SIM UOW ISIT307

ASSIGNMENT 2

Last updated: 8 Feb 2024

Timothy Mah

txamah001@mymail.sim.edu.sg

8750634

Tutorial Group T01F

Contents

[Introduction 2](#_Toc158305403)

[Usage 2](#_Toc158305404)

[Structure 2](#_Toc158305405)

[File structure 2](#_Toc158305406)

[Flow 2](#_Toc158305407)

[Screenshots 3](#_Toc158305408)

[“Database” files 6](#_Toc158305409)

# Introduction

Hello this is my documentation for SIM UOW ISIT307 ASSIGNMENT 2.

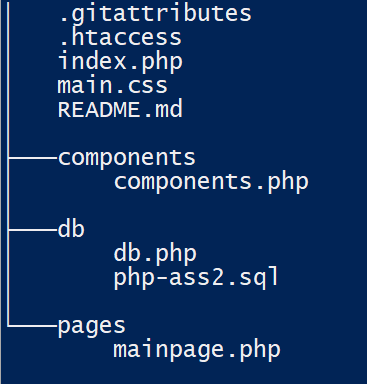
# Usage

Set up database

Access localhost in browser

# Structure

## File structure



* .htacess
  + This file allows me to configure the “starting point” of the website to be index.php
* Index.php
  + The “starting point” of the website. It’s the controller component of the MVC design that my website follows
* Main.css
  + All css code is stored in one file
* Components
  + Components.php
    - All functions that do not do processing but return html code
* Db
  + Db.php
    - Database connection is handled here
  + Php-ass2.sql
    - .sql file to create the database needed for my website
* Pages
  + Mainpage.php
    - The ‘main’ html page. Index.php will send components and all relevant data to mainpage.php

## Flow

The system design of my website is the same as in assignment 1. When the user enters the website, index.php will be the first file “called”. Then index.php will serve the components and relevant data for each route

All functions with the term “Component” in them only return HTML code. They do not perform any processing or logic. They are called multiple times in each route however I will omit their explanation.

For all search inputs, user can enter partial search term. If user enters a blank search input, all results will be shown.

Routes (in order of what’s written in my code)

/

This is the main route. User must be logged in to access. Else redirect to /login

If user is admin, the admin page features are displayed. Else if user is normal, normal user features are displayed

/users

This is the first feature of admin type account. User can view all user accounts and search for user account. Search can be partial. Search by name or ID.

There are 2 functions used. displayAllUsers() and searchForUser($searchUserName). These are similar functions but what’s different is the SQL statement used. Perhaps I could have merged the two functions. As their name suggests, displayAllUsers() uses a SELECT \* statement however searchForUser($searchUserName) uses a SELECT \* WHERE type statement for searching.

/ploc

Short for parking location. Very similar to the /users route however from this page, user can also add parking location.

Like in /users, there are 2 functions for searching and displaying parking locations. displayAllPlocs() and searchploc($searchploc).

The /insertparkinglocation route is “called” to add new parking location.

/userploc

This is the page where normal type user can search for available parking location. There are 2 functions used. displayPlocsNotFull() and plocnotfullsearch($searchLocation). They are similar functions that differ by SQL statements. The former uses a SELECT \* WHERE capacity > 0 to show all available parking location while the latter uses SELECT \* WHERE capacity > 0 AND xx LIKE $ searchLocation.

/plocfull

This is the page where admin type user can view and search for parking locations that are full. 2 functions are used. Plocfullsearch() and displayPlocsFull(). They are both similar functions but differ in SQL statement. The searching function uses more WHERE xx LIKE conditions to perform search

/plocnotfull

This is the page where admin type user can view and search for parking locations that are not full. 2 functions are used. plocnotfullsearch($searchLocation) and displayPlocsNotFull (). They are both similar functions but differ in SQL statement. The searching function uses more WHERE xx LIKE conditions to perform search

/insertparkingloc

This is the page that processes adding new parking location from the insert parking location form in /ploc. Once it has done processing, it redirects back to /ploc

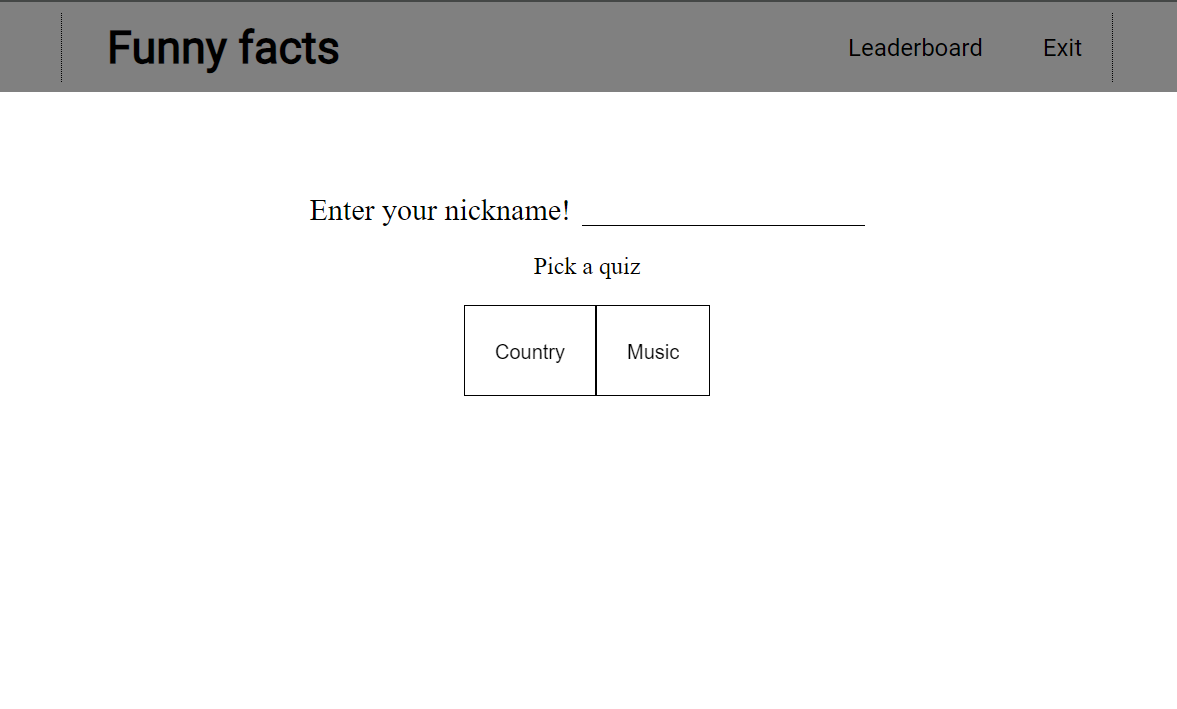
/check

This is the page where admin type user can view and search for checked-in parking locations. 2 functions are used. printCheckIn() and searchCheckedInLocation($searchName).

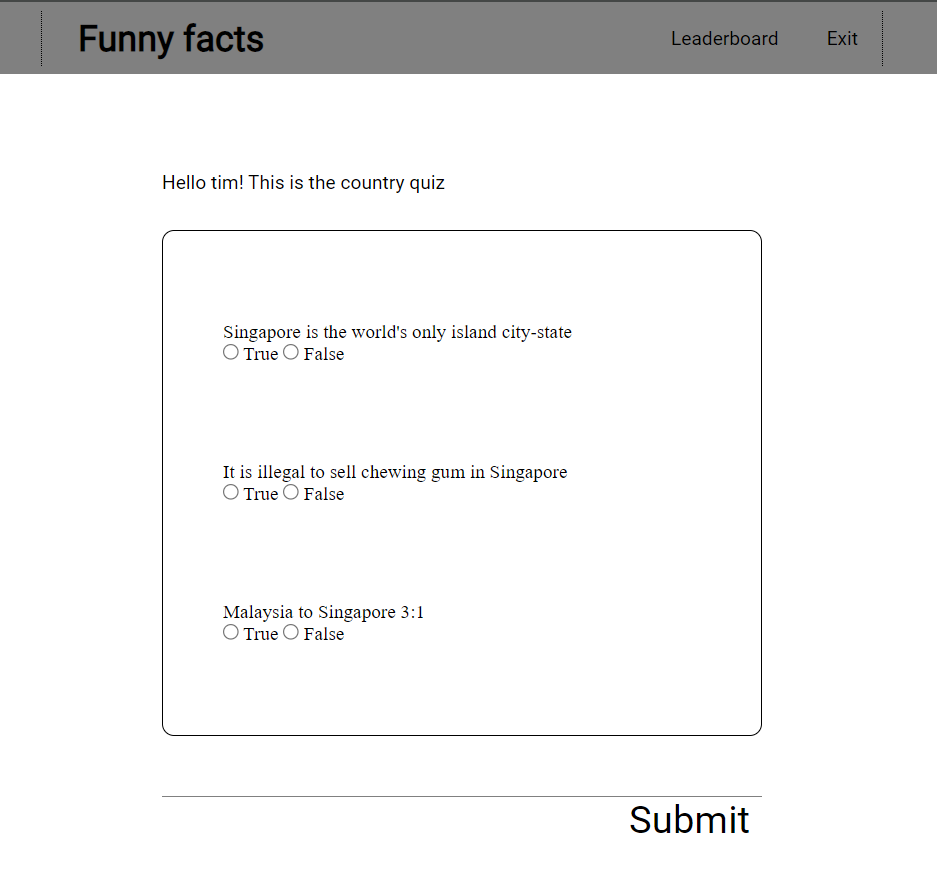
User can also check in and check out from this page. When checking out, the /checkout route is “called”.

## Screenshots

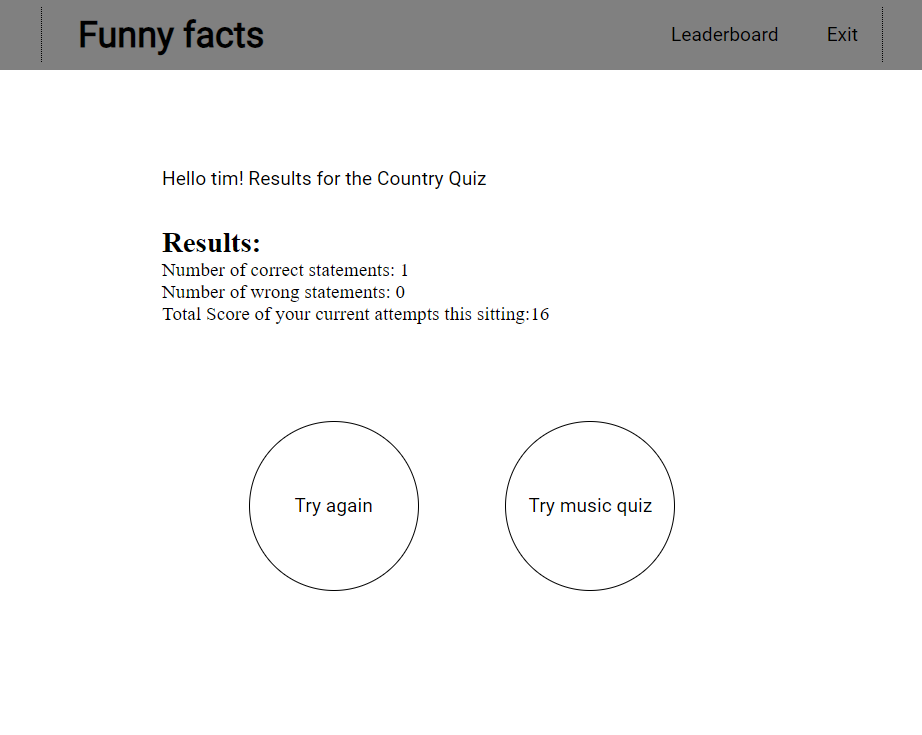
This is the first page the user sees. From here, the user may view the leaderboard. All other pages require nickname to be entered first.



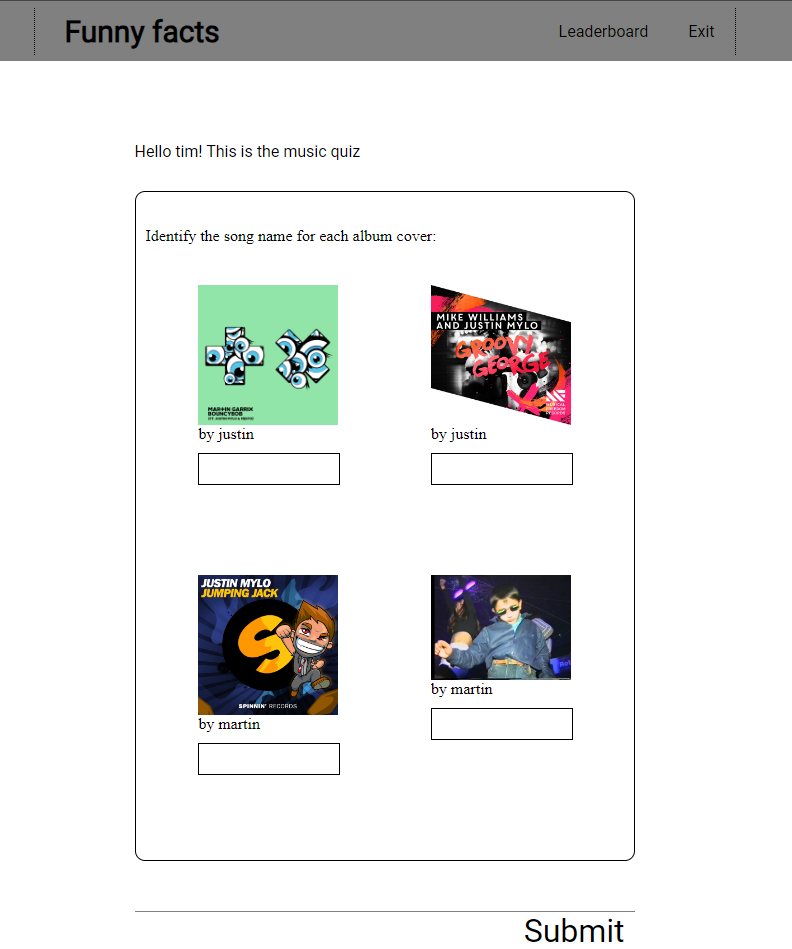
When user enters nickname and picks country, the country quiz is shown. Questions are at random as per assignment brief. From here, user can access the leaderboard page, exit page, or go back to the “home” page (the Funny facts logo) to enter different username or enter same username and attempt a different quiz



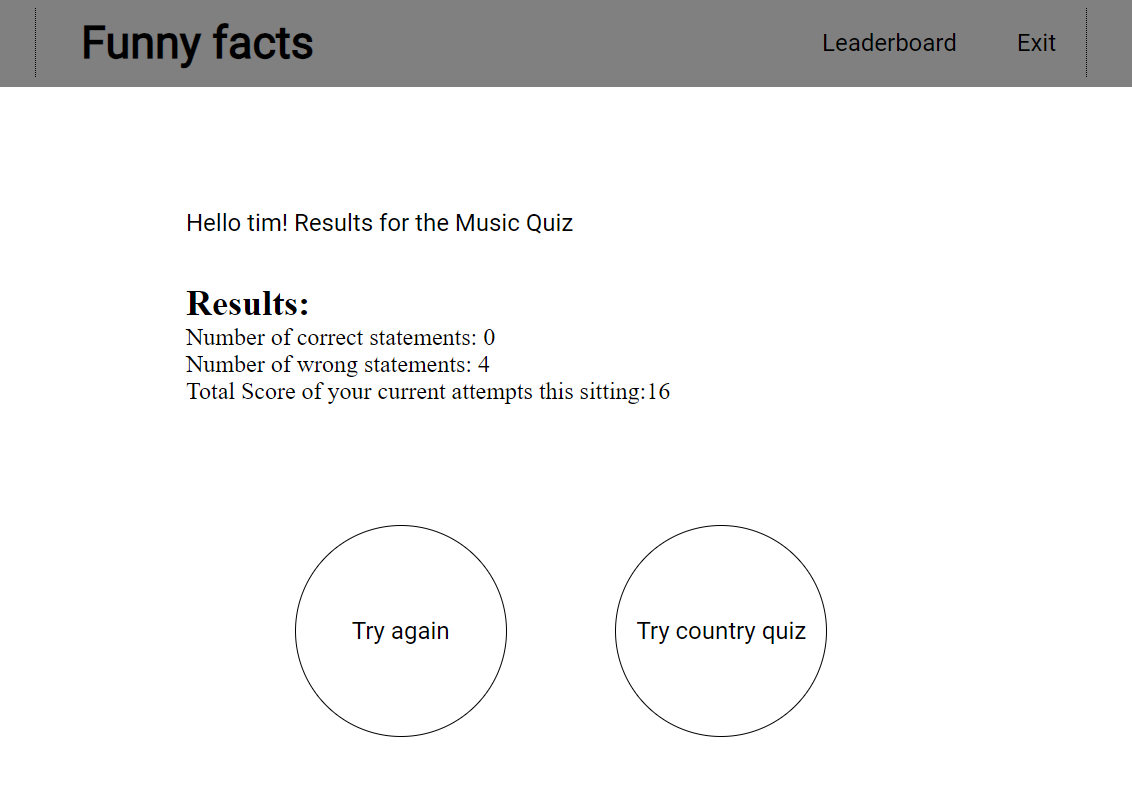
When user submits the country quiz, the results are shown. User can try the country quiz again or try the music quiz. From here, user can access the leaderboard page, exit page, or go back to the “home” page (the Funny facts logo) to enter different username or enter same username and attempt a different quiz.



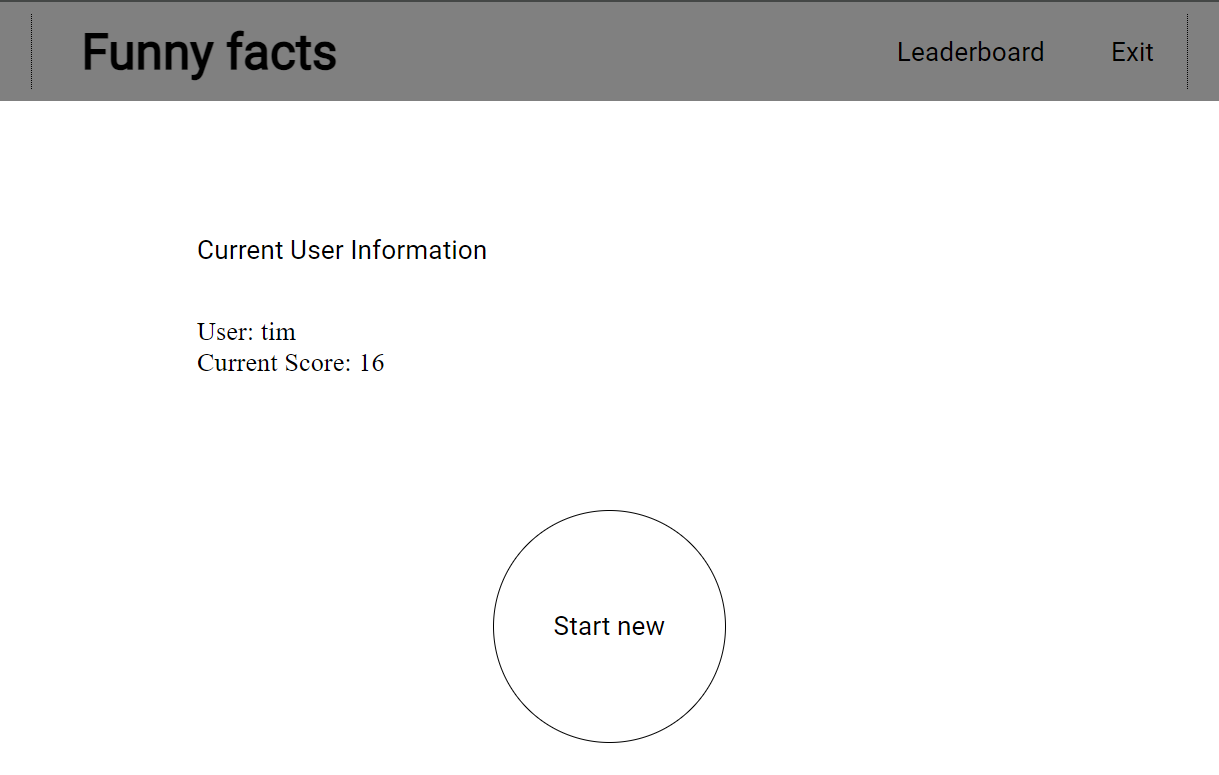
When user tries music quiz, the music quiz is shown. The album cover image followed by the composer data is shown. User may enter the album name. If left blank, the “question” is marked as wrong. From here, user can access the leaderboard page, exit page, or go back to the “home” page (the Funny facts logo) to enter different username or enter same username and attempt a different quiz.



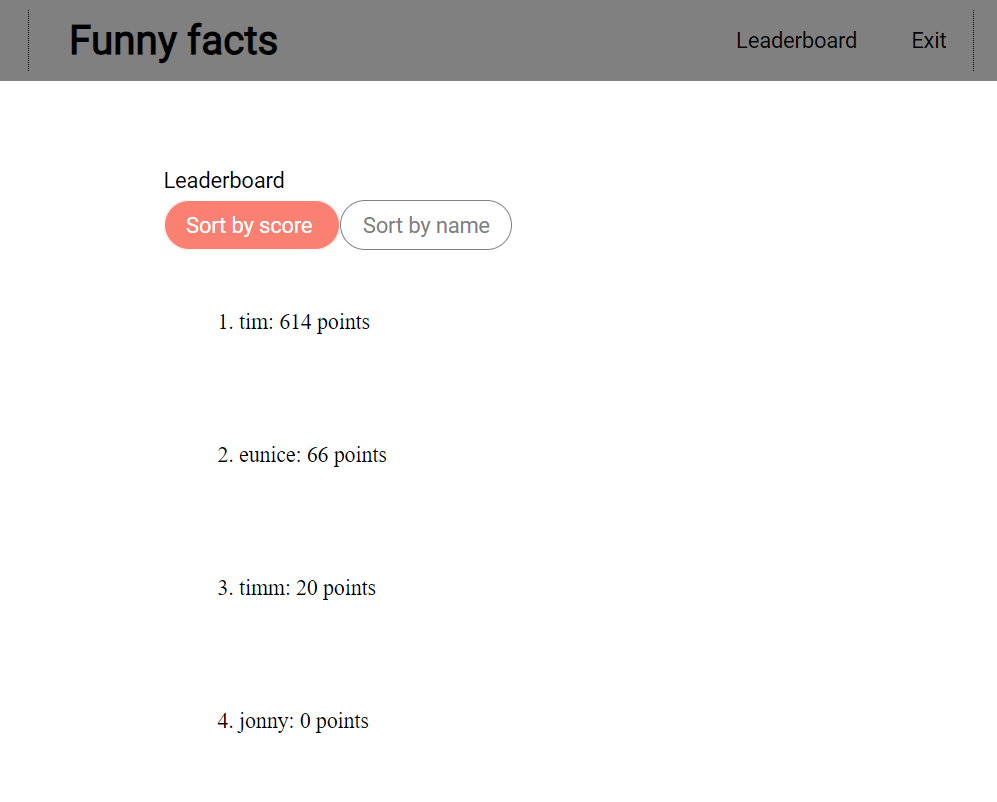
When user submits the music quiz, the results are shown. Similar to how the country quiz results are shown. From here, user can access the leaderboard page, exit page, or go back to the “home” page (the Funny facts logo) to enter different username or enter same username and attempt a different quiz.



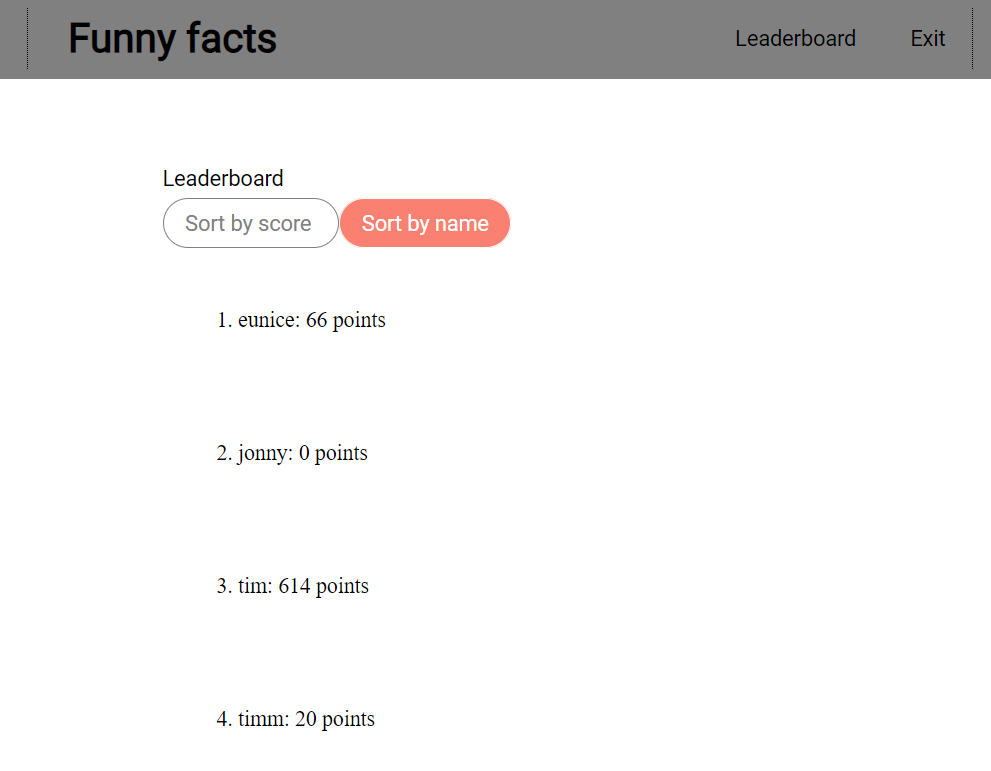
Should the user click on Exit, the exit page is shown. The user nickname and current total score of current sitting is shown. From here user can view leaderboard or start new quiz



From any page, user can view the leaderboard. By default it is sort by score. User can sort by name

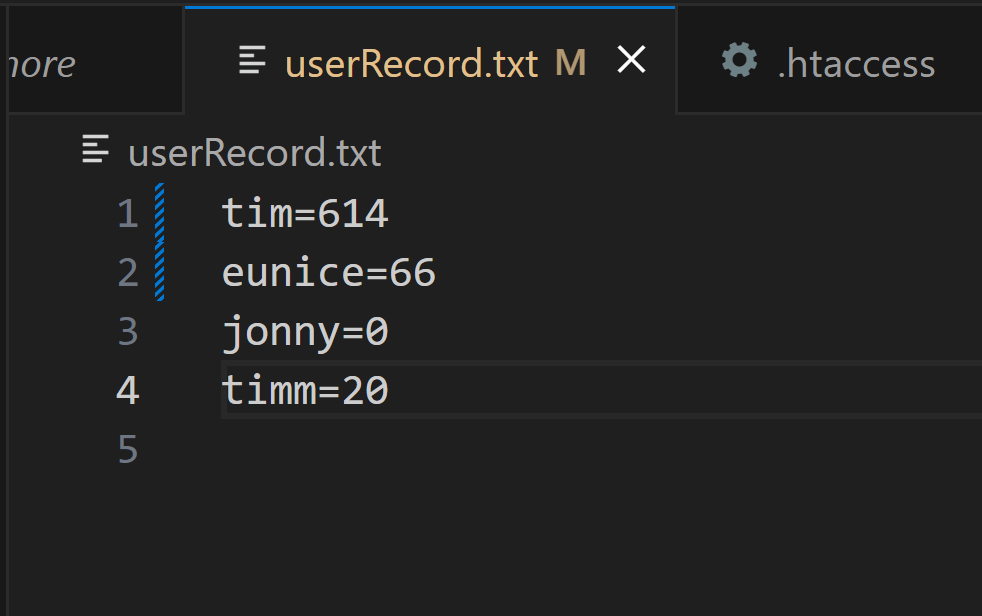
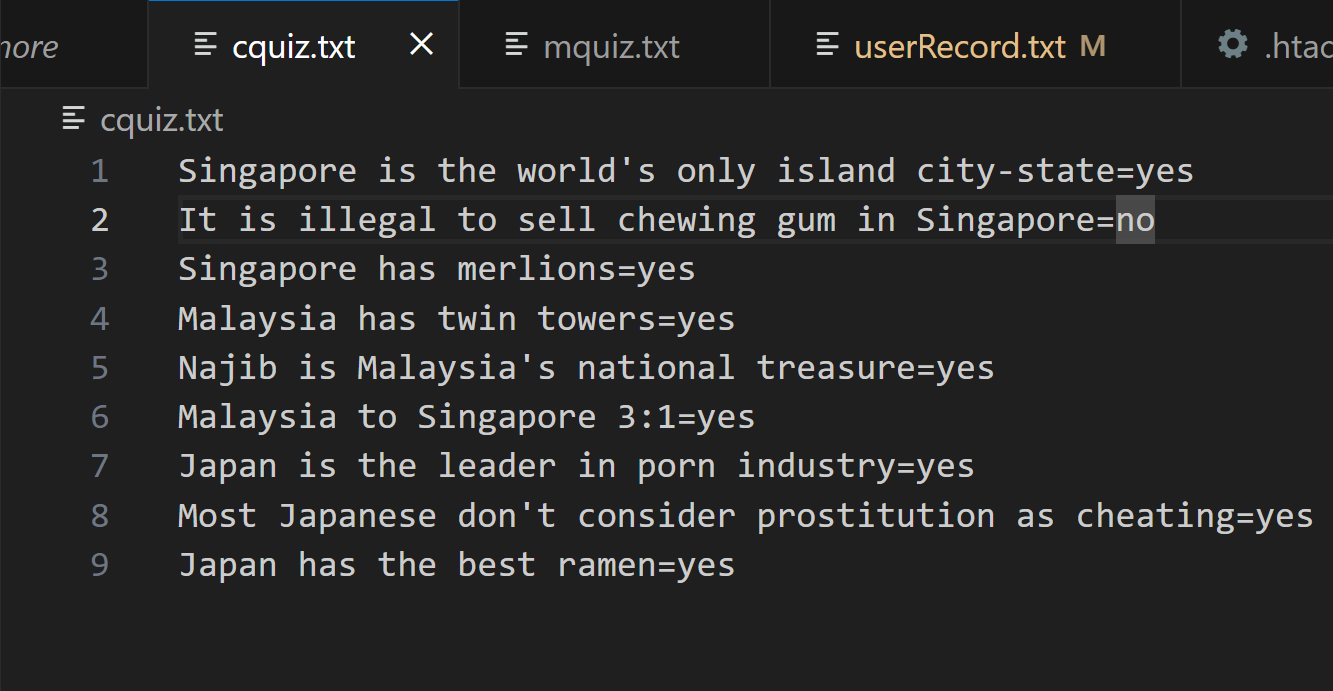


Sort by name



## “Database” files

There are three .txt files that act like database as per assignment requirements.

1. userRecord.txt. It holds the user nickname and their accumulated score. Separated by an “=”
   1. 
2. Cquiz.txt. It holds the country facts and true/ false. Separated by an “=”
   1. 
3. Mquiz.txt. It holds the music paths and the correct answer. Separated by an “=”
   1. 