -v

git push heroku master

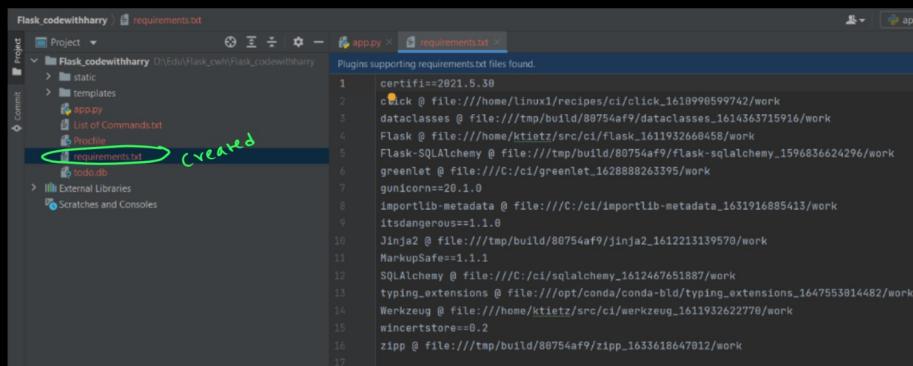
⇒ Deploying the other app (Code given by Code with Harry - Flask tutorial):

1, login to heroku ✓

2, pip install gunicorn

```
Terminal: Local + ~
Loading personal and system profiles took 1888ms.
(base) PS D:\Edu\Flask_cwh\Flask_codewithharry> conda activate flask_cwh
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry> pip install gunicorn ✓
Collecting gunicorn
  Using cached gunicorn-20.1.0-py3-none-any.whl (79 kB)
Requirement already satisfied: setuptools>=3.0 in c:\users\krishnaleti\anaconda3\envs\flask_cwh\lib\site-packages (from gunicorn) (58.0.4)
Installing collected packages: gunicorn
Successfully installed gunicorn-20.1.0
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry>
```

3, Pip freeze > requirements.txt



```
Project: Flask_codewithharry | requirements.txt
File | Project | Tools | Help | View | Editor | Run | Terminal | Local | + ~
Requirements.txt created
1 certifi==2021.5.30
2 click @ file:///home/linux1/recipes/ci/click_1610998599742/work
3 dataclasses @ file:///tmp/build/80754af9/dataclasses_1614363715916/work
4 Flask @ file:///home/kktietz/src/ci/Flask_1611932668458/work
5 Flask-SQLAlchemy @ file:///tmp/build/80754af9/flask-sqlalchemy_1596836624296/work
6 greenlet @ file:///C:/ci/greenlet_1628888263395/work
7 gunicorn==20.1.0
8 importlib-metadata @ file:///C:/ci/importlib-metadata_1631916885413/work
9 itsdangerous==1.1.0
10 Jinja2 @ file:///tmp/build/80754af9/jinja2_1612213139570/work
11 MarkupSafe==1.1.1
12 SQLAlchemy @ file:///C:/ci/sqlalchemy_1612467651887/work
13 typing_extensions @ file:///opt/conda/conda-bld/typing_extensions_1647553014482/work
14 Werkzeug @ file:///home/kktietz/src/ci/werkzeug_1611932622778/work
15 wincertstore==0.2
16 zipp @ file:///tmp/build/80754af9/zipp_1633618647012/work
17
```



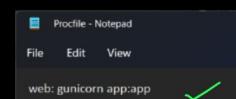
```
Terminal: Local + ~
Loading personal and system profiles took 1888ms.
(base) PS D:\Edu\Flask_cwh\Flask_codewithharry> conda activate flask_cwh
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry> pip install gunicorn
Collecting gunicorn
  Using cached gunicorn-20.1.0-py3-none-any.whl (79 kB)
Requirement already satisfied: setuptools>=3.0 in c:\users\krishnaleti\anaconda3\envs\flask_cwh\lib\site-packages (from gunicorn) (58.0.4)
Installing collected packages: gunicorn
Successfully installed gunicorn-20.1.0
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry> pip freeze > requirements.txt ✓
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry>
```

4, Create procfile → web: gunicorn app:app

5, git init

6, git add .

7, git commit -m "Initial Commit"



```

Terminal Local + v
Requirement already satisfied: setuptools>=3.0 in c:\users\krishnaaleti\anaconda3\envs\flask_cwh\lib\site-packages (from gunicorn) (58.6.0)
Installing collected packages: gunicorn
Successfully installed gunicorn-20.1.0
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry> pip freeze > requirements.txt
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry> git init
Initialized empty Git repository in D:/Edu/Flask_cwh/Flask_codewithharry/.git/
create mode 100644 .idea/.gitignore
create mode 100644 .idea/Flask_codewithharry.iml
create mode 100644 .idea/inspectionProfiles/Project_Default.xml
create mode 100644 .idea/inspectionProfiles/profiles_settings.xml
create mode 100644 .idea/misc.xml
create mode 100644 .idea/modules.xml
create mode 100644 .idea/vcs.xml
create mode 100644 List of Commands.txt
create mode 100644 Procfile
create mode 100644 app.py
create mode 100644 requirements.txt
create mode 100644 static/css/style.css
create mode 100644 static/harry.txt
create mode 100644 static/js/test.js
create mode 100644 static/pyvenv.cfg
create mode 100644 templates/base.html
create mode 100644 templates/index.html
create mode 100644 templates/update.html
create mode 100644 todo.db
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry>

```

8, heroku create todo-flaskdbapp

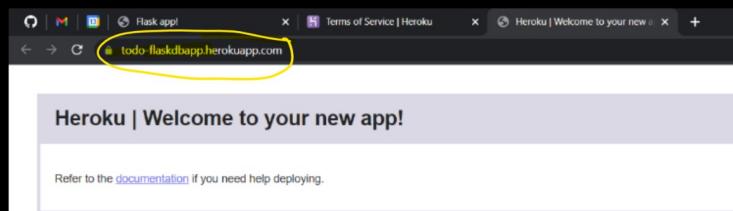
```

✓ (flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry> heroku create todo-flaskdbapp
  Warning: heroku update available from 7.53.0 to 7.60.1.
Creating ⬤ todo-flaskdbapp... ! ✨
!   Name must start with a letter, end with a letter or digit and can only contain lowercase letters, digits, and dashes.
Creating ⬤ todo-flaskdbapp... done
https://todo-flaskdbapp.herokuapp.com/ | https://git.heroku.com/todo-flaskdbapp.git
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry>

```

(app created)

app will be deployed to heroku with that name.



9, git remote -v

```

(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry> heroku create todo-flaskdbapp
  Warning: heroku update available from 7.53.0 to 7.60.1.
Creating ⬤ todo-flaskdbapp... done
https://todo-flaskdbapp.herokuapp.com/ | https://git.heroku.com/todo-flaskdbapp.git
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry> git remote -v
heroku https://git.heroku.com/todo-flaskdbapp.git (fetch)
heroku https://git.heroku.com/todo-flaskdbapp.git (push)
(Flask_cwh) PS D:\Edu\Flask_cwh\Flask_codewithharry>

```

10, git push heroku master

```

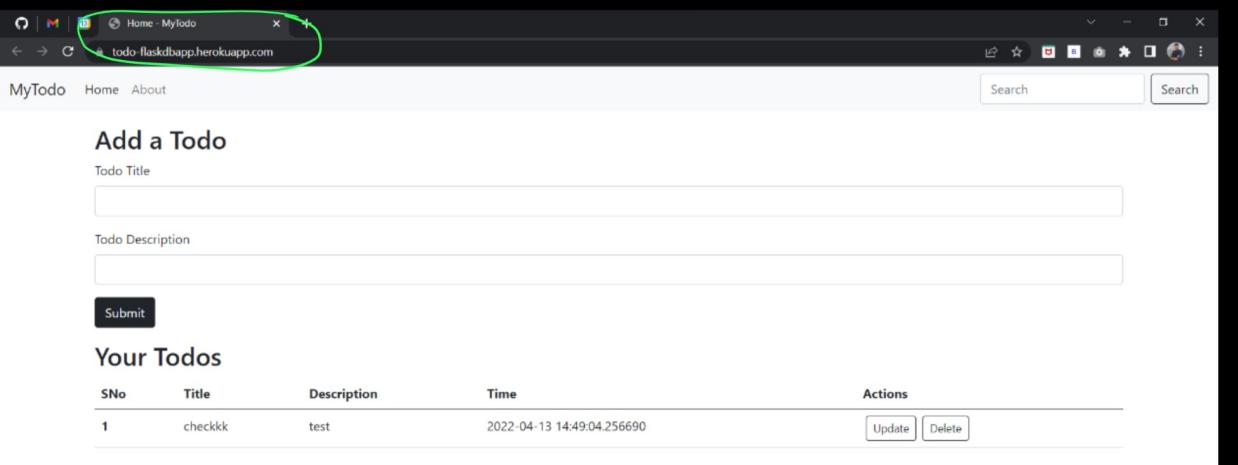
(Flaskdbheroku) PS D:\Edu\Flask_db_heroku\Flask_codewithharry> git push heroku master
Enumerating objects: 76, done.
Counting objects: 100% (76/76), done.
Delta compression using up to 8 threads
Compressing objects: 100% (69/69), done.
Writing objects: 100% (76/76), 11.6 KiB | 2.91 MiB/s, done.
Total 76 (delta 32), reused 0 (delta 0), pack-reused 0
remote: Compressing source files... done.
remote: Building source:
remote:
remote: -----> Building on the Heroku-20 stack
remote: -----> Determining which buildpack to use for this app
remote: -----> Python app detected
remote: No Python version was specified. Using the buildpack default: python-3.10.4
remote: To use a different version, see: https://devcenter.heroku.com/articles/python-runtimes
remote: -----> Installing python-3.10.4
remote: -----> Installing pip 21.3.1, setuptools 57.5.0 and wheel 0.37.0

```

```

Terminal Local + v
remote: Successfully installed Flask-1.1.2 Flask-SQLAlchemy-2.5.1 Jinja2-3.0.3 MarkupSafe-2.0.0 SQLAlchemy-1.4.35 Werkzeug-2.0.3 certifi-2021.5.30 click-8.0.4 dataclasses-0.6
greenlet-1.1.2 gunicorn-20.1.0 importlib-metadata-4.11.3 itsdangerous-1.1.0 typing-extensions-4.1.1 wincertstore-0.2 zipp-3.8.0
remote: -----> Discovering process types
remote: Procfile declares types > web
remote:
remote: -----> Compressing...
remote: Done: 64.3M
remote: -----> Launching...
remote: Released v3
remote: https://todo-flaskdbapp.herokuapp.com/ deployed to Heroku
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/todo-flaskdbapp.git
 * [new branch] master -> master
(Flaskdbheroku) PS D:\Edu\Flask_db_heroku\Flask_codewithharry>

```



⇒ Created a new app & deployed to heroku:
=====
(Code downloaded from 'Code with Harry' - Flask tutorial)

```
Terminal Local + v
Creating krish_todoapp... ! ← creating app
! Name must start with a letter, end with a letter or digit and can only contain lowercase letters, digits, and dashes.
(flaskdeploy) PS D:\Edu\Flask_db_heroku\Flask_cwh> heroku create krish-todo
> Warning: heroku update available from 7.53.0 to 7.60.1.
Creating krish-todo... done
https://krish-todo.herokuapp.com/ https://git.heroku.com/krish-todo.git
(flaskdeploy) PS D:\Edu\Flask_db_heroku\Flask_cwh> git remote -v
heroku https://git.heroku.com/krish-todo.git (fetch)
heroku https://git.heroku.com/krish-todo.git (push)
(flaskdeploy) PS D:\Edu\Flask_db_heroku\Flask_cwh>
```

```
(flaskdeploy) PS D:\Edu\Flask_db_heroku\Flask_cwh> git add .
(flaskdeploy) PS D:\Edu\Flask_db_heroku\Flask_cwh> git commit -m "comment"
[master 8144841] comment
1 file changed, 0 insertions(+), 0 deletions(-)
(flaskdeploy) PS D:\Edu\Flask_db_heroku\Flask_cwh> git remote -v
heroku https://git.heroku.com/krish-todo.git (fetch)
heroku https://git.heroku.com/krish-todo.git (push)
(flaskdeploy) PS D:\Edu\Flask_db_heroku\Flask_cwh> git push heroku master
Enumerating objects: 34, done.
Counting objects: 100% (34/34), done.
Delta compression using up to 8 threads
Compressing objects: 100% (27/27), done.
Writing objects: 100% (34/34), 6.59 KiB | 1.65 MiB/s, done.
Total 34 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Compressing source files... done.
remote: Building source:
remote:
remote: -----> Building on the Heroku-20 stack
remote: -----> Determining which buildpack to use for this app
remote: -----> Python app detected
remote: -----> No Python version was specified. Using the buildpack default: python-3.10.4
remote: To use a different version, see: https://devcenter.heroku.com/articles/python-runtimes
```

```
remote: Procfile declares types -> web
remote:
remote: -----> Compressing...
remote: Done: 6.5M
remote: -----> Launching...
remote: Released v3
remote: https://krish-todo.herokuapp.com/ deployed to Heroku →
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/krish-todo.git
 * [new branch] master -> master
(flaskdeploy) PS D:\Edu\Flask_db_heroku\Flask_cwh>
```

