

GABRIELE ROMANATO

Menu

Come convertire i numeri arabi in numeri romani con JavaScript

La conversione di numeri arabi (cioè i numeri che usiamo comunemente come 1, 2, 3, ecc.) in numeri romani è un esercizio interessante per migliorare la logica e le abilità di programmazione. I numeri romani sono un sistema numerico che utilizza lettere dell'alfabeto latino per rappresentare numeri. In questo articolo vedremo come implementare questa conversione in JavaScript.

Il sistema numerico romano utilizza le seguenti lettere per rappresentare specifici valori:

- I = 1
- V = 5
- X = 10
- L = 50
- C = 100
- D = 500
- M = 1000

Per numeri più complessi, i simboli vengono combinati seguendo alcune regole:

- Se una lettera con un valore minore precede una lettera con un valore maggiore, si sottrae il valore della prima dalla seconda (es. IV = 4).
- Se una lettera con un valore maggiore precede una con un valore minore, si sommano i valori (es. VI = 6).

Il processo di conversione da numero arabo a numero romano può essere suddiviso in diversi passaggi:

1. Definiamo una mappa che associa i numeri arabi ai loro equivalenti romani.
2. Creiamo una funzione che iteri attraverso i numeri arabi e sottragga il valore più alto possibile finché il numero non viene ridotto a zero.
3. Ogni volta che sottraiamo un valore arabo, aggiungiamo il corrispondente numero romano a una stringa di output.

Vediamo ora come implementare tutto ciò in JavaScript.

```
function convertToRoman(num) {
  // Definire una mappa di numeri arabi e i loro equivalenti romani
  const romanNumerals = [
    { value: 1000, symbol: 'M' },
    { value: 900, symbol: 'CM' },
    { value: 500, symbol: 'D' },
    { value: 400, symbol: 'CD' },
    { value: 100, symbol: 'C' },
    { value: 90, symbol: 'XC' },
    { value: 50, symbol: 'L' },
    { value: 40, symbol: 'XL' },
    { value: 10, symbol: 'X' },
    { value: 9, symbol: 'IX' },
    { value: 5, symbol: 'V' },
    { value: 4, symbol: 'IV' },
    { value: 1, symbol: 'I' }
  ];

  // Inizializzare una stringa per memorizzare il numero romano risultante
  let roman = '';
```

```

// Ciclo attraverso l'array mappa
for (let i = 0; i < romanNumerals.length; i++) {
  // Per ogni coppia valore-simbolo, sottrarre il valore arabo e aggiungere il simbolo
  romano
  while (num >= romanNumerals[i].value) {
    roman += romanNumerals[i].symbol;
    num -= romanNumerals[i].value;
  }
}

return roman;
}

// Esempio di utilizzo:
console.log(convertToRoman(1987)); // Output: MCMLXXXVII

```

Spiegazione del codice:

1. **La mappa di conversione:** Abbiamo creato un array chiamato `romanNumerals` che contiene oggetti con coppie `value` (numero arabo) e `symbol` (il corrispondente numero romano). L'ordine è importante: iniziamo con i numeri più alti per garantire che i numeri romani vengano costruiti correttamente.
2. **Ciclo attraverso la mappa:** Usiamo un ciclo `for` per attraversare ogni elemento dell'array. Per ogni elemento, verifichiamo se il numero arabo corrente è maggiore o uguale al valore nell'oggetto corrente. Se lo è, sottraiamo il valore e aggiungiamo il simbolo romano corrispondente alla stringa di output.
3. **Ciclo while:** La condizione `while` garantisce che continuiamo a sottrarre lo stesso valore e aggiungere il simbolo romano finché il numero non è più grande o uguale a quel valore (ad esempio, per il numero 3000, dobbiamo aggiungere "M" tre volte).
4. **Restituzione del risultato:** Alla fine, restituiamo la stringa `roman` che contiene il numero convertito.

Considerazioni:

1. **Numeri fuori dall'intervallo:** I numeri romani tradizionali non rappresentano numeri superiori a 3999. Se è necessario gestire numeri più grandi, si può aggiungere logica extra o utilizzare notazioni romane più avanzate come la barra sopra le lettere, che moltiplica per 1000. Tuttavia, queste estensioni non sono generalmente utilizzate e non sono coperte da questo esempio.
2. **Validità dell'input:** È una buona pratica aggiungere controlli per gestire input non validi, come numeri negativi o numeri decimali, che non sono rappresentabili con numeri romani.

Conclusione

La conversione di numeri arabi in numeri romani in JavaScript è un esercizio che aiuta a migliorare la capacità di pensare in modo algoritmico. La chiave è utilizzare una mappa ben ordinata e iterare attraverso di essa sottraendo il valore appropriato, costruendo gradualmente il numero romano finale. L'implementazione mostrata è semplice e efficace, e può essere facilmente adattata per diverse esigenze o migliorata con funzionalità aggiuntive come la gestione di numeri decimali o input non validi.

Applicazioni Correlate

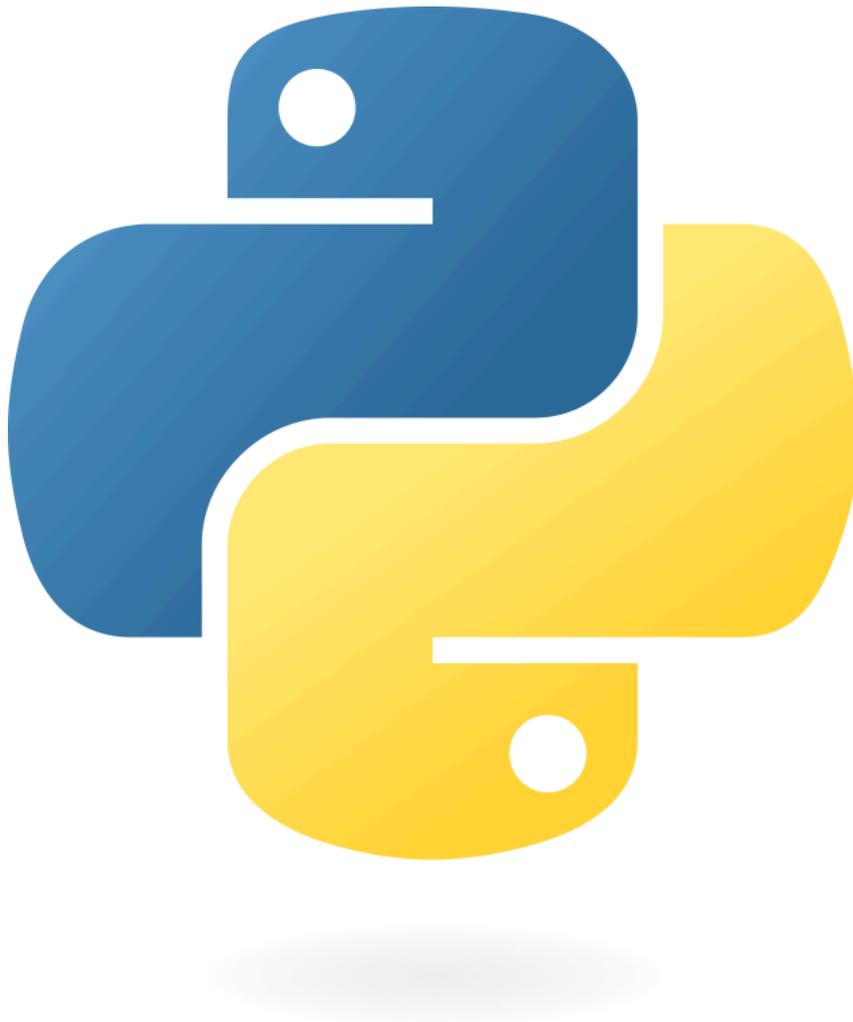


-

JavaScript Password Mask

Un esempio in JavaScript di mascheramento di una password con l'aggiunta della funzionalità di copia negli appunti.

Docker Docker Compose JavaScript



-

Python Placeholder Image

Applicazione sviluppata in Python con Flask per la creazione di immagini segnaposto.
Docker Docker Compose Python Flask JavaScript



-

Go Placeholder Image

Applicazione in Go per la creazione di immagini segnaposto.
Docker Docker Compose Go JavaScript