**CS472-WAP Online Dictionary Project**

**Seid Ahmed – 613435**

**November 16, 2021**

**---------------**

**//dict.html**

<!DOCTYPE html>

<!-- Author by Seid Ahmed Adem 613435 -->

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <meta name="Author" content="Seid Ahmed">

  <title>WAP Online Dictionary</title>

  <link rel="shortcut icon" type="image/jpg" href="images/dict.png" />

  <link rel="stylesheet" href="css/dict.css">

  <script src="./js/jQuery.js"></script>

</head>

<body>

  <div id="container">

    <header>

      <div class="container">

        <img src="images/dict.png" alt="Dictionary Image">

        <h2>CS472-WAP Online Dictionary Project </h2>

      </div>

    </header>

    <div class="container">

      <div id="lookup">

        <form action="#">

          <label for="search">Term : </label>

          <input type="search" name="word" id="search" minlength="1" placeholder="Search for a word" required />

          <button type="submit">Lookup</button>

        </form>

      </div>

      <div id="definition">

        <ol></ol>

      </div>

    </div>

<div id="checkers">

      <a href="http://validator.w3.org/nu/?doc=https://seidahmed38.github.io/project/dict.html">

        <img src="images/w3c-html.png" alt="Valid HTML" /></a>

      <a href="http://validator.w3.org/nu/?doc=https://seidahmed38.github.io/project/dict.css">

        <img src="images/w3c-css.png" alt="Valid CSS" /></a>

      <a href="https://webster.cs.washington.edu/jslint/?referer">

        <img src="images/jshint-small.png" alt="Valid JS" /></a>

    </div>

    <script src="./js/dict.js"></script>

  </div>

</body>

</html>

**//dict.css**

#search {

    border-radius: 5px;

    font-size: 18px;

    padding: 4px 8px;

    width: 25%;

}

#container{

    width: 80%;

    margin: auto;

    text-align: center;

}

button, label{

    border-radius: 5px;

    font-size: 18px;

    padding: 4px 8px;

}

#definition{

    text-align: left;

    border: 2px solid pink;

    border-radius: 10px;

    margin-top: 10px;

    font-size: 18px;

}

img{

    width: 10%;

}

#checkers {

    position: fixed;

    bottom: 0;

    right: -20%;

    width: 80%;

}

**//dict.js**

$(document).ready(function () {

  $("form").on("submit", function (event) {

    let searchItem = $(this).serialize();

    $.ajax({

      url: "http://localhost:3000/dict",

      type: "POST",

      dataType: "json",

      data: searchItem,

      success: function (result) {

        $("#definition > ol").html("");

        for (let i = 0; i < result.length; i++) {

          let defn = $(

            "<li>(" +

            result[i].wordtype +

            ")::  " +

            result[i].definition +

            "</li>"

          );

          defn.appendTo($("#definition > ol"));

        }

        console.log(result);

      },

      error: function (err) {

        console.log(err);

      },

    });

    event.preventDefault();

  });

});

**//dict.js**

Standard jQuery library

**//dict.js**

const express = require("express");

const mysql = require('mysql');

const Dict = require("./models/word");

const conn = mysql.createConnection({

  host: "localhost",

  user: "root",

  password: "613435@Miu\*\*"

});

conn.connect(function (err) {

  if (err) throw err;

  console.log("Connected to MySQL Database!");

});

const app = express();

app.use(express.json());

app.use(express.urlencoded({ extended: true }));

app.use(express.static("./client"));

// Home route

app.get("/", (req, res) => {

  res.redirect("dict.html");

});

app.post("/dict", (req, res) => {

  Dict.findAll({

    where: {

      word: req.body.word,

    },

    attributes: ["word", "wordtype", "definition"],

  })

    .then((dict) => {

      res.send(dict);

    })

    .catch((err) => {

      console.log(err);

    });

});

const port = process.env.port || 3000;

app.listen(port, () => {

  console.log(`Web server listening on port ${port} `);

});

**//word.js**

const Sequelize = require("sequelize");

const db = new Sequelize("entries", "root", "613435@Miu\*\*", {

  host: "localhost",

  dialect: "mysql",

});

const EnglishDict = db.define("entrie", {

  word: {

    type: Sequelize.STRING,

  },

  wordtype: {

    type: Sequelize.STRING,

  },

  definition: {

    type: Sequelize.STRING,

  },

});

module.exports = EnglishDict;

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |