



Assignment 3: A Website for Advertisement

This assignment aims to help you practice databases, HTML, CSS, browser-side scripting with JavaScript, and server-side scripting with Python Flask¹. Your main task in this assignment is to develop a website for people who want to make an advertisement for different purposes, such as, selling their electronic devices, selling their clothes, etc.

You are expected to use HTML to define the content of web pages, CSS to design the web pages, JavaScript for browser-side scripting for web form validation purposes where appropriate, and Python Flask for server-side scripting. You are also expected to use an **SQLite** database to store and manage data.

Overview

In this assignment, you are expected to develop an advertisement website for the northern part of Cyprus to allow people to make their advertisements. People first need to register to create their advertisements. These advertisements will be publicly available which means that there is no need to log in to see and search for the active advertisements.

Functionalities

The website should have the following functionalities for people who want to create an advertisement:

- **Registration:**

The home page should have a link to access another page that includes a web form for people to register. They should be able to register with the following details: (1) username, (2) password, (3) full name, (4) email address and (5) telephone number. All these details are mandatory.

The username should be unique for each person, and the password should include at least one upper case letter, one lower case letter, one digit and one of these symbols [+, !, *, -] and its length should be at least ten.

If there is any problem, they will be directed to the registration page again and the problem will be shown to just the under the form. Otherwise, they will be directed to a page which shows the confirmation message and includes a link to go back to the home page.

- **Login:**

The home page should have a web form to allow people to log in with their usernames and passwords. If there is any problem, the problem will be shown to the user just under the form. Otherwise, the login form will be removed from the home page, and the following menu will be displayed:

[Home Page](#) | [My Advertisements](#) | [My Profile](#) | [Logout](#)

- **Logout:**

After login, the home page should also have a link for people to log out from the website. Once they log out, they will be directed to the home page again.

¹ <https://flask.palletsprojects.com/en/2.2.x/>

- **My Advertisements:**

After login, the home page should have a link for people to manage their advertisement. This page should include a web form where they can create a new advertisement with the following details: (1) title, (2) description, and (3) category (which should be selected from available categories listed in a combo box). All these details are mandatory.

The status of a new advertisement will be "active" by default. A unique identifier will be automatically assigned to each advertisement using AUTOINCREMENT. If there is any problem, the problem will be shown just the under the form.

This page will also list the previously posted advertisements of the person who has logged in and allows him/her to activate/deactivate them as follows:

Title	Description	Category	Activate/Deactivate
...	Deactivate
...	Activate

- **My Profile:**

This page displays the details of the profile in a web form and allows the user to edit them by changing the details and clicking the Edit button.

The website should also allow all people to search for the **active** advertisements as explained below. Activate advertisements are the ones which are not deactivated.

- **Search:**

The home page should include a text box with a button which allows searching for the active advertisement based on keywords. It will also include a combo box with all the available categories to select the category. There should be also an option to consider all categories.

The advertisement whose titles, descriptions or contact full name includes at least one of the keywords will be listed. Please note that it does not have to be a full match, so if the keyword is "abc", and the title is "xyabcz", then it should be listed as well.

If all categories are selected, then the matching advertisements should be categorised based on the categories as shown below, and "No advertisement" should be displayed for a category that the website does not find any matching active advertisement. Otherwise, the advertisements are also listed in the following format, but without any categorisation, and "No advertisement" will be displayed if no record is found.

Clothes:

Title	Description	Contact Full Name	
...	...		See More
...	...		See More

Technology:

Title	Description	Contact Full Name	
...	...		See More
...	...		See More

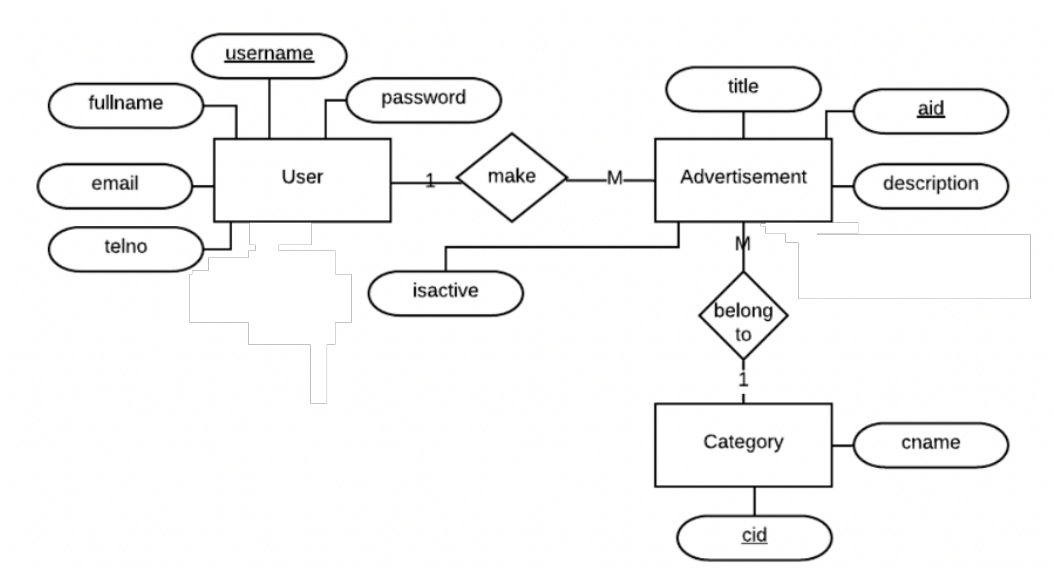
- **See Selected Advertisement:**

When people click a link called "See More", they will be directed to a page which shows all the details of the selected event as shown below. This page should also include a link to go back to the home page.

Title	...
Description	...
Category	...
Contact Full Name	...
Contact Email	...
Contact Telephone Number	...

Database

You need to create an **SQLite** database for this website based on the following entity-relationship diagram (ERD). You are expected to provide a separate Python script called **dbscript.py** to create necessary tables and insert necessary records into these tables.



You will also need to add the following details to the Category table. **Please note that if another category is added later, your website should be able to consider that category as well. So, you should not assume that you only have these categories.**

cid	cityname
1	Clothes
2	Technology
3	Cars
4	Food
5	Drink

Rules

- You need to write your program by using **Python 3.x**.
- You can **only** use all built-in functions and modules, apart from Flask.
- You need to create your CSS and JavaScript files as external files and include your HTML pages.
- **You are not allowed to use any design template.**
- You also need to create a file called **ReadMe.txt** which contains the following items. Please note that **if you do not submit ReadMe.txt, your submission will not be evaluated.**

- Team members
- Which version of Python 3.x you have used
- Which operating system you have used
- How you have worked as a team, especially how you have divided the tasks among the team members (who was responsible for what?), how you have communicated, how you have tested the program, etc.
- You need to put all your files (including your database file with the .db extension) into a folder that is named with your student id(s) and submit the compressed version of the folder in the **.zip** format.
- **Only one team member** should submit the assignment.
- **Code quality, modularity, efficiency, maintainability, and appropriate comments** will be part of the grading.

Grading Policy

The assignment will be graded as follows:

Grading Item	Mark (out of 100)
Database Creation Script	5
Login, Logout and Session Management	10
Register	5
Access user advertisements	10
Deactivate advertisements	10
Manage Profile	10
Search advertisements and list them categorised by category	20
See the details of the selected category	10
Navigation within the website	5
Form validation with JavaScript	10
Design with CSS	5