**Lezione 27/05/2024**

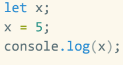
**web developer**

**JavaScript**

si occupa della parte logica del sito (funzioni, sessioni, interazioni,..... check lez iniziali)

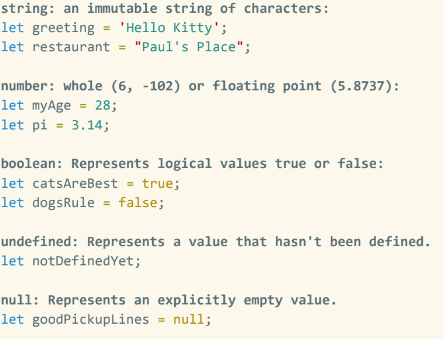
ogni istruzione in JS è una ***dichiarazione***

si fa uso delle ***variabili*** : si possono dichiarare ed inizializzare immediatamente, oppure inizializzare in due fasi, oppure cambiarne il valore

devono essere introdotte con *let* o *const*

non c’è bisogno di dichiarare la tipologia di variabile, basta inizializzarla o no (undefined no )

JS **è tipizzato** (uso di typeof: è un operatore unario. | NB **non è fortemente tipizzato**)

***Number, String, Boolean, Undefined, Null***

***Strings****:*lista ordinata di caratteri, selezionabili con [indice], oppure applicare funzioni come .length, .append…. (NB essere coerenti e scegliere di usare o “” o ‘’)

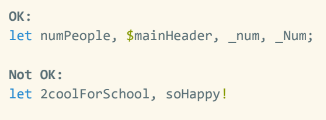
**Nomi delle variabili:**

possono iniziare anche con $ e \_ , evitando conflitti con altri elementi durante programmazione (es funzioni con stesso nome)

possono includere lettere, numeri, $ e \_ per maggiore leggibilità e dinamicità del codice

JS è **CASE SENSITIVE**

**evitare parole riservate if, class, return come nome variabile**

considerare **camelCase**

**Espressioni** (fatte anche nelle variabili)

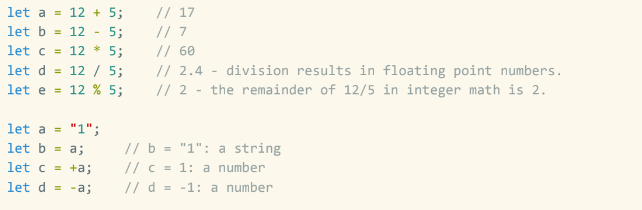
****

Inoltre non ha problemi a cambiare tipo di variabile quando si inizializza (da int a string è ok)

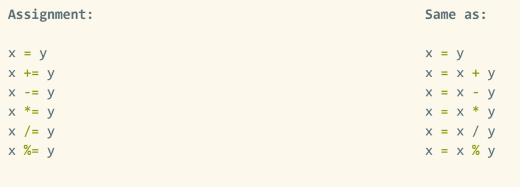
ma può essere di un tipo solo e non + insieme

**Operatori** (lavoro sugli operandi/valori numerici, possono essere unario, binario check)

*Aritmetici*

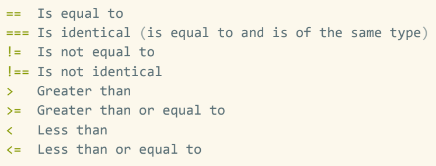
****

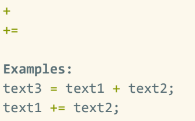
*Assegnazione (variabile iniziale dichiarata!)*

**

*Incremento(es i++, - -, !)*

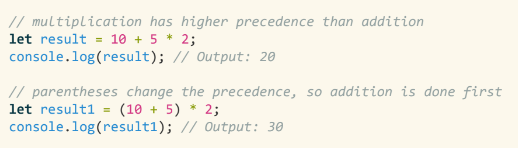
*Confronto (quindi true o false)*

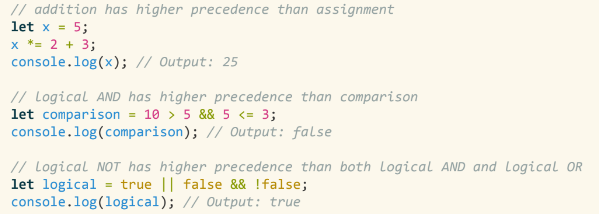
**

*Logici Operatori per stringhe*

NB numeratori non sono simboli ma anche testuali come **typeof,** restituisce il nome del tipo di variabile(ma essendo un nome è una stringa)

da tenere conto la **precedenza degli operatori** rispetto ad altri





**let:** riassegnabile / cambia tipo

**const:** non riassegnabile / non cambia tipo

entrambi sono a livello di blocco (funzionano in un blocco solo e non oltre)