TEST A - basic level

Goal:

Create a basic web-interface for Web Search API.

Tools:

Web Search API, News API, Image API, AutoSuggest API.

https://rapidapi.com/contextualwebsearch/api/web-search

Plan:

1. Get familiar with the API, make some tests on rapidapi.com

https://share.getcloudapp.com/2Nujxpb9

2. Use code snippets for JavaScript as a start. https://share.getcloudapp.com/L1uQ48xw
3. Create simple html page with an input for query and a button (like Google search).
4. Add simple JavaScript logic to handle click on button - send request to API and show results on the page without reloading.

TEST B - medium level

Goal:

Create a basic web-page to generate reminders or countdowns for some events.

Tools:

Html, JavaScript, Node.js

Plan:

1. Create simple UI using standard HTML5 elements https://share.getcloudapp.com/v1uxlB0x
2. Make a list of created reminders. Add dynamic countdown to visualize how much time left for each item
3. Show alert when time for reminder reached

Additional functionality

- Show previously created reminders after page reloading. (Use LocalStorage, Node.js server, etc.)
- Add UI to remove reminders.
- Use Web Speech API to alert user about event occured

TEST C - medium/high level

Goal:

Create a simple game using canvas element.
(Something like classical Snake)

Tools:

Html, JavaScript

Plan (example):

1. Move some figure using mouse or keyboard
2. Create random figures that can interact with player object (it can be obstacle, enemy, or food);
3. Add score calculation, score board, player name

TEST D - high level

Goal:

Create simple Node.js server for resizing or crop images (Interact over simple html page or using Postman)

Tools:

Html, JavaScript, Node.js,

Plan:

1. Create simple http server using Node.js or use some framework (express, sails, nest.js, meteor, etc)
2. Create a route to accept image file and parameters for resizing or cropping image
3. return image in response from server.

Additional functionality:

- Store history of operations in DB (for example MongoDB) or in a file.
- Add API endpoint to get history of operations, ability to specify time range.

Reference list

https://www.codecademy.com/
https://learn.javascript.ru/
https://developer.mozilla.org/ru/
https://nodejs.org/
https://docs.mongodb.com/manual/installation/
https://www.npmjs.com/package/mongodb