

#JavaScript-with-JC

```
// 👉 1) Let's implement ClearAllTimeout()
let globalObject;
if (typeof window === "undefined") {
  globalObject = global;
} else {
  globalObject = window;
}

const timeoutIds = [];

const originalTimeout = globalObject.setTimeout;

globalObject.setTimeout = function (cbFn, delay) {
  const timeoutId = originalTimeout(cbFn, delay);
  timeoutIds.push(timeoutId);
  return timeoutId;
};

globalObject.clearAllTimeout = function () {
  while (timeoutIds.length) {
    clearTimeout(timeoutIds.shift());
  }
  console.log("All above setTimouts are cleared");
};

setTimeout(() => {
  console.log("first");
}, 1000);

setTimeout(() => {
  console.log("second");
}, 2000);

setTimeout(() => {
  console.log("third");
}, 3000);

clearAllTimeout();
// 👉 output
// All above setTimouts are cleared
```

(Jayesh Choudhary)

#JavaScript-with-JC

```
// 👉 2) Let's implement ClearAllInterval()
let globalObject;
if (typeof window === "undefined") {
  globalObject = global;
} else {
  globalObject = window;
}

const intervalIds = [];

const originalInterval = globalObject.setInterval;

globalObject.setInterval = function (cbFunc, delay) {
  const intervalId = originalInterval(cbFunc, delay);
  intervalIds.push(intervalId);
  return intervalId;
};

globalObject.clearAllInterval = function () {
  while (intervalIds.length) {
    clearInterval(intervalIds.shift());
  }
  console.log("All above setIntervals are cleared");
};

setInterval(() => {
  console.log("first");
}, 500);

setInterval(() => {
  console.log("second");
}, 1000);

setInterval(() => {
  console.log("third");
}, 2000);

clearAllInterval();
// 👉 output
// All above setIntervals are cleared
```

(Jayesh Choudhary)