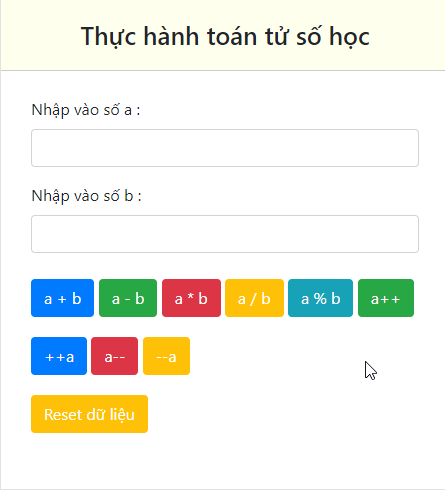
**Chú ý:**

* **Có thể áp dụng linh hoạt Bootstrap hoặc FontAwesome để hoàn thành bài tập tương ứng**

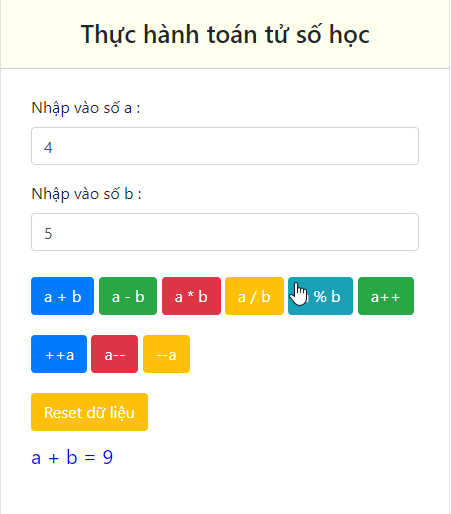
# Bài tập

1. Xây dựng giao diện như hình sau, và thực hiện các chức năng sau:

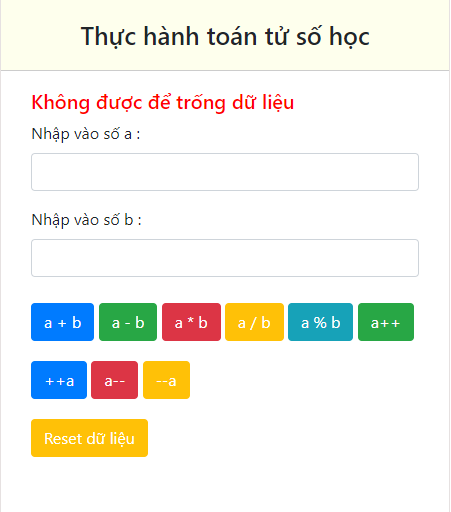
* Hiển thị các kết quả tương ứng với các chức năng tính toán như trong hình
* Khi click Reset dữ liệu, sẽ xóa các giá trị của 2 trường Nhập vào số a và Nhập vào số b



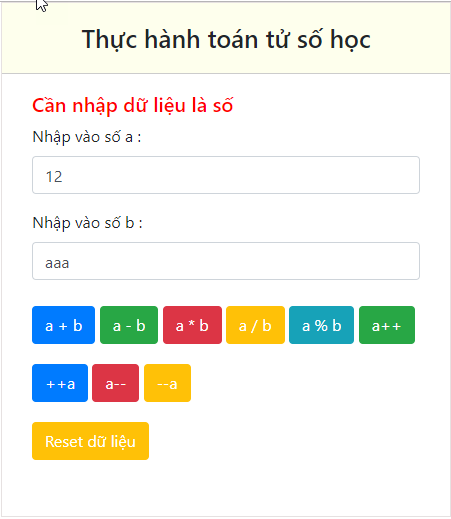
* Khi nhập dữ liệu hợp lệ (không để trống, là số) thì hiển thị kết quả tính toán như hình sau



* Trong trường hợp user để trống, khi click chức năng tính toán không thực hiện phép tính, và hiển thị ra thông báo lỗi như sau:



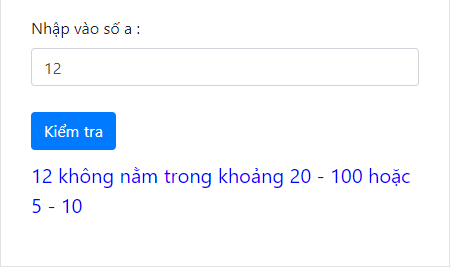
* Trong trường hợp user nhập dữ liệu không phải là số, khi click chức năng tính toán thì không thực hiện phép tính mà đưa ra thông báo lỗi như sau:



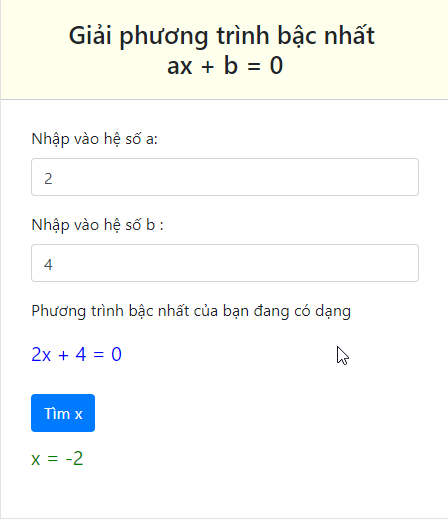
* **Gợi ý các hàm có thể được sử dụng trong bài tập:**
* Hàm check 1 giá trị có phải số hay không: **isNaN(<giá-trị-cần-check>),** hàm này trả về TRUE nếu không là số, FALSE nếu là số
* Hàm ép kiểu từ chuỗi sang số nguyên: **parseInt(<chuỗi-cần-ép>),** hàm này trả về số nguyên tương ứng sau khi ép kiểu thành công

1. Xây dựng giao diện như hình sau, yêu cầu:

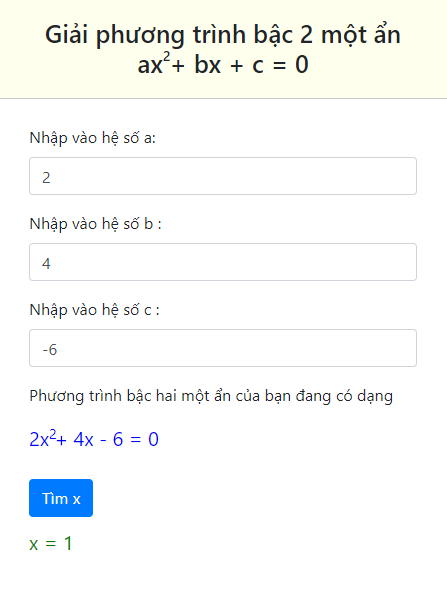
* Nhập vào 1 số bất kỳ, kiểm tra xem số đó có nằm trong phạm vi từ 1 – 10 hoặc 50 – 100 hay không
* Kết quả sẽ được hiển thị như trong hình



1. Giải phương trình bậc nhất 1 ẩn với giao diện như sau



1. Giải phương trình bậc 2 một ẩn với giao diện như sau:

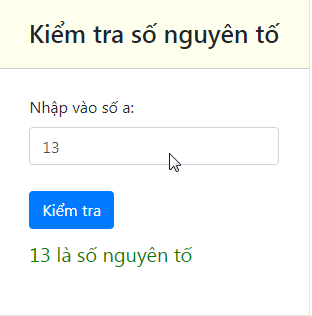


1. Xây dựng giao diện sau, yêu cầu:

* Kiểm tra xem số nhập vào có phải số nguyên tố hay không

**Tips**: số nguyên tố là số chỉ chia hết cho 1 và chính bản thân nó

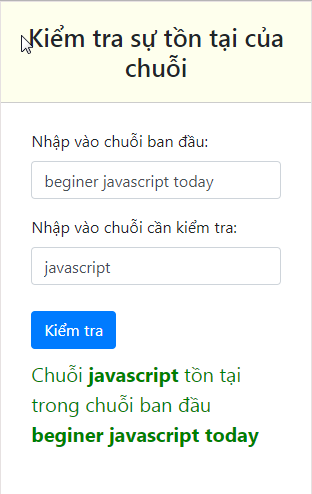
**VD**: 5, 7, 11, 13 là số nguyên tố



1. Xây dựng giao diện như sau, yêu cầu:

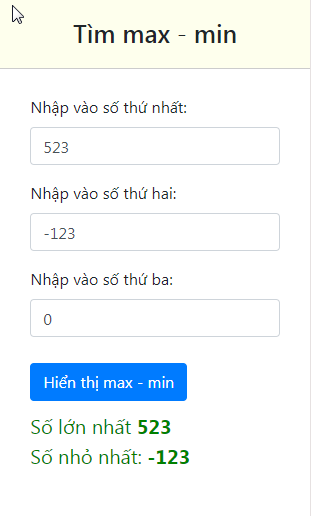
* Nhập vào chuỗi ban đầu và chuỗi cần kiểm tra, nếu chuỗi cần kiểm tra xuất hiện trong chuỗi ban đầu, thì hiển thị message tồn tại, ngược lại sẽ hiển thị không tồn tại

Chú ý: tự tìm kiếm để tìm ra phương pháp phù hợp để xử lý yêu cầu



1. Xây dựng giao diện như hình sau, yêu cầu:

* Nhập vào 3 số bất kỳ
* Tìm số lớn nhất, nhỏ nhất trong 3 số vừa nhập, hiển thị thông báo như trong hình



1. Xây dựng giao diện như sau:



Yêu cầu:

* Khi hover qua từng item, sẽ hiển thị giao diện như sau
* SỬ DỤNG JAVASCRIPT ĐỂ HIỂN THỊ HOVER, KHÔNG SỬ DỤNG CSS TRONG TRƯỜNG HỢP NÀY

