### Global Execution Context Global Memory Value Identifier

### CREATION PHASE

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}

console.log(broadcast(input));
```

Global Memory	
Value	
<global object=""></global>	
<window object=""></window>	

Current Operation: **Program** 

### CREATION PHASE

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}

console.log(broadcast(input));
```

Global Memory	
Identifier	Value
window	<global object=""></global>
this	<window object=""></window>
name	undefined

Current Operation: Variable Declaration

### CREATION PHASE

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}

console.log(broadcast(input));
```

Global Memory	
Identifier	Value
window	<global object=""></global>
this	<window object=""></window>
name	undefined
input	<uninitialized></uninitialized>

Current Operation: Variable Declaration

### CREATION PHASE

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}

console.log(broadcast(input));
```

Global Memory	
Identifier	Value
window	<global object=""></global>
this	<window object=""></window>
name	undefined
input	<uninitialized></uninitialized>
broadcast	<fn object=""></fn>

Current Operation: Function Declaration

```
Global Execution Context
```

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}

console.log(broadcast(input));
```

Global Memory	
Identifier	Value
window	<global object=""></global>
this	<window object=""></window>
name	"Luigi"
input	<uninitialized></uninitialized>
broadcast	<fn object=""></fn>

Current Operation: Variable Assignment

```
Global Execution Context
```

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}

console.log(broadcast(input));
```

Global Memory	
Identifier	Value
window	<global object=""></global>
this	<window object=""></window>
name	"Luigi"
input	"Hello, World!"
broadcast	<fn object=""></fn>

Current Operation: Variable Assignment

```
Global Execution Context
```

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}

console.log(broadcast(input));
```

Global Memory	
Identifier	Value
window	<global object=""></global>
this	<window object=""></window>
name	"Luigi"
input	"Hello, World!"
broadcast	<fn object=""></fn>

Current Operation: Call Expression

### 

console.log(broadcast(input));

```
Local Execution Context
```

CREATION PHASE

```
function broadcast(message) {
  return `${name} says ${message}`;
}
```

Local Memory	
Identifier	Value
arguments	{length: 0}
this	<window obj=""></window>

Current Operation: Function Expression

```
Global Execution Context
                                                                 EXECUTION PHASE
                                                             Global Memory
 var name = "Luigi";
 let input = "Hello, World!";
                                                                        Value
                                                          Identifier
                                                                     <Global Object>
                                                       window
 function broadcast(message) {
                                                                     <window Object>
    return `${name} says ${message}`,
                                                       this
                                           Local Execution Context
 console.log(broadcast(input));
                                           function broadcast(message) {
```

Current Operation: Function Expression

### CREATION PHASE

return `\${name} says \${message}`;

Local Memory	
Identifier	Value
arguments	{     0: message,     length: 1 }
this	<window obj=""></window>

```
Global Execution Context
                                                                  EXECUTION PHASE
                                                              Global Memory
 var name = "Luigi";
 let input = "Hello, World!";
                                                          Identifier
                                                                         Value
                                                                      <Global Object>
                                                        window
 function broadcast(message) {
                                                                      <window Object>
    return `${name} says ${message}`,
                                                        this
                                           Local Execution Context
 console.log(broadcast(input));
```

function broadcast(message) {

return `\${name} says \${message}`;

CREATION PHASE

Value

0: message,

<window Obj>

undefined

length: 1

Local Memory

Identifier

arguments

this

message

Current Operation: Variable Declaration

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
   return `${name} says ${message}`
}

Local Execution Context

EXECUTION PHASE

Global Memory

Identifier Value

**CGlobal Object**

**Local Execution Context**

Local Execution Context**

**Context**

**Context**

**Context**

**Local Execution Context**

**Context**

**Local Execution Context**

**
```

console.log(broadcast(input));

```
function broadcast(message) {
  return `${name} says ${message}`;
}
```

Local Memory	
Identifier	Value
arguments	<pre>{   0: message,   length: 1 }</pre>
this	<window obj=""></window>
message	"Hello, World!"

**EXECUTION PHASE** 

Current Operation: Variable Assignment

```
Global Execution Context
                                                                         EXECUTION PHASE
                                                                     Global Memory
   var name = "Luigi";
   let input = "Hello, World!";
                                                                 Identifier
                                                                                Value
                                                                             <Global Object>
                                                              window
    function broadcast(message) {
      return `${name} says ${message}`,
                                                              this
                                                                             <window Object>
                                                 Local Execution Context
                                                                                                     EXECUTION PHASE
                                                                                                   Local Memory
   console.log(broadcast(input));
                                              function broadcast(message) {
                                                                                                Identifier
                                                                                                            Value
                                                   return `${name} says ${message}`;
                                                                                               arguments
                                                                                                          0: message,
                                                                                                          length: 1
                                                                                               this
                                                                                                         <window Obj>
                                                                                                          "Hello,
                                                                                               message
                                                                                                           World!"
Current Operation: Block Statement
```

```
Global Execution Context
                                                                 EXECUTION PHASE
                                                             Global Memory
 var name = "Luigi";
 let input = "Hello, World!";
                                                                       Value
                                                         Identifier
                                                                     <Global Object>
                                                       window
 function broadcast(message) {
                                                                    <window Object>
    return `${name} says ${message}`,
                                                       this
                                          Local Execution Context
 console.log(broadcast(input));
                                           function broadcast(message) {
                                             return `${name} says ${message}`;
```

Current Operation: Return Statement

### **EXECUTION PHASE**

Obj>

Local Memory	
Value	
<pre>{   0: message,   length: 1 }</pre>	
<window obj<="" th=""></window>	
"Hello, World!"	

### 

function broadcast(message) {
 return `\${name} says \${message}`;
}

Local Memory Identifier Value arguments 0: message, length: 1 this <window Obj> "Hello, message World!"

**EXECUTION PHASE** 

```
Current Operation: Return Statement
```

// checking var `name` in local memory

### Global Execution Context **EXECUTION PHASE** Global Memory var name = "Luigi"; let input = "Hello, World!"; Value Identifier <Global Object> window function broadcast(message) { <window Object> return `\${name} says \${message}`, this Local Execution Context

```
function broadcast(message) {
 return `${name} says ${message}`;
```

Local Memory	
Identifier	Value
arguments	<pre>{   0: message,   length: 1 }</pre>
this	<window obj=""></window>
message	"Hello, World!"

**EXECUTION PHASE** 

Current Operation: Return Statement

```
// `name` doesn't exist in local memory
```

console.log(broadcast(input));

```
Global Execution Context
```

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}
```

console.log(broadcast(input));

Global Memory		
Identifier	Value	
window	<global object=""></global>	
this	<window object=""></window>	
name	"Luigi"	
input	"Hello, World!"	
broadcast	<fn object=""></fn>	

### **EXECUTION PHASE**

Local Memory	
Identifier	Value
arguments	{ message, length: 1 }
this	<window obj=""></window>
message	"Hello, World!"

Current Operation: Return Statement

// look for var `name` in parent global memory

```
Global Execution Context
```

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}
```

```
console.log(broadcast(input));
```

Global Memory		
Identifier	Value	
window	<global object=""></global>	
this	<window object=""></window>	
name	"Luigi"	
input	"Hello, World!"	
broadcast	<fn object=""></fn>	

### **EXECUTION PHASE**

Local Memory	
Identifier	Value
arguments	{ message, length: 1 }
this	<window obj=""></window>
message	"Hello, World!"

Current Operation: Return Statement

// `name` exist in parent global memory

# Var name = "Luigi"; let input = "Hello, World!"; function broadcast(message) { return `\${name} says \${message}} Local Execution Context EXECUTION PHASE Global Memory Identifier Value window <Global Object> this Local Execution Context

function broadcast(message) {

return `\${name} says \${message}`;

**EXECUTION PHASE** 

```
Identifier Value

arguments

{
    0: message, length: 1
    }

this <window Obj>

message

"Hello, World!"
```

Current Operation: Return Statement

// takes value of `name` in parent global memory

console.log(broadcast(input));

### Global Execution Context **EXECUTION PHASE** Global Memory var name = "Luigi"; let input = "Hello, World!"; Value Identifier <Global Object> window function broadcast(message) { return `\${name} says \${message}`, this <window Object> Local Execution Context console.log(broadcast(input)); function broadcast(message) {

return `Luigi says \${message}`;

Identifier Value

arguments {
 0: message, length: 1
 }

this <window Obj>

message "Hello, World!"

**EXECUTION PHASE** 

Current Operation: Return Statement

```
// assign "Luigi" literal in ${name}
```

### Global Execution Context **EXECUTION PHASE** Global Memory var name = "Luigi"; let input = "Hello, World!"; Value Identifier <Global Object> window function broadcast(message) { return `\${name} says \${message}`, this <window Object> Local Execution Context console.log(broadcast(input)); function broadcast(message) { return `Luigi says \${message}`;

**EXECUTION PHASE** 

```
Local Memory
 Identifier
                 Value
arguments
               0: message,
               length: 1
this
              <window Obj>
               "Hello,
message
                World!"
```

Current Operation: Return Statement

// checking var `message` in local memory

### Global Execution Context **EXECUTION PHASE** Global Memory var name = "Luigi"; let input = "Hello, World!"; Value Identifier <Global Object> window function broadcast(message) { return `\${name} says \${message}`, this <window Object> Local Execution Context console.log(broadcast(input)); function broadcast(message) { Identifier return `Luigi says \${message}`; arguments

Current Operation: Return Statement

// `message` exist in local memory

**EXECUTION PHASE** 

Value

0: message,

length: 1

Local Memory

```
Global Execution Context
                                                                          EXECUTION PHASE
                                                                      Global Memory
   var name = "Luigi";
   let input = "Hello, World!";
                                                                                  Value
                                                                  Identifier
                                                                              <Global Object>
                                                               window
    function broadcast(message) {
      return `${name} says ${message}`,
                                                               this
                                                                              <window Object>
                                                 Local Execution Context
                                                                                                       EXECUTION PHASE
                                                                                                     Local Memory
   console.log(broadcast(input));
                                                  function broadcast(message) {
                                                                                                 Identifier
                                                                                                             Value
                                                    return `Luigi says ${message}`;
                                                                                                arguments
                                                                                                            0: message,
                                                                                                            length: 1
                                                                                                this
                                                                                                           <window Obj>
                                                                                                            "Hello,
                                                                                                message
                                                                                                            World!"
Current Operation: Return Statement
// takes value of `message` in local memory
```

### Global Execution Context **EXECUTION PHASE** Global Memory var name = "Luigi"; let input = "Hello, World!"; Value Identifier <Global Object> window function broadcast(message) { return `\${name} says \${message}`, this <window Object> Local Execution Context **EXECUTION PHASE** Local Memory console.log(broadcast(input)); function broadcast(message) { Identifier Value return "Luigi says Hello, World!"; arguments 0: message, length: 1 this <window Obj> "Hello, message World!" Current Operation: Return Statement

// assign the value of `message` in \${message}

## Var name = "Luigi"; Global Memory let input = "Hello, World!"; Identifier Value function broadcast(message) { window <Global Object> return `\${name} says \${message}` this <window Object>

console.log(broadcast(input));

### Local Execution Context

### **EXECUTION PHASE**

```
function broadcast(message) {
  return `${name} says ${message}`;
}
```

// exit "Luigi says Hello, World!"

```
Identifier Value

arguments

{
    0: message, length: 1
    }

this <window Obj>

message

"Hello, World!"
```

Current Operation: Exit Function

// return to parent execution context with value

```
Global Execution Context
```

```
var name = "Luigi";
let input = "Hello, World!";

function broadcast(message) {
  return `${name} says ${message}`;
}
```

console.log("Luigi says Hello, World!");

Global Memory		
Identifier	Value	
window	<global object=""></global>	
this	<window object=""></window>	
name	"Luigi"	
input	"Hello, World!"	
broadcast	<fn object=""></fn>	

### **EXECUTION PHASE**

Local Memory	
Identifier	Value
arguments	{ message, length: 1 }
this	<window obj=""></window>
message	"Hello, World!"

Current Operation: Console Log Message

// display `Luigi says Hello, World!` in console