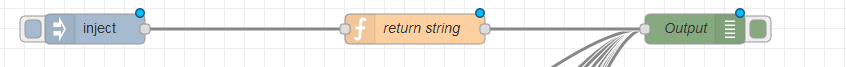
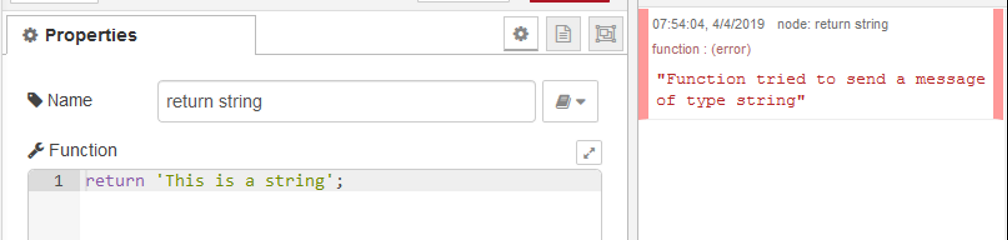


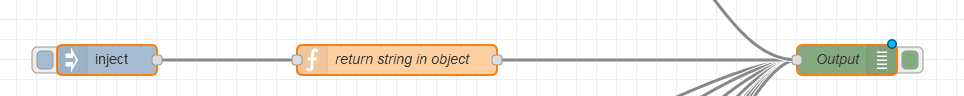
01 return string

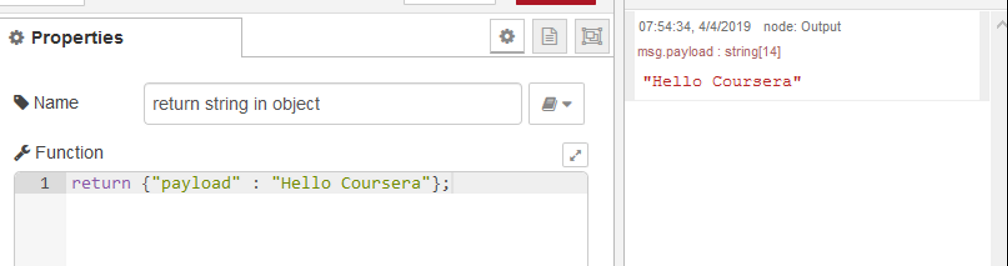




return 'This is a string';

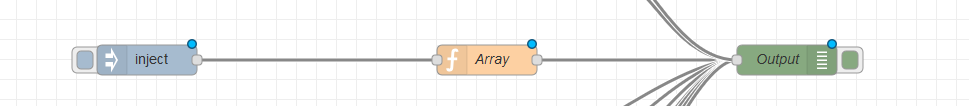
02 Return String in object





return {"payload" : "Hello Coursera"};

03 Array





var msg1 = { payload:"first message" };

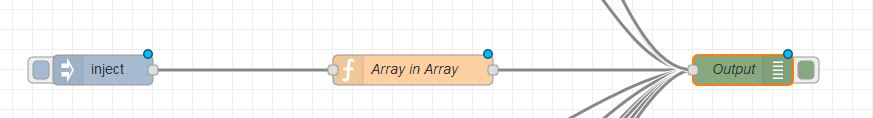
var msg2 = { payload:"second message" };

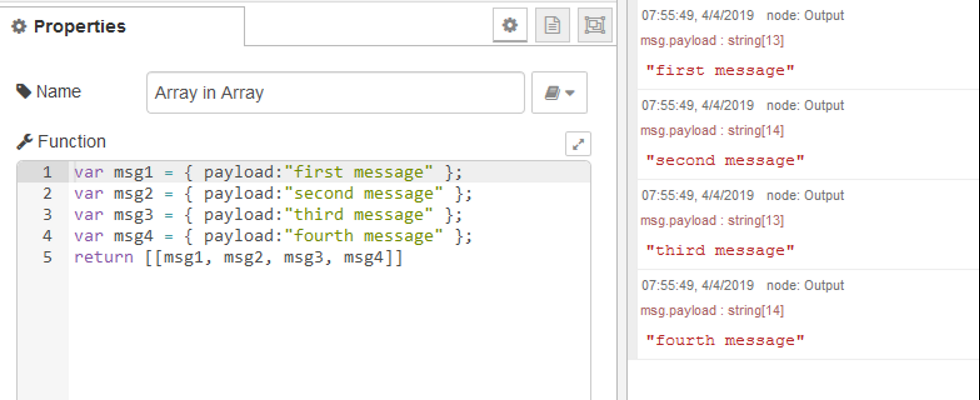
var msg3 = { payload:"third message" };

var msg4 = { payload:"fourth message" };

return [msg1, msg2, msg3, msg4];

04 Array in array





var msg1 = { payload:"first message" };

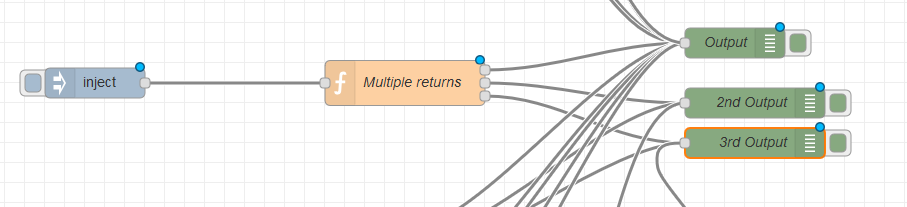
var msg2 = { payload:"second message" };

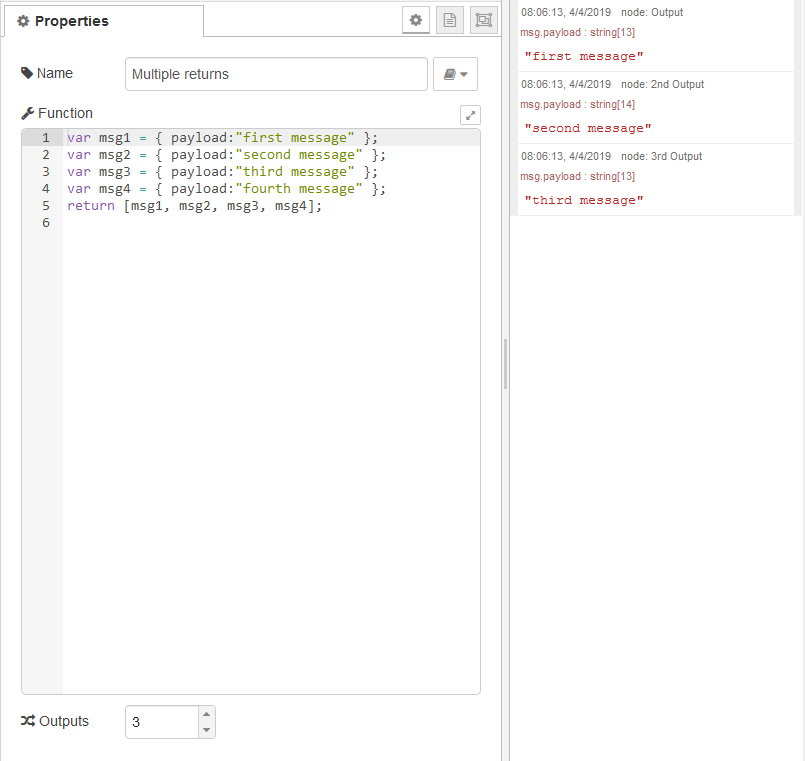
var msg3 = { payload:"third message" };

var msg4 = { payload:"fourth message" };

return [[msg1, msg2, msg3, msg4]]

05 Multiple return





var msg1 = { payload:"first message" };

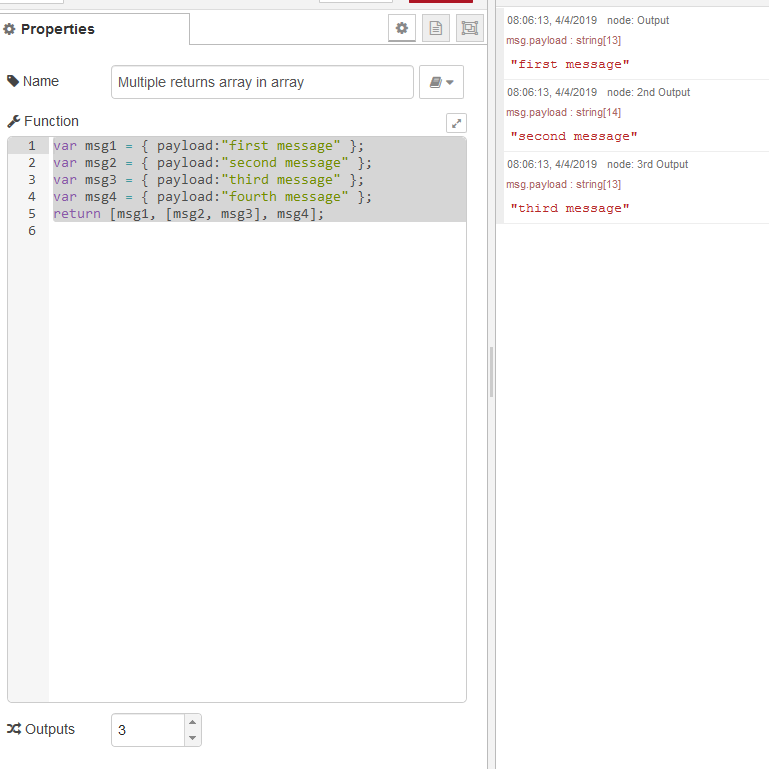
var msg2 = { payload:"second message" };

var msg3 = { payload:"third message" };

var msg4 = { payload:"fourth message" };

return [msg1, msg2, msg3, msg4];

06 Multiple returns array in array



var msg1 = { payload:"first message" };

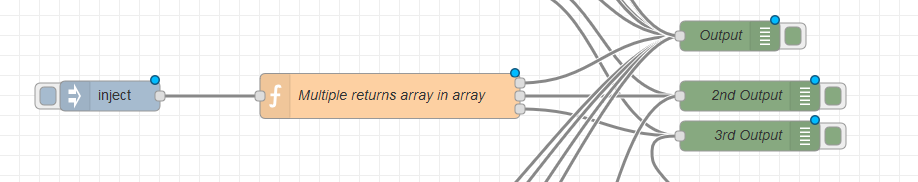
var msg2 = { payload:"second message" };

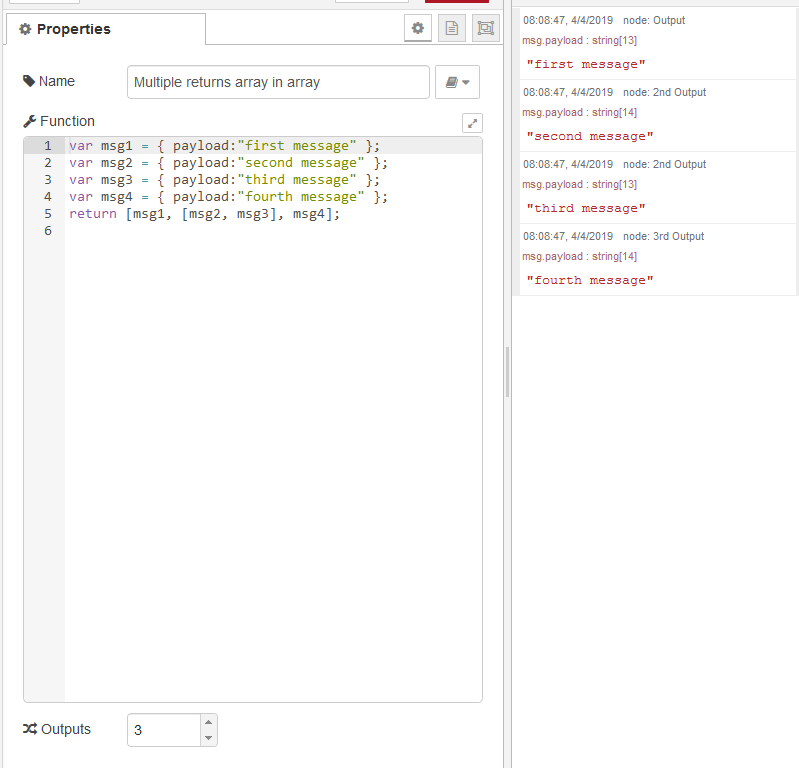
var msg3 = { payload:"third message" };

var msg4 = { payload:"fourth message" };

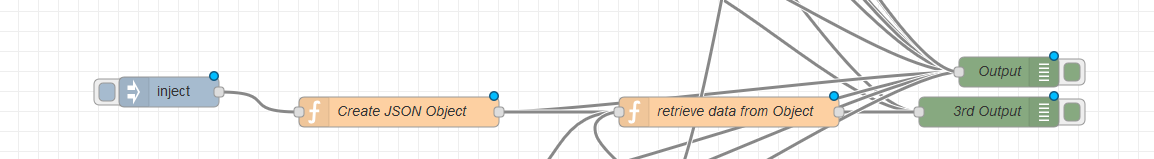
return [msg1, [msg2, msg3], msg4];

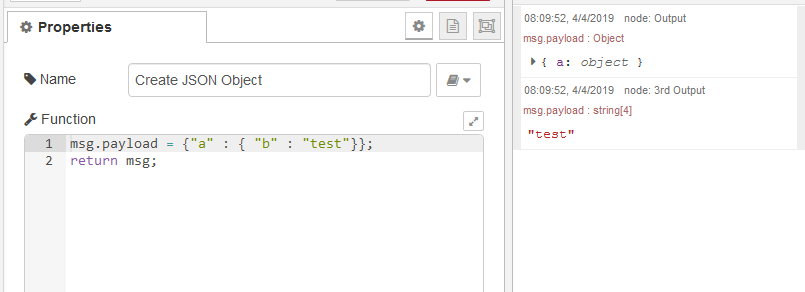
07 Multiple returns array in array





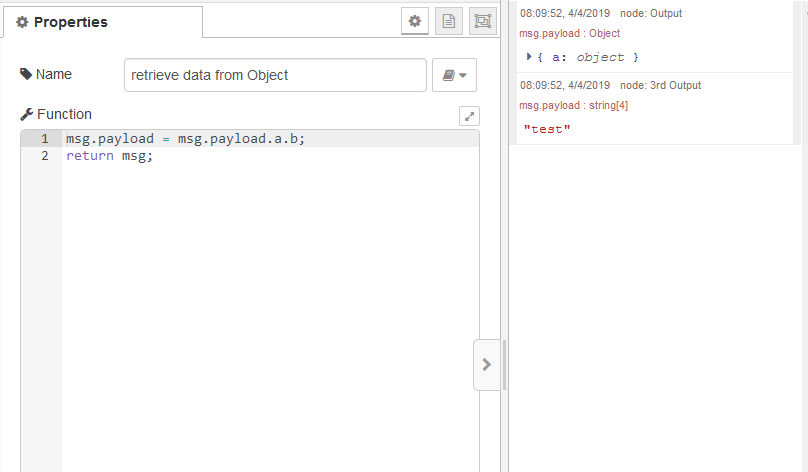
08 Create JSON Oject





msg.payload = {"a" : { "b" : "test"}};

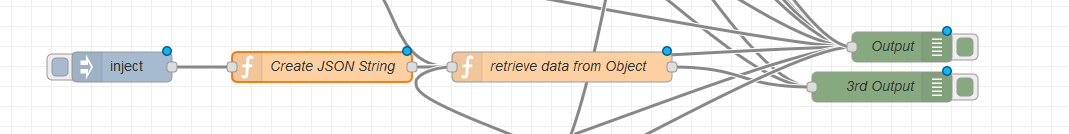
return msg;

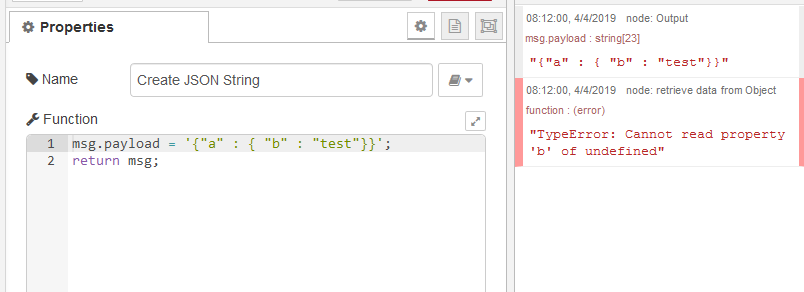


msg.payload = msg.payload.a.b;

return msg;

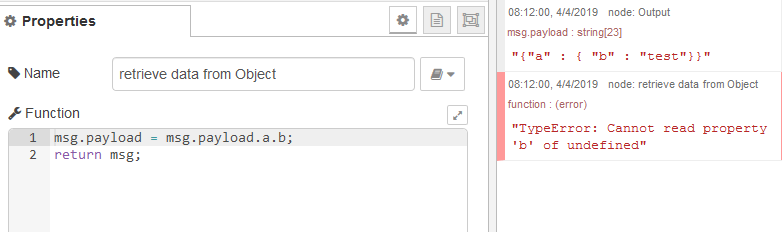
Create JSON String





msg.payload = '{"a" : { "b" : "test"}}';

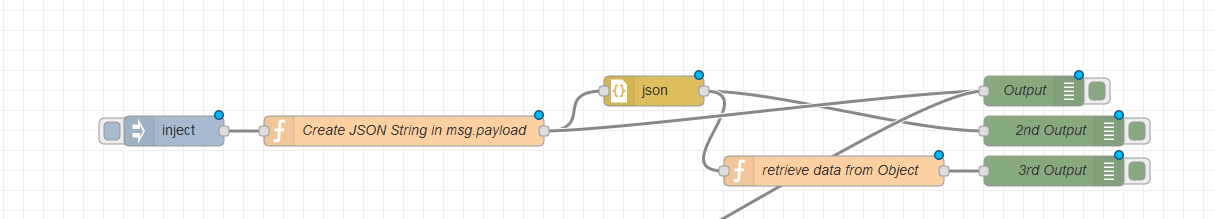
return msg;

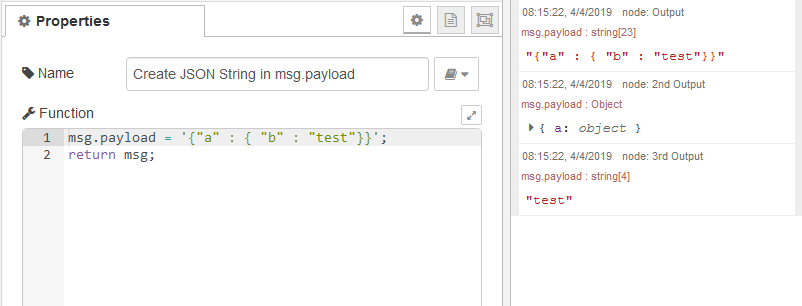


msg.payload = msg.payload.a.b;

return msg;

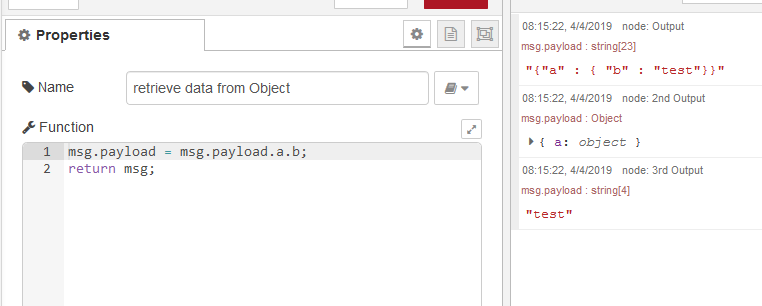
Create JSON String in msg.payload





msg.payload = '{"a" : { "b" : "test"}}';

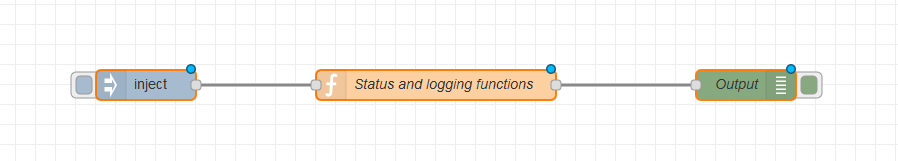
return msg;

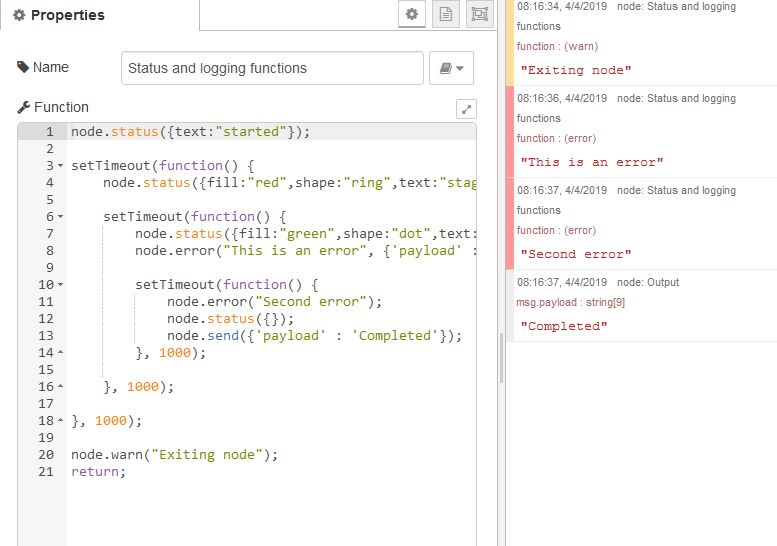


msg.payload = msg.payload.a.b;

return msg;

Status and logging functions





node.status({text:"started"});

setTimeout(function() {

node.status({fill:"red",shape:"ring",text:"stage 1"});

setTimeout(function() {

node.status({fill:"green",shape:"dot",text:"stage 2"});

node.error("This is an error", {'payload' : 'Caught error'});

setTimeout(function() {

node.error("Second error");

node.status({});

node.send({'payload' : 'Completed'});

}, 1000);

}, 1000);

}, 1000);

node.warn("Exiting node");

return;