INTRODUZIONE A

1. COS'È

TIPIZZAZIONE STATICA

TYPESCRIPT PERMETTE DI DEFINIRE IL TIPO DI DATI DI VARIABILI E FUNZIONI, AIUTANDO A PREVENIRE ERRORI E RENDENDO IL CODICE PIÙ CHIARO E LEGGIBILE

MAGGIOR CONTROLLO

STRUMENTI AVANZATI PER LA GESTIONE DI LIBRERIE E MODULI

TYPESCRIPT OFFRE STRUMENTI AVANZATI PER LA GESTIONE DI LIBRERIE E MODULI, FACILITANDO LA SCALABILITÀ E LA MANUTENZIONE DI PROGETTI DI GRANDI DIMENSIONI

COMPILAZIONE IN JAVASCRIPT

IL CODICE TYPESCRIPT VIENE COMPILATO IN JAVASCRIPT STANDARD, RENDENDOLO COMPATIBILE CON QUALSIASI BROWSER O AMBIENTE JAVASCRIPT

ASSEGNAZIONE DI TIPO

IN TYPESCRIPT, SPETTA AL PROGRAMMATORE ASSEGNARE ESPLICITAMENTE IL TIPO DI DATO ALLE VARIABILI AL MOMENTO DELLA LORO DICHIARAZIONE, OFFRENDO MAGGIORE CONTROLLO E PRECISIONE MA RENDENDO IL CODICE PIÙ VERBOSO

INTRODUZIONE A

2. I TIPI IN

OGGETTI

IN TYPESCRIPT, È POSSIBILE DEFINIRE COME UN OGGETTO DEVE ESSERE COMPOSTO, SPECIFICANDO IL TIPO DI DATO DI OGNI SUA PROPRIETÀ

let persona = {
name : string,
surname : string,
eta: number
}

ARRAY

TYPESCRIPT PERMETTE DI SPECIFICARE IL TIPO DI VARIABILI CHE DEVONO ESSERE CONTENUTE IN UN ARRAY, RENDENDOLO PIÙ SICURO E PREVENENDO ERRORI DI TIPO

let numbers = number [] = [1,2,3]

TUPLE

LE TUPLE IN TYPESCRIPT SONO ARRAY CON UNA LUNGHEZZA FISSA E CON TIPI DI DATI SPECIFICATI PER OGNI ELEMENTO, OFFRENDO MAGGIORE PRECISIONE E CONTROLLO

Let game = [
id: number,
codiceAccesso: string,
players: [string, string] = ['Bob', 'Alice'] → accettiamo un array con solo due elementi di tipo stringa, questa è una tupla e viene dichiarata in questo modo
]

ANY

IL TIPO ANY IN TYPESCRIPT RAPPRESENTA UN TIPO GENERICO CHE PUÒ ASSUMERE QUALSIASI VALORE POSSIBILE, MA PUÒ PORTARE A ERRORI DI COMPILAZIONE SE NON UTILIZZATO CORRETTAMENTE

let variabile: any = "valore";

UNION

LE VARIABILI DI TIPO UNION IN TYPESCRIPT POSSONO ASSUMERE UNO QUALSIASI DEI TIPI SPECIFICATI, OFFRENDO MAGGIORE FLESSIBILITÀ MA RICHIEDENDO UNA SCELTA PRECISA DEI VALORI POSSIBILI DA PARTE DEL PROGRAMMATORE

let persona = 'adult' | 'teen' | 'elder'

NB: i valori non possono essere delle primitive ma dei valori spefici scelti dal programmatore !

CUSTOM TYPE

È POSSIBILE DEFINIRE TIPI PERSONALIZZATI IN TYPESCRIPT, SPECIFICANDO I VALORI PRIMITIVI CHE DEVONO ESSERE RISPETTATI PER OGNI VARIABILE DI QUEL TIPO

type Persona = {
name: string,
cognome: string,
eta: number,
ageRange: 'teen' | 'adult' | 'elder'
indirizzo: [
via: string,
civico: number,
citta: string
]}

ENUM

GLI ENUM IN TYPESCRIPT RAPPRESENTANO INSIEMI DI COSTANTI NUMERICHE O LETTERALI CON NOMI SIGNIFICATIVI, OFFRENDO UN MODO SEMPLICE PER DEFINIRE INSIEMI DI VALORI PREDEFINITI E UNIVOCI

enum Giorno {
Lunedì,
Martedì,
Mercoledì,
Giovedì,
Venerdì,
Sabato,
Domenica
}

let oggi: Giorno = Giorno.Mercoledì;

stiamo dicendo che la variabile oggi è di tipo Giorno, nello specifico Giorno.Mercoledi