> Functions:

calling:
const result = const-var (params) **Functions are also a kind of Objects.

Recursion: A function can refer to & call itself. There are 3 ways for a function to refer to itself. il using the function name ii) specifying arguments.callee iii) An in-scope variable that refers the function. Ex: const foo= function bav(1) | bar //statements \Rightarrow 2. arguments.callee() 3.foo()

Name conflicts clousers argument obj, params, zest param Arrow functions, shorter functions, this gre-defined functions.

objects

An object is a collection of properties, & a property is an association b/w a name(key) & a value. * A property value can be a function, in which case the property is known as a method. We can create an object in 3 different Ways. i) Using object initializer.

i) Using Object initializer. Object initializers are also called as Object literals. * We generally use "." (Dot) Operator to reference the properties of an object. However, we can't use the " operator for the properties containing spaces (5) Obj. property = value 1 Numeric properties. In that case, we use '[]' index operator. let Obj= { 2:valuez "property no :value 3 Obj[property n] => value 3 Console log(obj) => will print the entire object (all the properties)

(i) Using new Keyword with Object Constructor. . To create an object, we can use the "new" keyword with object() Constructor > we can access the properties using the "." operator for inder '[j' operator let Obj-name = new Object() -> console. (og (obj) -> prints the entire objectObj. property: "property: "
Obj. property: "property: "

creates a base object initially. (building on top of a base object)

all the properties of an object will come from "Object Constructor". [var Object: Object Constructor] ii) Using Object-Create():

(15yn>
 let Obj = Object · create({ propi: val 1, prop 2: val2 3)}

. We won't use this way very often. we can create a new object from existing prototype. let Obj = Object-createc prototype-created-earliers, props) * Diff. b/w new & Object. Create(): New actually runs constructor code, whereas Object. create will not execute the Constructor code * We can have nested objects in Is.