1. **useToggle**

- Build a hook that manages a boolean.
- Returns: `{ value, toggle, setTrue, setFalse }`

2. **useCounter**

- Build a hook that manages a number with `inc/dec/reset`.
- Inputs: `initial`
- Returns: `{ value, inc, dec, reset }`

3. **useCounterWithStep**

- Like `useCounter`, but with step.
- Inputs: `initial, step`
- Returns: `{ value, inc, dec, reset }`

4. **useInput**

- Manage an input’s value.
- Inputs: `initial`
- Returns: `{ value, onChange, setValue, reset }`
  (`onChange` should work directly in `<input onChange={...} />`)

5. **useTextarea**

- Same as `useInput` but intended for `<textarea>`.
- Keep the API identical to `useInput`.

6. **useLocalStorageState**

- Keep state synced to `localStorage`.
- Inputs: `key, defaultValue`
- Returns: `{ value, setValue, remove }`
  (`remove` clears the key and resets to default)

7. **useSessionStorageState**

- Same as above but using `sessionStorage`.

8. **usePrevious**

- Store the previous value of something.
- Inputs: `value`
- Returns: `previousValue`

9. **useDebouncedValue**

- Debounce a changing value.
- Inputs: `value, delayMs`
- Returns: `debouncedValue`

10. **useThrottleValue**

- Throttle a changing value.
- Inputs: `value, intervalMs`
- Returns: `throttledValue`

11. **useInterval**

- Run a callback every X ms while enabled.
- Inputs: `callback, intervalMs, enabled`
- Returns: `{ start, stop, isRunning }`

12. **useTimeout**

- Run a callback once after X ms.
- Inputs: `callback, delayMs, enabled`
- Returns: `{ start, cancel, isScheduled }`

13. **useDocumentTitle**

- Set `document.title` from React state.
- Inputs: `title`
- Returns: nothing (just side effect)

14. **useBodyClass**

- Add/remove a class on `document.body`.
- Inputs: `className, enabled`
- Must cleanup on unmount or when disabled.

15. **useWindowSize**

- Track `{ width, height }` of the window.
- Returns: `{ width, height }`
- Updates on resize.

16. **useOnlineStatus**

- Track whether browser is online.
- Returns: `isOnline`
- Use `online/offline` events.

17. **useKeyPress**

- Detect if a specific key is currently pressed.
- Inputs: `key` (example: `"Shift"` or `"Escape"`)
- Returns: `isPressed`

18. **useOutsideClick**

- Detect clicks outside a given element.
- Inputs: `ref, onOutsideClick, enabled`
- Use a `ref` created by `useRef()` in the component.

19. **useFetchJson**

- Fetch JSON from a URL.
- Inputs: `url`
- Returns: `{ data, loading, error, refetch }`

20. **useClipboard**

- Copy text to clipboard.

- Returns: `{ copy, lastCopied, error }`
- `copy(text)` attempts to copy and updates state.