

2024-

TYPE SCRIPT MASTERY:~ NODEJS

1st of all we download Nodejs from google. and we click LTS version. and then now we open window and search

Command prompt and check the version of Nodejs. now we typing:

node -v

v20.10.0

(GIT)

npm -v

10.2.3

Now we install Git from google and we click window and search

command prompt and check the version of Git now we typing.

git --version

v2.43.0

TYPE SCRIPT INSTALLING

open a Command prompt on terminal and run the following command.

npm install -g typescript@5.0.2

Note: on enter this command you will wait few minutes this site is waiting and now we enter this site to check the version typescript we will type

tsc -v

5.0.2

VS CODE INSTALLING :-

open vs code and go to extensions -
Search Extension. Search for typescript
and install the typescript, javascript
with language features extensions.

now you make new folder and enter
cmd and folder name typescript.
(main.ts) and make

1) `console.log("Hello, world!");`

Note: To compile this code to
Javascript, open command prompt or
terminal and run the following
command;

~~COMMAND LINE INTERFACE~~
`tsc main.ts`

COMPILE THIS CODE TO JAVASCRIPT

This will create a Javascript file
name main.js in the same directory
you can run this Javascript file
using Node.js

`node main.js`

This will print the text "Hello, world"
to the console.

STEP 00- helloworld

29-02-2024: My 3RD CLASS WORK

Create a new directory and its name is {step00-helloworld}

Then:

Press shift + right click in the folder you want to open command prompt.

Now from the context menu. Select the "open command window here" option. => [new terminal]

Now we type tsc --init
then open tsconfig.json

Now open command prompt and giving the following command

npm init -y

When we enter this command they giving the version of tsconfig.json

/n

newline me exist karne
ki lie hota hai

$= =$

\neq

$>$

$<$

{ kaya dono number baraber ha
{ dono value baraber nahi ha
{ Greater than 3
{ less than 3

OPERATORS In TYPESCRIPT

Sab se pehle new file create karyge
wo k bad pkg.json ki file banayge, js ki file
banayge uske bad arithmatic operators ki
practice karyge uske bad logical operators

TODO list

Todo list me (push) ki
help se file me cheze add karsaktey
Rain

1. Push addition karta hai
2. Pop last element hide kartatai
3. Concat is se dono bracket kaek lejaty

/// while loop isme wo print karwata hai
Jisne hum use number den
Example.

let ramadanDays = 0;
while (ramadanDays <= 6) {

console.log("Fasting")

console.log("Prayers")

ramadanDays++

ise ye 7 be print karwage

q k 6 upper hai or bad me humne
post increment.

~~no function :-~~

Multiple lines of code ko store karne ke lie function use hoty hain varia me choli value store karte hain function ko simple console karwane ka syntax

```
function dinner(eat:string, drink:string){  
    console.log(`—— ${eat} —— ${drink}`)  
}
```

dinner()

Jaise upper dinner likha hai isse multiple time likh sakte hain.

RETURN FUNCTION:-

```
1) Syntax:- function month(name:string){  
    return name  
}
```

console.log(month("Monday")).

let monthh = month("March")

console.log(month)

Agar return statement me multiple cheje console karwani hotsa ham bar bar console karyge.

Syntax

function Postcolon(name: string) {

 return name

}

console.log(Postcolon("purple"))

a

u

y

a

u

y

Jaise isko barabar console
karwaya hai to isे tarkha return
Statement ko use karly hore hum value
barabar print karwa sakte hain.

Agar return ko value nahi to vo sirf
return wali value ko hi print karyge

Syntax

function digit(.) {

 return 90

}

console.log(digit(.)), // 90
 caller

Jab return Statement me value assign
karyge To hum () round bracket me
kuch bh mali likhye.

(Block Scope) (Arrow Function)

Offrow function bhi function k
jaise hote hain usi iska syntax
hota alog hota hai

const month = () => {}

ye bhi same function ki
tarha hota hai isme hum return
keyword bhi use kar sakte hain.

Block Scope::

ye wo function hota hai ki
block k ander ki value raha.
Ja kav wo value block kahi na kah
mattab hum last {} me jo curly
bracket use karte hain jo curly
tak to dela shie ho block scope
or jo bracket k bahan kam ho

GLOBAL Scope:: hota ho

ARROW FUNCTION WITHOUT RETURN

Simple Single variable RD without return Statement print karva Sakli return Statement print karva Sakli

return is se code ki length short hogi

Or jo kam hum return k karne hasty

They wo ho Jayga is ka syntax ye hai

Const fname = () : string => "maira"

```
let ans = fname()
```

```
console.log(ans)
```

ARRAY FUNCTIONS []

array me hum same type

multiple data store karva sakte

ham iske coding ase hoga or console ase

hoga.

```
let fruits: String [] = ['mango', 'apple', 'peach']
```

```
console.log(fruits[0])
```

```
console.log(fruits[1])
```

INDEXING

```
console.log(fruits[2])
```

are jisme elements hoge hum use number

dy k print karwaya hoga. Array ke

ander jo bh kam hota hai use elements

henty hain or number dy kar console karwane

ko indexing henty hain. Or agar sub

elements bracket k sake print

Mama hoto hum. Variable ko

```
console.log(fruits)
```

POP Functions

Agar array me se last element remove karna hoto pop method ko use karyge - iske coding ase hoti hai

let launcher = ['car', 'bike', 'auto', plane]

launcher.pop()

Console.log (launcher)

PUSH METHOD &

Push me last me elements add

hoty hain variable k bad variable ki name k bad push () round bracket me hum wo cheze likhye jo ham add karne hon ek se rojade elements bh add kar sakte hain. push me jo last me print hoga.

POP AND PUSH

Agar last element rem

mark sati ke usme add bh karne ho last me to variable k bad

"pop ()" phir - push (" ") agar to clone kam eku salt ho jaye.

SHIFT

Agar first element ko karna hoto hame shift method apply karyge.

UNSHIFT

Agar pehla element add karna
kota variable se bad uski method
use hogi or hum ise element digne k
kone kaya add karna hai.

WT ARRAYS

Agar blank array print karana
hota ham firne elements likhye utne hi
pop use karyge to [] ase output
agya. ise tarha unhe hi time *push
use karyge to [] ase output show
hoga.

ARRAY ME ARRAYS

Jay me array laga kar bh
argument deya hain usko jab tak console
kbara hota wo element ki indexning
karyge console me variable ki bad
[2] ya jo bh argument print karna
wo wo barab se number digne agar
away me away ki indexing hota
wo bh o se count hogi use 4,5,6
nahi karyge wo new array hai to
0 se uske indexing hogi agar 3 me
new array start karaka hai to
usko conste me variable ki bad
[3][2][1] ase likhye to delsa
and ay print hogi (1 conste & 2 range).

UPDATING VALUE & SPLICE

Apne value update

ko means ke second element hote

kar kuch or kuch nahi hooke jage

to use coding ase hogे.

let fruitString = ["D", "L", "A", "B"]

Fruits [1] = "Banana".

Console log (Fruits) to hamara

1 wale ki jaga updating array

ayga.

No. SPLICE & V

ye beach k elements ko
print karwata hai iske lie new
variable lega; ha phir slice laga
in π argument lega apne
Jas (1, 4) phr ye console me
1 element se 3 element tak
print karwayga 4 wala nahi
karwata. Slice k matlab portion
karna hai.

SPLICE & V

ye element ko remove or
add karta hai jose name String
k & element remove karne hon
to ham isk coding ase kryga

```
let fruits = ['apple', 'banana', 'mango', 'orange']  
let ans = fruits.splice(1, 2, "cherry")
```

console.log(fruits)

to ye out put me apples, cherry

orange) print karwayga.

Or agar second me hum 0 likhye to
ye kuch bh cut nahi karyga. matlab
purana braileet asae karyga or ye hum
isme add bh karva sakte hain 0 k bad
double code me elements add karyge.

▪ OBJECTIVE Splice & Reverse

agar last k elements change jase k
last k 2 element change to hum
splice k bad(-2) likhye ase hone jilne
bh elements change hon to (-5) ase
karyge to last wale element
print kore print nahi karyga
wo elements ko cut karyga.
use print make karwayga.

OBJECTIVE

agar single object me kam kore
use same use key deni hoti hai or
use console.log ase nahi
console.log(obj.name). console.log(obj.name)
console.log(obj.age)

is kaha variable console loga hogा means
jo cheez pehle chahye ux. loga kai
deenge.

Multiple Obj

Is kii variable k bad = arr
bracket phir curly bracket phir key
k sah key declare karoge jab ek
curly bracket hogay to john close hog
wala, laga kar new curly bracket
se new key dege key declare karoge
or jab console karvana hota

```
console.log(student[0].name)
console.log(student[1].age)
console.log(`Student [2].number`)
to ye sabse pehle wale k nam  
print karoga. Second wale ki age or  
third wale k number output me  
dega.
```

Union Type

iska matra
hai k ek variable ki ek se rayada
type assign kar sakte hain. jaise k
let data: String || number = "mara"
Console. log (data) // maira
ye iska syntex hai isko asew
ne bh use kya sakhi ham
let rollno: Number | String [] = console.
rollno = [123, 124, 567]

Type Of Interface

Iska syntax ye hai ~~not~~ example

Type human = String

Const Human : human = "Maira"

Console.log (Human).

isko hum object me bh use kar saktey
hain, jaise ki (type bridge = {

name : String

model : Number

color : String

}

Const Bridge = {

name : "blue"

model : 2023

color : "Black"

Console.log (Bridge)

agew ase console kariye to pura
object console hogar.

Console.log (Bridge.name)

agew ase console kariye to pura

name console hogar ise taika

hum console me bridge ki bad

iya kar jo bh key ko print

Karwana chahiye to print karwao

Saktey hain. (Type ko name assign

Karne ko, type alaise kerte hain.

Literals types

isme hum ~~oify~~ wahi data store karwaty hain jo hame console karwana ho jaoe k

let car: "Corolla" || 2023 = "Co
isme humne car ko variable bana le
2 output die hain ab hum indene me
Se hi kisi ek ko console karwa sakte
hain means 1c = tc bad hum agar
"Suzuki" likhyge to ye error dyga gk
humne variable mein Suzuki" likha h
nahi.

OBJECT ME ARROW FUNCTION

Object me arrow function bh use hota
hai agar object ki type hum type
alaise ko zarriye de sakte hain example

const car = {

name: "mehran"

colour: "Black"

model: 2023

Start: () => {

return "My Car is Started"

ab agar hame name console karna hoto
console.log(car.name) ase likhyge or q

Start ko console karna hoto

console.log(car.return()) ase likhyge

Nested Objects

Object k ander object create
karne ko nested object kہتھی hain jaise ko
const slip = {

CustomerName: "Moura",
product: {

Product Name: "Cake"

Price: 2500

{

{

ise tarha multiple object create kar
sakte hain means object k ander object
banane ko nested object کہتھی hain

Type Interfaces

Interface same as type above
but it use only object. iska syntax ye
hai

interface mycar {

name: String,

model: number

{

const my car: mycar = {

name: "Toyota",

model: 2024

{

Type Intersection

2 object ki type ko merge
karne ke lie type intersection use hota hai
iska syntax yeh hota hai

type Teacher = {

T-name : String

exp : number

}

type Student = {

St-name : String

roll-number : number | string

}

Const classroom: Teacher & Student = {

T-name: "ali"

Exp: "14 year"

rollnumber: 12345

St-name: "maaro"

}

console.log (classroom.StudenName);

ARRAY ME ARRAYS

array me array bh kar
sakty hain phr usko console index
ke liye karyge iska syntax yeh hai.

let fruits: string[] = ["mango",
"watermelon", "banana"];

console.log (Prints [1])

UNION TYPE IN FUNCTION:-

union type me hum 1 variable ko
2 ya use se koyada type dyte hain usko
function me toh use kar sakty hain. Syntax is
Function combine (a: number, b: number) {

 return a+b;

}

 Console.log (combine(10,20)) // 30

ise tarkha name ko bh print karwa sakty
hain. upper type strong dyge.

TYPE UNION IN FUNCTION & IF/ELSE

Agar name or number dono print
karna ho or hum function ke parameter me
2 type dyge to ye return me error dyga iske
lie ham if else Condition use karne hoga (Syntax)
Function combine (a: number | string, b: number | string) {

 if (type of a === 'number' && type of b === "number")
 return a+b;

}

 else {

 return a.toString() + b.toString();

}

 Console.log (combine (10,20))

 Console.log (combine ("maira", "naseer"))

Enums~

Syntax :-
Mango, apple, banana
3 console.log (fruits).
ye hame indexing w se value
print karwata hai or index bh data hai or agar
size 1 value print karri hoto
console.log(variable)
jab console me logayge to ye khud hame
suggestions decta hai jo upper array me hota hai.

Enum In Functions~

function p1 (p1 String) {
return p1
console.log (p1 (fruits.mango)) }

ARRAY KI LENGTH

Array ki length maloom karn
k lie console.log me variable ke bad . length
karyge to array ki length maloom hoga.

FOR LOOPS~ There are three types of

For loops:-

- ① For loop 2) For of loops 3) For in loop

FOR LOOPS~

Syntax :-

for (let i=0 ; i<10 ; i++) {
console.log (i)
}

FOR loop 3 expression. ($i = i + 5$) ye 5, 5 kar ke plus karega
use hi multiply bh kar sakte hai lekin & expression me bh
hain se indexing karne hain -

Example Syntax

```
for (1 expression ; 2 expression ; 3 expression) {  
}
```

Jaise upper i ki value 0 di hai to first expression complete hogaya ab $i \leq$ less than kia 10 ke to 2 parameter complete hogaya or third me humne post increment ki mean k 1 wale me humne bataya k 0 se start karo. Second me bataya k 0 se 10 tak le jao or third me bataya k 11 karo k value ko print karo.

FOR OF ~

ye array me use hota haiisme limit mali deti jaise home for loop me second expression me limit di isme limit deni nahi parto array k element ki counting k liye for off use hota hai. Syntax

```
let array = [10, 20, 30, 40, 50, 60]
```

```
for (let i = 0; i < array.length; i++) {
```

```
    console.log(array[i])
```

ab ye output one line se aya dega is loop ko

TRADITIONAL LOOP keliy hain.

(Ye easy hai)

FOR OF K 2 TAREEKHA

ye box of k 2 tareeka hai or ye
koyada easy hai syntax

let array = [100, 20, 500, 10, 40, 80]

for (let val of array)

{

console.log(val)

}

FOR IN

ye bh array ki tarha kam

karta hai lekin ye hame indexing data
hai

Syntax:

let array = [20, 8, 90, 40, 1000]

for (let i in array) {

console.log(i)

ye output me 0 se S tak indexing dega
or

CONSOLE 2 TAREEKHA:-

isko console karne ka 2 tareeka ye
hai jis se ye value bh indexing k sath
print karyga

console.log(array[i])

indexing k sath value print karne k lie
for in ka method use hota hai.

WHILE LOOP:-

isme per
hain or inne Sirf
hai

Syntax:

let a

while (a <

console.lo

a ++

{

iski un

ko plus karo.

FOR LOOP OR

isme

ending point pa
hota white loop

ENUM :-

Enum

done me thora za

enum

person

else

gas

{

Console.

(1) agar ase con

Console. 10

WHILE LOOPIN

Isme pehle variable declare karty
hai or isme Sirf starting or ending point aata
hai

Syntax :-

```
let a = 1  
while ( a <= 10 ) {  
    console.log(a)
```

a++
}

iska matlab hai ki ek ek kar ke value
ko plus karo.

FORLOOP OR WHILE LOOP :-

Ismi fung hota hai for loop me
ending point pata hata hai or agar pata nahi
hota while loop use hogा.

ENUM :-

Enum object ki tarha bata lein lekin
dono me thora fazil hui enum Syntax.

```
enum petrol {
```

petrol

diesel

gas

}

(1) console.log (petrol, diesel)

agar aise console karye to ye indexing dega

(2) console.log (petrol[0]) // 0



Jaise diesel ki indexing 1 hai agar index
me value chahye hon to enum ko do
karyge. **ENUM VALUE** Agar
enum ki value print karni hoto

Syntex:

```
enum Fuel {  
    patrol = "Rs 280"  
    gas = "Rs 200"  
    diesel = "Rs 230"
```

console.log(Fuel.patrol)

agar ase console karyge to patrol ki
value means ke "Rs 280" print

log.

DESTRUCTURING ARRAY

Array ko destructure ka
k lie new variable bana k upper array
me jilne elements honge atne hi new nm
type array me to print array wala hi hog

```
let StudentInfo: (String, number, boolean)[] = ["mara", 123,  
let [ename, rollno, onsite] = StudentInfo  
console.log(ename, rollno, onsite).  
to console me array wali values print  
honge.
```

NESTED ARRAY

Tuples me
banati hain iska syntax
let Enums [String, number, boolean]
["Sherry", 123, true], ["Anmol", 123]

Tuples ki nested

TUPLES

Sequences
isme jilne

elements array me add
1 number ya 1 boolean
string + number + boolean
String ki value print
dyge to error ayga
elements add honge.

let colour: [String, String, boolean].
console.log(colour)
to Console me

MODULES

js file me

variable se phle
or jaha import
import li khyge
kuch sugest
karna hoi phir
bari kar = var
to yeh file 2
file me

NESTED ARRAY IN TUPLES

Tuples me nested array bh bh
baratij hain ioka syntax ye hai.

```
let times [String, number, boolean] [] = [[ "maia", 12, true ],  
[ "Sherry", 123, true ], [ "Anna", 123, false ]]
```

Tuples & nested array are banta hai

TUPLES

Sequential data kliye ye use karta hai
isme jitne data type honge utne hi
elements array me add hargye jise agar 1 string
1 number ya 1 boolean hai to hum array me bh 1
string 1 number 1 boolean hi likhye agar 2 bar.
String ki value print hargye or type ek hi bar
dyge to error aya ja jtna data types utne hi
elements add honge. Syntax :-

```
let colour: [String, String, String] [] = [ "blue", "green", "yellow" ]
```

console.log(colour[1])

to Console me green print hogा.

MODULES

js file me kam karye wala.

variable se pahle export & keyword use hogा
or jaha import karna to us wala
import likhye jab lower karye to.

klwd suggestion dega k is file ko import
karna hai phir us file me new variable
bara kari = variable karni & console karye

to us file 2 folder me hoge wala karni is
file me console hogा.



Narrowing &

Iske lie if statement lagmi hai Syntex
let age: String number
age = 25.20205222
if (typeof age == "number") {
 console.log ("This is number")
} else {
 console.log ("This is string")
} // This is number

To Fixed &

is se ye hoga Variable me
define bh number honge ham. To fixed()
bracket me jo bh number odyge sun utne
hi number Console me ayge isse value
fir hujati hai example Syntaxes

let age: String | number

age = 25, 22235498

if (typeof age == "number") {
 console.log (age. toFixed(2))
}

To To Console me 25.22
print hogi qk hamere. fir me a likha

hum kisi bh u
use kar sakty ha
wali use kar

TYPE UNI

lekin ye hame
dega or ye u
hai.

let Val:

let Val1:

KHOD ER

function -

Three

3

- Error

REST OP

rest op

dyge Iske Typ
jitne bh
dygar. Syn
fun

String []) {

Console

3

walled

batata hai
hai syntax

hum kisi bh variable me to fixed ko
use kar sakte hain. Zaruri nahi ij ki condition
ko hi use karen.

TYPE UNKNOWN

ye type any ki barta hai
lekin ye home hamare galio per error
dega or ye use karna zayada behen
hai.

let Val: unknown

let val1: unknown = Val

KHUD ERROR CREATE KARNA

function Error(): never {

 throw new error

}

- Error()

ye home error lakin dega.

REST OPERATORS

rest operators me ... k bad name
dype iske type hamisha away hoge isme
jisme coh argument dega ye error nahi
deg. Syntax

function walled(colour:string, ...args:
string[]) {

 console.log(colour, ...args);

}

walled("red", "nic", "conveny", "actm")

06-06-2024

OPTIONAL

PARAMETER

Iska syntax ye hai or
function Student (name: String, info?: String);
console.log (name, info);

Student ("Maira", "Master")
ye arivou or normal dono
me chalta hai. isme ye hota hai jis
parameter ko optional rakhna ho use
parameter dy kar laga kar use type dedye
means k agar argument degy to bh thi
rabi degy to bh thi. isko interface me bh
use karsakte hain iska syntax hai

INTERFACE ME OPTIONAL PARAMETER

iska syntax ye hai
interface Product {
 name: String
 price?: number
}

let product: Product = {
 name: "Maira"
}

pehle ye hota tha jin
properties upper dyte they niche define

kamni pasti the
waja se ye ek
rest parameter
MEANING

ki type batana
(data-type) ye
hai. agar ham
hai to ye
mahi karyga
use pata
pari kary.
let

cons

let ag
age =
cons

) ;
ab

chala k
jabs. lag
kardi jo
Sakhi

Kamni pasti the lekin optional parameter ki
waja se ye error nahi dyga {optional or
rest parameter last me aye}

EXPLICIT CASTING

Iska matlab hai kabardasti kisi

ki type batana. Yaha Casting k Matlab hai
(data-type) ye unknown type k saath use hota
hai. agar hume java k built-in function chalana
hai to ye console me intellisense provide
nahi karyga qk upper type unknown hai
use pata hi nahi k woh kya intellisense
pass kary. Iska syntax hai -

let age: string;

console.log((age as string).toUpperCase());

let age: unknown;

age = "Twenty"

console.log(

((age as string).toUpperCase())

) ;

ab jab console me string kia to use peta
chala kise String ki tarha treat karna hai
jab lagaya to isme intellisense deni start
kardi jo bw intellisense use karni ho to kaw
Sakty hain.

STRUCTURAL TYPING

Iska matlab hai jab hum
apne multiple object ki type ko compare
karne object ki type k matlab hai object
me موجود properties os unki data type
iska syntax hai:

interface ball {
 property diameter : number → type
}

interface Sphere {
 diameter : number

let ball : Ball = {
 diameter : 10

let Sphere : Sphere = {
 diameter : 10

→

Structural Typing Iska matlab hai object ki
properties or un properties ki data
type ko compare karna.

STRUCTURAL TYPING | OBJECTS IN STRUCTURE

Ball = Sphere
Iska matlab
baaber hai sphere
na structure bhi
me lein or don
lein

NEW STRUCTURE

interface
length
diameter

let tube {
 length
 diameter

let HS {
 ball =
 = K

properties xayy
= se peh

xayada a
to compare

minimum it

to wo va

Tubes 1

one properties

Typecasting

ab hai jab hum
typo ko compare
matlab hai object
ki data type

number → type

number

= Σ

→

object ki
size ki data
na.

EGT KV STRUCT

(Ball = Sphere)

isk matlab hai ye ball k object
baaber hai sphere k object k qk in dono
na structure bkl same hai diameter dono
me hui or dono me use type number
hui

REVIEW STRUCTURAL TYPECASTING

interface tube Σ
length & number
diameter & number

Σ

let tube : tube = Σ

length : 25,

diameter : 150

LHS RHS
 $ball = tube$

STALE OBJECT

= K bad wale variable me agar

properties zayada hain to valid hai lekin agar

= Se pehle wale variable me properties

zayada or bad wale me kam hain

to compare nahi hogi agar RHS me

minimum value hon jo LHS me hain

to wo valid hai

Tubes ball invalid hai q k tube
me properties zayada hain

OBJECT RE-INITIALIZE &

let myType = {}

id: 10,

fname: "maira"

}

myType = {}

id: 56,

fname: "Ami"

}

Agar iske gaga Country kange
to ye error dega q k reinitialized me
property change nahi kar sakty new property
k lie new object ki banana paryga.
property k sath cherchao nahi kar sakty.

INDEX SIGNATURES

iska matlab hoi ki jab
advance me object ki properties na
pata hon to index signature ko use
kar sakty hoi Syntax :-

var x: {}

id: number,

keyki type any hon [x: String]: any

{:}

x = {}

id:

fname:

{}

x = {}

id

add:

x = {}

id

add

fname

hai k jab
quantity
nhi m

Safety ka
lo

KEY ST

ko comp

b

LHS

means L
D

SHIGER &

9
Pg Up

6

→

3

Pg Dn

Enter

Delete

$X = \{\}$

id: 100

fname: "maiva"

3

$X = \{\}$

id: 100

add: "Karachi"

$y = \{\}$

id: 190

add: "Karachi"

Rname: "Maiva"

Index Signature jab use hota
hai k jab type pata nahi hon or type ki
quantity bhi maloom nahi hon to ye use hota
haiisme arrow function tak hoga.

Salty hai for example:

login: () => booleam

STATE OBJECT

87 types

Iska matlab hai object ik variable

KO compare karna jax

ball = cube

LHS = RHS

means LHS barabar hai RHS k or
RHS me zayada properties hotsakti hain

FRESH OBJECT

jis line me variable ya
Object banta hai wo fresh object
hota hai or neche jab compare kya
hain to wo stale object kehlae hai

SYNCHRONOUS & ASYNCHRONOUS

jo code line log line
or without waiting chale wo Synchronous
hota hai jo waiting le wo Asynchronous
hota hai Aga Sync & beachi aesa task
perform ho jo Async ho to pure programming
rakhi Jayge

CALLBACK FUNCTIONS

As an argument function
pass karne ko call back function
kehte hain Syntax :-

Higher Order Function

```
function one() {  
    cb();  
}  
one();  
cb() {  
    console.log("one");  
}
```

Main Function

function two() {
 console.log("Async Function");
}
two();
One(two);
Ye function data lekar data hoga

NESTED

Sym
function c
const

3
inner
3
outer

OBJECT

Ty

{ little
{ little
};

is k.

IPng

T

the variable
fresh object
ab compare ka
kehla hai

SYNCHRONOUS

log line
two Synchronous
two Asynchronous
which easier task
in programming

your

function
unction

Main
Function

Call
back
way



Call back function ko as an argument
pass karte hain to (two()) & are
vali likege (two) are likege.

NESTED FUNCTION

Syntax:

```
function outerfunction() {  
    const innerfunction = outerfunction() {  
        console.log("Hello from inside");  
    }  
    innerfunction();  
}  
outerfunction();
```

OBJECT In ARRAYS~

Type book = {

title: String;

author: String;

isbn: String

}

let library: Book[] = [

{ title: "The Hobbit", author: "J.R.R.", isbn: "234567" }

{ title: "1984", author: "George Orwell", isbn: "94326" }

];

object me array banane ke lie
iska syntax ye hai

Set Time

// Main
let pro

library.push ({
name: "Dol",
author: "J.S Salinger",
isbn: "1123499624"

});
3);

Synchronous Programming

1) Call Stack :

2) Call back Queue:

Set Time Out

In function me hum kaha

ekty hain ke apko chahne me one second

kene hain tum hum apni maghi ka

Set var ekty hain isme milie se me

Time apne hain. Spiter g~

Set Timeout () => {

3, 5000)

Set Timeout () => {

console.log ("After 5 Sec");

3, 1000)

Ye agar code hai

Call back

apna wo

Code Review

Set Time Out Examples.

// Main Function.

let product = [

{ name: "Dolu", price: 120 },

{ name: "palak", price: 50 }

Sampling

J;

function one(cb: () => void) {

console.log("rowsing")

Set time out(l) => {

T cb(l)

isko yaha call isle kawayan ki itna

burn baki

One Second

}

}

}

}

Call back.

// Call back.

function two() {

console.log(product),

}

one();

Call back me jisda data hogya jo data base se

ayga wo call back me call karwaye.

Promises

```
const myfunc = () => {
    const promise ((resolve, reject) => {
        return new Promise((resolve, reject) => {
            resolve ('5+5');
        })
    })
}
```

```
my func () .then (res) => {
    console.log (res);
}
```

Agar promise pura noga agar resolve
noga to wo. then ko dhunqga.

Or agar reject noga to wo catch ko
dhunqga.

PROMISE

REJECT BOTH ARE PUNE - • CATCH

```
• catch (err) => {
    console.log (err);
}
```

```
3) • console.log (err);
```

Agar resolve change to wo
change ko. on .then wali execution kangi.

Agar reject change to wo. then
wali execution dhunqga.
cati

Cricket
• then
• catch
co

3)
Agar set

reject
Set timeo

Another Example Of Promise

reject) =>

```
set is_victory = False;
```

```
const Cricket = () => {
```

```
return new Promise ((resolve, reject) => {
```

```
if (is_victory) {
```

```
    resolve ("we won")
```

```
} ; else {
```

```
    reject ("we lost")
```

```
}
```

```
}
```

```
});
```

```
});
```

```
Cricket()
```

```
.then ((response) => {
```

```
    console.log(response);
```

```
)
```

```
.catch ((err) => {
```

```
    console.log(err);
```

```
});
```

```
Agar set line out logon koto wo are lagya
```

```
on koyga
```

```
resolve se pehle wali line me
```

```
Set timeout (1) => {
```

```
    resolve ("we won")
```

```
});
```

```
Agar li reject me bh lagya
```

```
Set timeout (1) => {
```

```
reject ("we lost")
```

```
});
```

PROMISES IN CHARTER

right

100

no ~~per~~ no. be off
hum setvan likege. Sympt:

then (response) \Rightarrow { }

console.log(response)
return "Man of the match"

$$y_{\text{obs}}(1/\text{year}) \Rightarrow 3$$

2. $\{ \text{soil} \} \text{ is a } \text{subset}$

```
console.log('second. Then, yes')
return "50 of 204"
```

= m

$$3. \quad C = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \text{ and } 0$$

console.log(res);

Finally (1) \Rightarrow 3

3) console.log('a single pattern')

When they're young. Catch them while they're small. Lop off

ASYNC

length
syntax are

st k karne

then lagne

res)

han lagye
e logni

get data()

A-Sync

TRY Successful response handle { }

Catch Error handling () { }

use some . then on . catch tha one .
try . on . catch kora hai ika syntax ye hai

async function get data() { }

try {

let res = await Cricket();

console.log(res);

} catch (err) {

console.log(`"Error", err);

{ }

get data()

logni

Object Oriented Programming

isme har cheg object hote hai.

1-class

2-object

ye dono bolat important hai jab kt
hum esp karayge unme sabse pabla ek class
kanti hai class ek map ki tashha hai subse
pabla ek class kanti hai unme hum dekhne
karte hain k is object me kaya kaya chera
hongi // Class map ka synlesu ye hai

Class Car {

 name: String = "honda"

²
 make: String = "Japan"

let mycar = new Car()

function
constructor

³

 phir jo ye

 class pro

 bababar

 > let

 let p

 let pa

 const

 isco

 sakhi rani

in OOP

parameter of
constructor
class

name

age

Car

Programmierung

nr 009. Constructors in Usings

& se object dinamico koyega means ke

parameter dekho barabar assumption console

karva sakte hain iska syntex ye hai

"ant hai jab h

pebble ek class

barha hai Sabse

hum declare

ya kaya che

ek ye hai

honda"

Java

name : String

age : number

// class property hote hain

Class person {

 name : String

 age : number

 Constructor(n : String, a : number) {

 this.name = n

 this.age = a

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

Java

name : String

age : number

Constructor(n : String, a : number) {

 this.name = n

 this.age = a

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

Java

name : String

age : number

Constructor(n : String, a : number) {

 this.name = n

 this.age = a

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

Java

name : String

age : number

Constructor(n : String, a : number) {

 this.name = n

 this.age = a

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

OOP: Constructors and Get Details:-

hamare jab bh class properties use hoge
this ka keyword lazmi lagya pehe wali
coding ko dekhi hue agay continue karyge.
constructor ke bad ek function banayge.
Syntax: Half Syntex pehe hai iska:-

get detail (): String {

return `My name is \${this.name} and my
age is \${this.age}`;

Console isko use karyge

Console.log(p.getDetails());

or ~~this~~ ^{let} constructor k function ko
call karne ke lie

Console.log()

let p = new person ("Maiya", 12)

Console.log(p.getDetails())

1. OOP

ye
hota ha

1. Inher
parent

2 Child

IS

bana

phir us

ek m

agar

error

error

Details:

classes use hoga
peche wali
karyge.
anage.
ika.

and my

on ko

↳ Extends ↳ Own Child Class ↳ Own
keyword use hota hai

WOP INHERITANCE ↳

ye wo type hai jisme har data repeat
hota haiisme a concept hote hain

↳ Inheritance ↳
parent/Base/ Superclass ↳
↳ Child Class

isme ye hota hai ek hi inherit class
bana k alag alag child class banao means
phir isme error nahi ayga warna agar hum
ek interface bana kar usko value den to
agar use interface ko again value dyge to
error ayga. but in the help of inheritance
error nahi ayga. Syntax

Class vehicle {

 engine: String

 tyre: number

 name: String

Constructor(e: String, t: number, n: String) {

 this.engine = e

 this.tyre = t

 this.name = n

}

}

Child

Class Car extends vehicle {



Constructor (engine: String, tyres: number
name: String) {
 Super(engine, tyres, name)
}

}

let car = new Car("2400cc", 4, "Rivo")

console.log(car)

ye super & keyword satata hai ke
child class me jo variable ayge wo hamare
parent class ke hope or ye keyword
constructor ke ander aya bahar nahi aya
Inheritance me ham bar bar object aya
ki parent nahi ek hi super class hama
Kao we child class me aya aya
Values de sakty hain.