

What is an Anonymous function?

A function without a name is called anonymous function. Difference b/w function statement & function expression? function Expussion function statement: Vau b = function () ?

_ consolo · log ("b called";
] function a () {

console · log ("a is called");

} Diff. blw for expression & for statement is how ting.

i.e. in function Statement we can call the function

before initializing to it but in function expression

we can not call the function before initializing

because it will state to is treated by any

other variable. It is assigned 'undefined' initially Function Statement (or Function Declaration):

Syntax:

```
javascript

function myFunction() {
   // code here
}
```

Characteristics:

- Defined with the function keyword followed by the function name.
- Can be called before it's defined in the code (due to hoisting).
- Creates a named function that can be used throughout its scope.

Function Expression:

Syntax:



Characteristics:

- o Defined as part of an expression and assigned to a variable.
- o Cannot be called before it's defined (no hoisting).
- Creates an anonymous function (or can be named if needed) that's tied to the variable.

Key Differences:

1. Hoisting:

- Function statements are hoisted, meaning they can be used before they appear in the code.
- Function expressions are not hoisted, so they must be defined before they are called.

2. Usage Context:

- Function statements are used for defining functions that are meant to be available throughout their scope.
- Function expressions are often used for creating functions that are assigned to variables or passed as arguments.

In summary, function statements are like setting up a recipe that's available all the time, while function expressions are like writing down a recipe on a sticky note and only using it once you've put the note in place.

Overview of Topics Covered in Below pages:

Anonymous Function: A function without a name. Used when a name isn't needed.

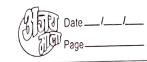
Named Function: A function with a name. Easier to use and debug.

Parameters: Names in the function definition (e.g., a, b in function add(a, b)).

Arguments: Actual values passed to the function (e.g., 3, 4 in add(3, 4)).

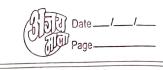


function statement aka function declaration. Anonymous fr: function without name is called anonymous It doesn't have its own identity. Anonymous for are used in a place where the function are used as values. fanction () } It will give euroy function () § console log ("b is called");



* Named function Expression It is same like function expression but instead of using anonymous for we using must a name.

Vacab = function xyzl): Consolo. log ('bis called') We can't call function as: -Xyz(); -> It will give every. Référence euroy: Xy z i's not defined. We can the it inside the function i.e. -Vaub = function xyz() { Consola log (xyz);



* Difference b/w parameters of Arguments? The values which we pass inside function are se known as alignments. > These label & identifier which get those father values are parameters. * What are first class functions? The ability to use functions as Values is known and can be passed as arguments to another function.

In can be suctions evolution the function, is called first class function. varb = function (param!) { we can also forms deturn function() { > lecturing a function 3 Consob. log(b()) 3 - 0/p: