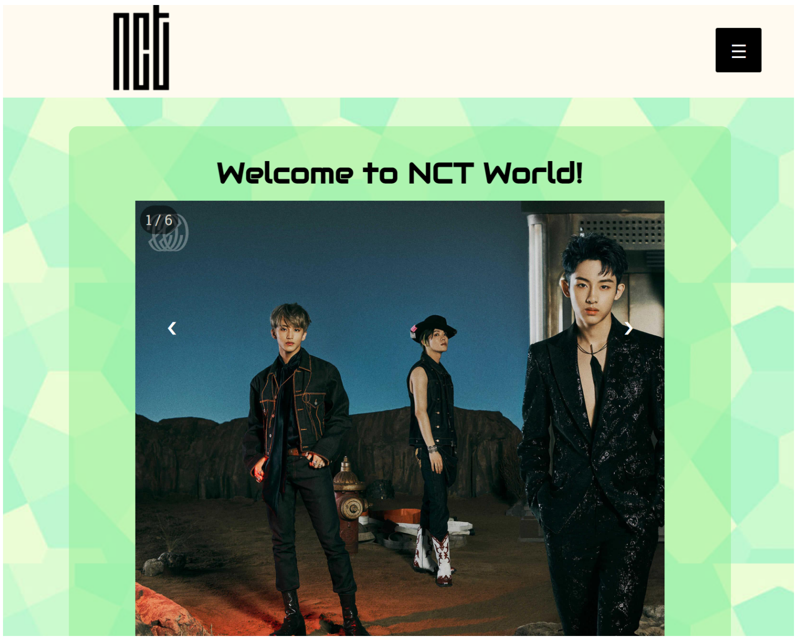
Ge Gao
Comp 266
3527100

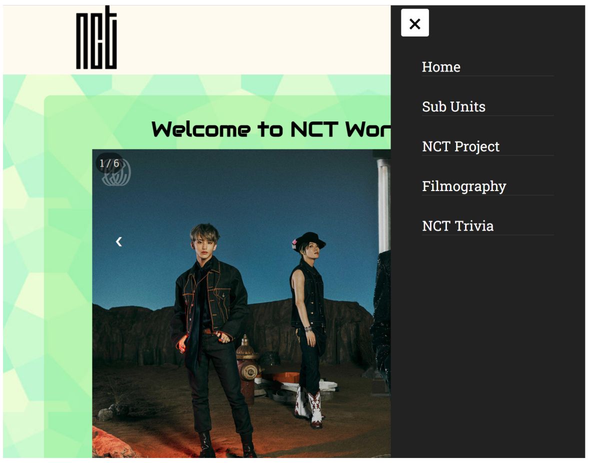
Unit 5 Part 2

Design 1: Responsive navigation bar
Users who have smaller screens will be able to use responsive side navigation bar. Firstly, I added CSS styles for side menu. There is a button design for opening menu and a button for closing menu. New CSS styles for screen smaller than a specific width were added. Using javascript will enable opening and closing menu with user clicks.I set side navigation bar section invisible for default, click event will toggle between opening menu style and invisible style.

Before click CSS Style:



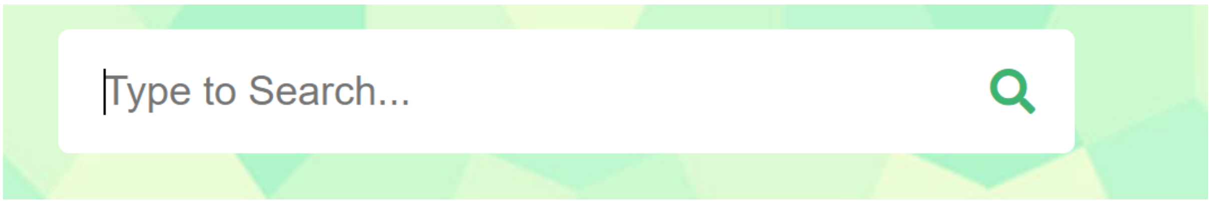
After click CSS Style:



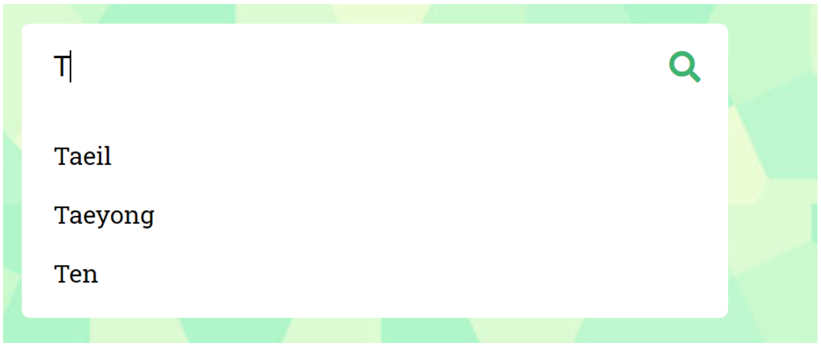
JS:
//set constant variable
const openNavMenu= select open button,
closeNavMenu=select close button,
navMenu= select navigation bar part which including all the links
//Method
When the user clicks open button, a function that opens menu will run
When the user clicks close button, a function that closes menu will run
//function
toggleNav(): navigation bar toggle between class of opening menu and removing that class (toggle between open and close menu)

Design 2: Search bar with auto-complete suggestion
Users who want to search more information about specific member can use search bar. When users try to type a name, the search bar will provide auto-complete suggestions. My webpage may inspire users to look for some other information that my website doesn't include. At this time, the click button will direct users to google with the result that user searched. In the html section, I added a blank hyperlink in search bar.
Javascript is used to filter suggestions and direct users to google result page with click.

Before user typing:



After user typed, the auto-complete suggestion box showed up:

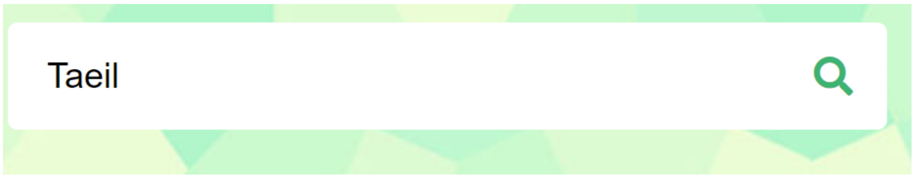

T

Taeil

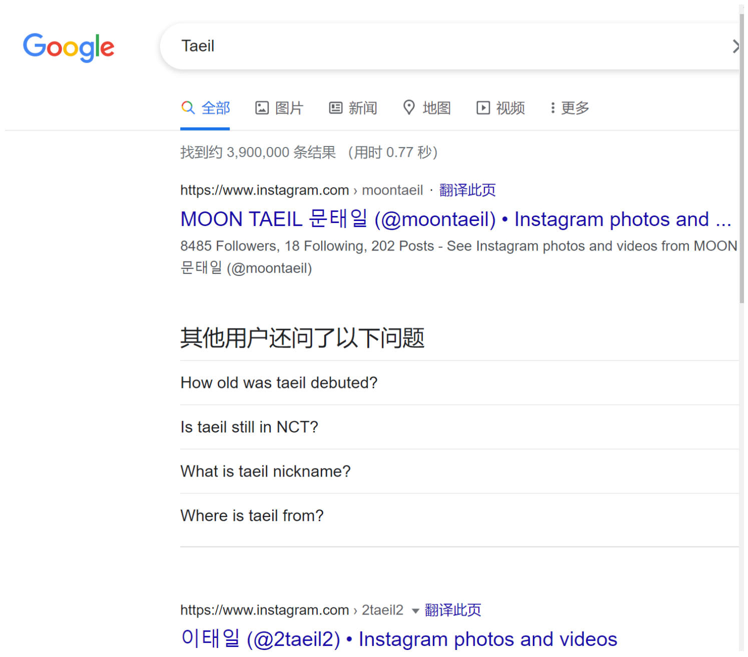
Taeyong

Ten

After user select a suggestion:



After click search icon, it will direct to google:


Taeil

找到约 3,900,000 条结果 （用时 0.77 秒）

https://www.instagram.com › moontaeil · 翻译此页

MOON TAEIL 문태일 (@moontaeil) • Instagram photos and ...

8485 Followers, 18 Following, 202 Posts - See Instagram photos and videos from MOON 문태일 (@moontaeil)

其他用户还问了以下问题

How old was taeil debuted?

Is taeil still in NCT?

What is taeil nickname?

Where is taeil from?

https://www.instagram.com › 2taeil2 ▾ 翻译此页

이태일 (@2taeil2) • Instagram photos and videos

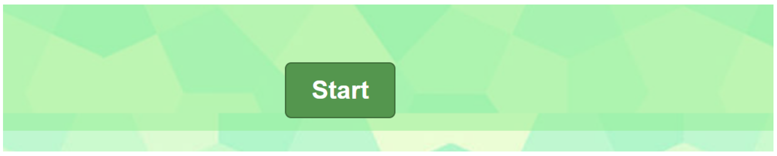
JS:

```
//set constant variables to get required elements
const searchWrapper=get the whole section of search bar
const inputBox=get the typing section of search bar
const suggBox=get the autocomplete section of search bar
const icon=get the icon section of search bar
//variable names
let linkTag=return the <a> tag of search bar
let webLink
let suggestions=[names of NCT members]

//define event, arrow function set up
//if user press any key and release
constant inputBox.onkeyup=(e)=>{//e is parameter
variable userData=user entered data
set an empty array called emptyArray

//if statements
if user typed {
//define event, arrow function setup
click search icon=()=>{
variable weblink=google search address + user entered data;
add href attribute with the value of weblink to linkTag;
simulate a mouse-click to linkTag, like searching in google
}
//set emptyArray array, define event, arrow function setup
emptyArray=suggestions.filter((data)=>{ //data is parameter, filter suggestions array value and user characters to lowercase and return only those words which are start with user enetered chars
})
emptyArray = emptyArray.map((data)=>{
passing return data inside li tag
})
//show autocomplete box, add CSS class
searchWrapper.classList.add(“active”);
function showSuggestions(emptyArray); //show auto-complete suggestion array
variable allList=select all li of suggBox
//for loop
for (i=0;i<list length of allList;i++)
{add onclick attribute in all li tag}
else{
remove active CSS style //hide autocomplete box}}

//function setup
select(element){//select function is for allList adding attribute
variable selectData=element.textContent;
inputBox value=selectData //make the value user typed = selectData
click icon =()=>{
weblink=google search address + user entered data;
add href attribute with the value of weblink to linkTag;
simulate a mouse-click to linkTag, like searching in google;
}
After click, remove auto-complete suggestion section}

function showSuggestions(list){
variable name listData;
if(list.length not 0){
variable userValue=inputBox.value;
listData = '<li>'+ userValue +'</li>';//make user value be a list
}
else{listData stays same}
Inner HTML content of suggBox = listData;
}
```

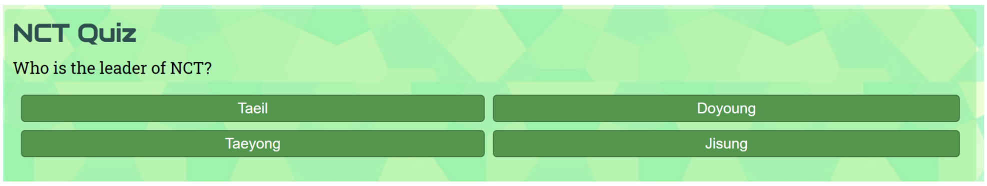
Design 3: Quiz

The quiz section added to NCT Trivia page is to make the user experience more interesting. The JS is used to validate buttons to choose an answer and make it move to the next question. When user clicks answer, the background color of each answer button will change to show correct and wrong.

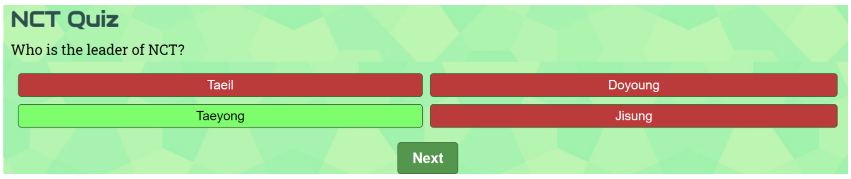
The default look for quiz:



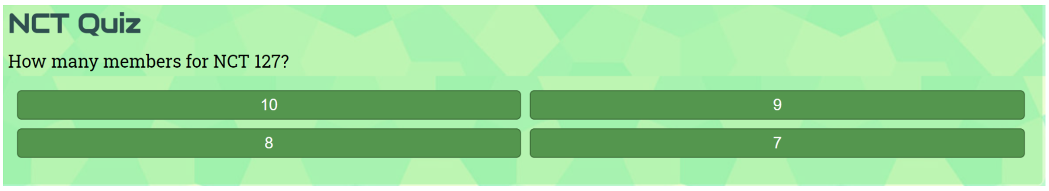
User clicks start, question and choices will be shown:



User select answer, background color of button will change



User click next, move to next question:



JS:

```
//set constant variables to get required elements
const startButton
const nextButton
const questionContainerElement= get the whole question section set which include
question and answer part
const questionElement=get question part
const answerButtonsElement=get choice buttons

const questions=[{question: ‘what’s the question?’,
answers:[{text: ‘’,correct:ture},{text: ‘’,correct:false}]

//variable names
let shuffledQuestion,currentQuestionIndex;

//define event
When click startButton, function startGame() is called;
which click nextButton, {
currentQuestionIndex++;//move to next question
```

```
setNextQuestion() is called}

//function setup
startGame(){
add class ‘hide’ CSS style to startButton;//start button doesn’t show
sheffledQuestions=sort questions array randomly;
currentQuestionIndex=0;
remove class ‘hide’ CSS style to questionContainerElement;
call function setNextQuestion();
}

setNextQuestion(){
call function resetState();
call function showQuestion(shuffledQuestions[currentQuestionIndex]);//parameter is questions array element
}

showQuestion(question){//parameter is question in questions array
questionElement.text content =questions.question;
call function for all answers{
const button=create a button element ;
insert text to button= answer.text;
append class ‘btn’ to button;
if(answer is correct){
add data attribute ‘correct’  to button element=answer.correct;}
add event listener to button, when it’s clicked, call function selectAnswer();
append button to answerButtonsElement;
}

resetState(){
add class ‘hide’ CSS style to nextButton;//next button doesn’t show
loop all children of answerButtonsElement to remove child;//so the original default answer button will not shown in choice list
}

selectAnswer(e){
//set constant variable
const selectedButton=e.target;//get which button user selected
const correct=selectedButton.dataset.correct //check if the selected button is correct
call function setStatusClass(document.body, correct);
loop all answer button=>{
setStatus(button,button.dataset.correct)}
}
```

```
setStatusClass(element, correct){
call function clearStatusClass(element);
if（correct）{add class ‘correct’ class to element}
else{add class ‘wrong’ class to element}
}

clearStatusClass(element){
reset all the status by removing ‘correct’ and ‘wrong’ class
}
```

Reference:
https://www.youtube.com/watch?v=vb_u7mj84Tc
https://www.w3schools.com/howto/tryit.asp?filename=tryhow_js_collapse_sidepanel
https://www.youtube.com/watch?v=cRrEGkNJ1A8
https://www.youtube.com/watch?v=riDzcEQbX6k