'use strict';

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* part.1 - function titleClickHandler \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**function titleClickHandler(event){**

  event.preventDefault();

  const clickedElement = this;

  /\* remove class 'active' from all article links \*/

  const activeLinks = document.querySelectorAll('.titles a.active');

  for(let activeLink of activeLinks) {

    activeLink.classList.remove('active');  }

  /\* add class 'active' to the clicked link \*/

  clickedElement.classList.add('active');

  /\* remove class 'active' from all articles \*/

  const activeArticles = document.querySelectorAll('article.active');

  for (let activeArticle of activeArticles) {

    activeArticle.classList.remove('active');  }

  /\* get 'href' attribute from the clicked link \*/

  const articleSelector = clickedElement.getAttribute('href');

  /\* find the correct article using the selector (value of 'href' attribute) \*/

  const targetArticle = document.querySelector(articleSelector);

  /\* add class 'active' to the correct article \*/

  targetArticle.classList.add('active');

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* part.2 - function generateTitleLinks \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

const optArticleSelector = '.post',

  optTitleSelector = '.post-title',

  optTitleListSelector = '.titles',

  optArticleTagsSelector = '.post-tags .list';

**function generateTitleLinks(customSelector = ''){**

  /\* remove contents of titleList \*/

  const titleList = document.querySelector(optTitleListSelector);

  titleList.innerHTML = '';

  /\* for each article \*/

  const articles = document.querySelectorAll(optArticleSelector + customSelector);

  let html = '';

  for (let article of articles) {

    /\* get the article id \*/

    const articleId = article.getAttribute('id');

    /\* find the title element \*/

    /\* get the title from the title element \*/

    const articleTitle = article.querySelector(optTitleSelector).innerHTML;

    /\* create HTML of the link \*/

    const linkHTML = '<li><a href="#' + articleId + '"><span>' + articleTitle + '</span></a></li>';

    /\* insert link into html variable \*/

    html = html + linkHTML;

  }

  titleList.innerHTML = html;

  const links = document.querySelectorAll('.titles a');

  for(let link of links){

    link.addEventListener('click', titleClickHandler);

  }

}

generateTitleLinks();

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* part.3 - function generateTags \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**function generateTags(){**

  /\* find all articles \*/

  const articles = document.querySelectorAll(optArticleSelector);

  /\* START LOOP: for every article: \*/

  for (let article of articles) {

    /\* find tags wrapper \*/

    const tagsWrapper = article.querySelector(optArticleTagsSelector);

    /\* make html variable with empty string \*/

    let html = '';

    /\* get tags from data-tags attribute \*/

    const articleTags = article.getAttribute('data-tags');

    /\* split tags into array \*/

    const articleTagsArray = articleTags.split(' ');

    /\* START LOOP: for each tag \*/

    for(let tag of articleTagsArray){

      /\* generate HTML of the link \*/

      const linkHTML = '<li><a href="#tag-' + tag + '">' + tag + '</a></li>';

      /\* add generated code to html variable \*/

      html = html + '  ' + linkHTML;

    /\* END LOOP: for each tag \*/     }

/\* insert HTML of all the links into the tags wrapper \*/

    tagsWrapper.innerHTML = html;

  /\* END LOOP: for every article: \*/

  }

}

generateTags();

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* part.4 - function tagClickHandler \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**function tagClickHandler(event){**

  /\* prevent default action for this event \*/

  event.preventDefault();

  /\* make new constant named "clickedElement" and give it the value of "this" \*/

  const clickedElement = this;

  /\* make a new constant "href" and read the attribute "href" of the clicked element \*/

  const href = clickedElement.getAttribute('href');

  /\* make a new constant "tag" and extract tag from the "href" constant \*/

  const tag = href.replace('#tag-', '');

  /\* find all tag links with class active \*/

  const activeTags = document.querySelectorAll('a.active[href^="#tag-"]');

  /\* START LOOP: for each active tag link \*/

  for (let activeTag of activeTags) {

    /\* remove class active \*/

    activeTag.classList.remove('active');

  /\* END LOOP: for each active tag link \*/   }

  /\* find all tag links with "href" attribute equal to the "href" constant \*/

  const tagLinks = document.querySelectorAll('a[href="' + href + '"]');

  /\* START LOOP: for each found tag link \*/

  for (let tagLink of tagLinks) {

    /\* add class active \*/

    tagLink.classList.add('active');

  /\* END LOOP: for each found tag link \*/   }

  /\* execute function "generateTitleLinks" with article selector as argument \*/

  generateTitleLinks('[data-tags~="' + tag + '"]');

}

**function addClickListenersToTags(){**

  /\* find all links to tags \*/

  const allTagLinks = document.querySelectorAll('a[href^="#tag-"]');

  /\* START LOOP: for each link \*/

  for (let allTagLink of allTagLinks) {

    /\* add tagClickHandler as event listener for that link \*/

    allTagLink.addEventListener('click', tagClickHandler);

/\* END LOOP: for each link \*/

  }

}

addClickListenersToTags();

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* part.5 - function generateAuthors \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/