# Debouncing in Search - Concept Notes

What is Debouncing?

Debouncing is a technique used in programming to delay the processing of input until a certain amount of time has passed without any new input.

Why Use Debouncing in Search?

Without debouncing:

- Typing in a search box like "Nikhil" would trigger an API call for every keystroke.

- That means calls for "N", "Ni", "Nik", etc.

Problems without debouncing:

- Too many API calls

- Server overload

- Out-of-order or flickering search results

How Debouncing Works

Debouncing waits until the user stops typing for a fixed delay (e.g., 300ms) before triggering the search.

JavaScript Example:

```
function debounce(func, delay) {

let timeout;

return function (...args) {

clearTimeout(timeout);

timeout = setTimeout(() => func(...args), delay);

};
```

```
}

const searchHandler = debounce((query) => {

fetch(`/search?q=${query}`);

}, 300);


document.getElementById("searchInput").addEventListener("input", (e) => {

searchHandler(e.target.value);

});
```

Summary:


Without Debounce:

- Fires API on every key

- High API load

- May show wrong/slow results


With Debounce:

- Waits for user to stop typing

- Efficient API calls

- Smooth user experience