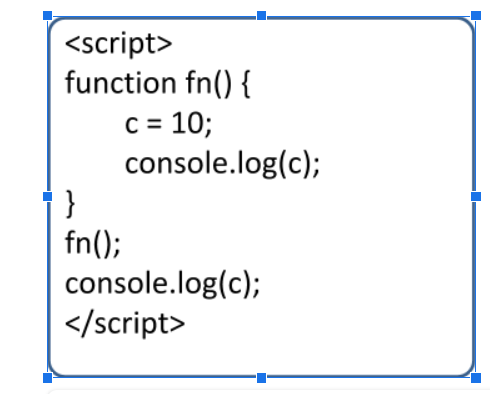
Rajalakshmanan. C

E0219054

1.For



In Compilation:

If c is declared using var, let or const keyword the compiler throws error as variable c is function scope, hence it can’ t be accessed outside the function.

If the variable is not assigned with var , let or const then the c is global scope so it is accessible any where

In Execution:

This is a case of undeclared so inside the function scope at c=10 both declaration and initialization then assignment occurs at c=10

With var:

<!DOCTYPE html>

<hmtl>

    <head>

        <title>Obervation var</title>

        <script>

            function fn(){

                var c=10;

                console.log(c);

                }

                fn();

                console.log(c);

        </script>

    </head>

</hmtl>

Output:



For Let:

<!DOCTYPE html>

<hmtl>

    <head>

        <title>Obervation let</title>

        <script>

            function fn(){

                let c=10;

                console.log(c);

                }

                fn();

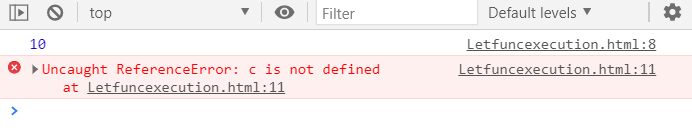
                console.log(c);

        </script>

    </head>

</hmtl>

Output:



With const:

<!DOCTYPE html>

<hmtl>

    <head>

        <title>Obervation const</title>

        <script>

            function fn(){

                const c=10;

                console.log(c);

                }

                fn();

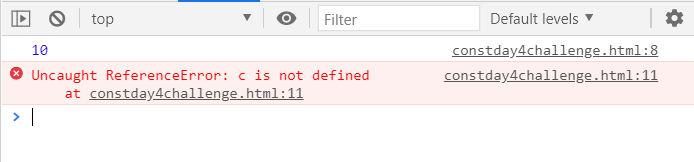
                console.log(c);

        </script>

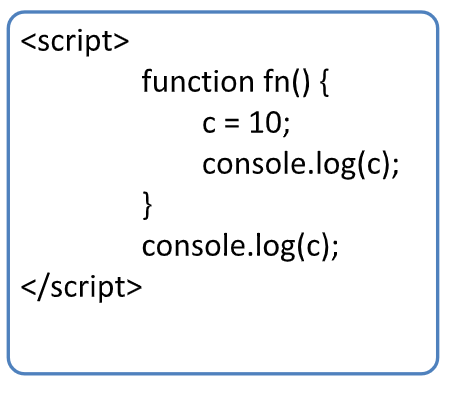
    </head>

</hmtl>

Output:



For:



In compilation:

Nothing happens console.log(c) will be skipped in compilation step

In Execution:

In execution state we will get error as c is not declared as the declaration and intialization won’t happen

For var:

<!DOCTYPE html>

<hmtl>

    <head>

        <title>Obervation var</title>

        <script>

            function fn(){

                var c=10;

                console.log(c);

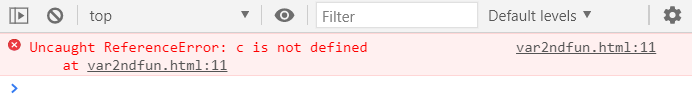
                }

                console.log(c);

        </script>

    </head>

</hmtl>

Output: 

For let:

<!DOCTYPE html>

<hmtl>

    <head>

        <title>Obervation var</title>

        <script>

            function fn(){

                let c=10;

                console.log(c);

                }

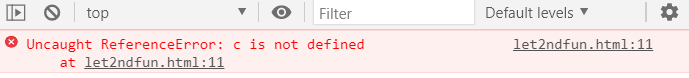
                console.log(c);

        </script>

    </head>

</hmtl>

Output:



For const:

<!DOCTYPE html>

<hmtl>

    <head>

        <title>Obervation var</title>

        <script>

            function fn(){

                const c=10;

                console.log(c);

                }

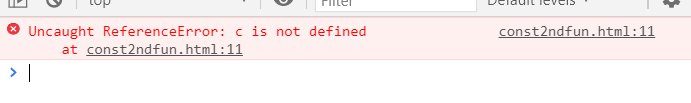
                console.log(c);

        </script>

    </head>

</hmtl>

Output:



2)EG1:

<!DOCTYPE html>

<hmtl>

    <head>

        <title>arrow execution</title>

        <script>

            const x =function(a, b) {

              return a \* b;

            };

            console.log("Anonymous function");

            console.log(x);

            console.log(x(4,3));

            const y=(l,h)=> l\*h;

            console.log("Arrow function");

            console.log(y);

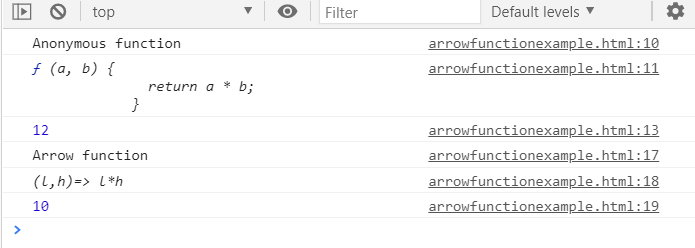
            console.log(y(5,2))

            </script>

    </head>

</hmtl>

Output:



Eg2:

<!DOCTYPE html>

<hmtl>

    <head>

        <title>arrow execution</title>

        <script>

            const prime =function() {

                var pn=prompt("enter a number");

                var flag =true;

                for(var j=2;j<(pn/2)+1;j++){

                    if (pn%j == 0){

                    flag=false;

                    break

        }

    }

        if (!flag){

            alert(pn + " is not a prime number");

            console.log(pn+" "+"is not prime number ");

        }

        else{

            alert(pn+ " is a prime number");

            console.log(pn+" "+"is a prime number ")

        }

    }

            console.log("Anonymous function");

            console.log(prime);

            console.log(prime());

            const y=()=> {

                var pn=prompt("enter a number");

                var flag =true;

                for(var j=2;j<(pn/2)+1;j++){

                    if (pn%j == 0){

                    flag=false;

                    break

        }

    }

        if (!flag){

            alert(pn + " is not a prime number");

            console.log(pn+" "+"is not prime number ");

        }

        else{

            alert(pn+ " is a prime number");

            console.log(pn+" "+"is a prime number ")

        }

            };

            console.log("Arrow function");

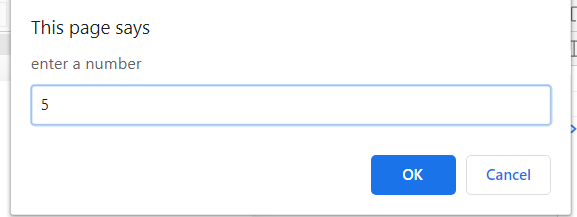
            console.log(y);

            console.log(y())

            </script>

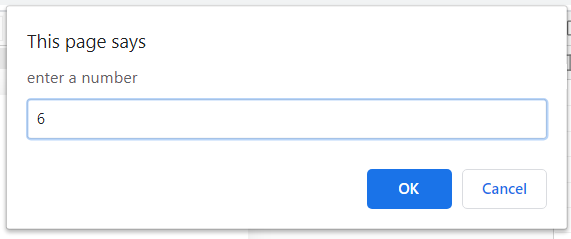
    </head>

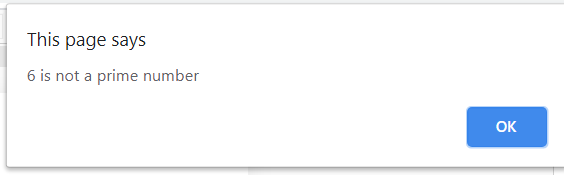
</hmtl>

For Anonymous:  


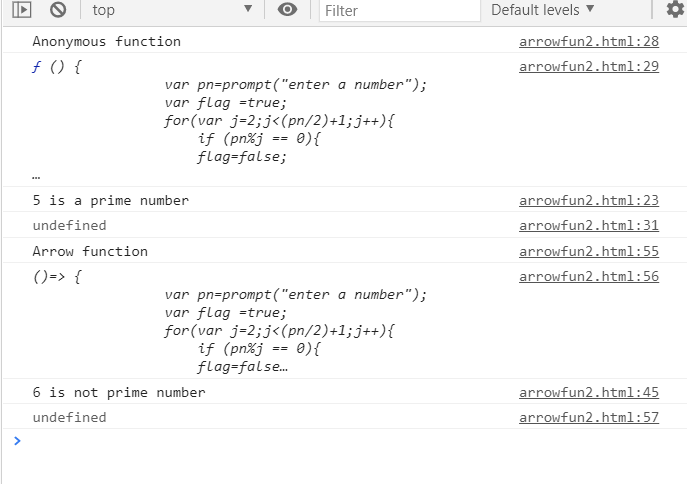


For Arrow:





Console output:



Eg3:

<!DOCTYPE html>

<html>

    <head>

        <title>

            Arrow execution

        </title>

        <script>

            console.log("Anonymous function execution");

            var an = function(a) {

                if(a==0){

                    return a+" "+"Neither even nor odd ";

                }

                else if(a%2==0){

                    return a+" "+" is an Even number ";

                }

                else{

                    return a+" "+" is an odd number ";

                }

            }(6);

            console.log(an);

            console.log("Arrow function");

            var un=(a)=>{

                            if(a==0){

                                return a+" "+"Neither even nor odd ";

                            }

                            else if(a%2==0){

                                return a+" "+" is an Even number ";

                            }

                            else{

                                return a+" "+" is an odd number ";

                            }

                            };

                        console.log(un(7));

        </script>

    </head>

    <body>

    </body>

</html>

Output:

