

Create a Flask project

| | |
|--------------|--------------------------|
| Date | 20 September 2022 |
| Team ID | PNT2022TMID13865 |
| Project Name | News Tracker Application |

Flask Project:

The image shows a Windows 10 desktop environment. The primary focus is the Visual Studio Code (VS Code) editor, which is open to a Python file named `app.py`. The code is a Flask web application that uses a SQLite database (via `dao.py`) to manage user credentials and categories. The application is running on a local server at `http://127.0.0.1:5000`.

The VS Code interface includes an Explorer sidebar on the left showing the project structure, a central editor window with the `app.py` file, and a bottom panel with a Terminal window. The terminal displays the output of the `python app.py` command, showing the Flask server starting up and listening on port 5000. The taskbar at the bottom shows the Windows Start button, a search bar, and several open applications including VS Code, a file explorer, and a web browser.

The `app.py` code is as follows:

```
1 from flask import Flask, render_template, redirect, request, url_for
2 import threading
3 from hourlyscheduler import schedule
4 from dao import dbCon
5 from ibm_db import fetch_assoc, exec_immediate, prepare, bind_param, execute
6
7 app = Flask(__name__)
8 userID = None
9 total_categories = ["business", "entertainment", "environment", "food", "health", "politics", "science", "sports", "technology", "top", "world"]
10
11 @app.route("/", methods=['GET', 'POST'])
12 def home():
13     global userID
14     if userID is None:
15         getUserCat = "Select CATEGORY, EMAIL_PREF from user_credentials where ID = '"+userID+"'"
16         stmt = exec_immediate(dbCon, getUserCat)
17         dict = fetch_assoc(stmt)
18         user_categories = dict['CATEGORY'].split(",")
19         user_email_pref = dict['EMAIL_PREF']
20         if (request.method == 'POST'):
21             sel_categories = list(request.form.keys())
22             sel_categories.remove("email_pref")
23             updateUserCat = "Update user_credentials set CATEGORY=?, EMAIL_PREF=? where ID = ?"
24             prep_stmt = prepare(dbCon, updateUserCat)
25             bind_param(prepare_stmt, 1, ",".join(sel_categories))
26             bind_param(prepare_stmt, 2, request.form.get('email_pref'))
27             bind_param(prepare_stmt, 3, userID)
```