

# ChatGPT: Post Objects with Messages

Does javascript postMessage send objects or just strings?

The postMessage method in JavaScript can send **both strings and objects**, but there are some important details to keep in mind:

---

## 1. Sending a String

The most basic way to use postMessage is by sending a string:

```
iframe.contentWindow.postMessage("Hello, iframe!", "https://example.com");
```

On the receiving page:

```
window.addEventListener("message", function(event) {  
    console.log(event.data); // Outputs: "Hello, iframe!"  
});
```

---

## 2. Sending an Object

You can also send JavaScript objects using postMessage. **The object is automatically serialized (converted to JSON) and deserialized.**

```
const messageData = { action: "scroll", position: 500 };
```

```
iframe.contentWindow.postMessage(messageData, "https://example.com");
```

On the receiving page:

```
window.addEventListener("message", function(event) {  
    console.log(event.data.action); // Outputs: "scroll"  
    console.log(event.data.position); // Outputs: 500  
});
```

---

## 3. Security Considerations

- Always **specify the target origin** instead of using "\*" to prevent security vulnerabilities.
- If the message contains sensitive data, validate event.origin before processing.

Example of validating the sender:

```
window.addEventListener("message", function(event) {  
    if (event.origin !== "https://trusted-source.com") return;  
    console.log(event.data);  
});
```

---




#### 4. Sending Complex Data (With Transferable Objects)

For better performance, `postMessage` also supports **Transferable Objects**, like `ArrayBuffer`, which are moved instead of cloned.

Example:

```
const buffer = new ArrayBuffer(1024);  
  
iframe.contentWindow.postMessage(buffer, "https://example.com", [buffer]);
```

#### Summary:

-  **Strings** → Simple messages.
-  **Objects** → JSON serialization.
-  **Transferable Objects** → Faster data transfer.

Let me know if you need an example! 