

Explain Callback functions and Some scenarios where they are used.

How will you handle Errors in a callback function.

A callback function is a function that is passed as an argument to another function and is invoked later, typically after a task is completed.

```
function greet(name, callback) {  
  console.log("Hello, " + name);  
  callback();  
}
```

```
function sayBye() {  
  console.log("Goodbye!");  
}
```

```
greet("Bharat", sayBye);
```

Handling Errors in Callback Functions

To handle errors properly in callbacks, use the error-first callback pattern, where the first parameter of the callback is reserved for the error (if any).

```
function fetchData(callback) {  
  let error = false;  
  let data = "Here is your data";
```

```
if (error) {  
    callback("Error occurred", null);  
} else {  
    callback(null, data);  
}  
}
```

```
fetchData(function (err, result) {  
    if (err) {  
        console.error("Error:", err);  
    } else {  
        console.log("Result:", result);  
    }  
});
```