Data Type

String: chứa các ký tự. Ví dụ "hello"

Number: là số nguyên hoặc thập phân. Ví dụ 20, 32.46

Array: chứa nhiều phần tử, các phần tử có thể cùng hoặc không cùng kiểu dữ liệu

Object: gồm các cặp key-value

Boolean: true hoặc false

Null: thể hiện rằng biến đó không trỏ đến bất kỳ object nào

String

```
let s1 = "Hello World";
s1.length; // 11
s1.ind exO f("l "); // 2
s1.rep eat(4); // Hello WorldHello WorldHello
WorldHello World
s1.toL owe rCa se(); // hello world
s1.toU ppe rCa se(); //HELLO WORLD
s1.sli ce(0, s1.ind exO f("o ")); // Hell
let s2 = " 100.33 2";
let num1 = parseI nt(s2) + 10; // 110
let num2 = parseF loa t(s2) + 10; // 110.332
```

Number

```
let num1 = 100;
let num2 = 0.5;
let num3 = num1 + num2; // 100.5
let num4 = num1 - num2; // 99.5
let num5 = num1 * num2; // 50
let num6 = num1 / num2; // 200
let s1 = num1.t oSt ring();
consol e.l og( s1.r ep eat (3)); // 100100100
```

Object

```
const obj1 = {
        " dat a": {
                " nam e": " luc as",
                " age ": 20,
                " is_ adm in": true
```

Object (cont)

```
>    },
   "string": "Hi"
};
// Access value
let name = obj1.data.name;
let age = obj1.data["age"];
// Add key-value
obj1.new_key = "new key here";
// Edit value
obj1.data.is_admin = false;
// Delete key
delete obj1.string;
// Copy object
let obj2 = JSON.parse(JSON.stringify(obj1));
```

Array

```
const arr = ["english", "german", "french",
"spanish"];
// Get value
let first = arr[0];
let last = arr[ar r.l ength - 1];
// Add item
arr.pu sh( " las t_i tem _1", " las t_i tem _2");
//at the end
arr.un shi ft( " fir st_ ite m_1 ", " fir st_ ite -
m_2 "); //at the beginning
// Edit value
arr[4] = "edit value";
// Remove item
let anothe r_last = arr.pop(); //remove last item
of array
let anothe r_first = arr.sh ift(); //remove first
item of array
// Query index of value
let position = arr.in dex Of( " eng lis h");
// From position 0, delete 1 item then add 3 other
items
let s2 = arr.sp lice(0, 1, " tex t1", " tex t2",
" tex t3");
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Data format: Json

Cặp key-value
- Key là string
- Value có thể là array, object, string, number, boolean hoặc null
Không có dấu phảy cho item cuối cùng

Example

```
{
        " id": 1,
        " nam e": {
                " fir st_ nam e": " Ver non ",
                " las t_n ame ": " Har per "
       },
        " ema il": " tes t@g mai l.c om",
        " is_ adm in": true,
        " cou rse s": ["Ac cou nti ng", " Sta tis -
tic s"],
        " oth er_ inf o": null
}
```

JSON object

To convert Json <--> String

JSON.parse(text) ---> json
JSON.stringify(json) ---> string

Variables

Global: cho toàn app postman

Collection: cho những request nằm trong collection

Environment: cho những request nào sử dụng, không quan tâm collection

Data: khi đọc file data (csv hoặc json), chỉ có khi sử dụng Runner hoặc Newman

Local: Khi sử dụng postman script

Thứ tự ưu tiên sẽ là:
Local (mạnh nhất) -> data -> environment - > collection -> global (yếu nhất)
Note:
- object và array: trước khi set thành variable, cần phải **stringify** trước
- object và array: sau khi get từ variable, cần phải **parse** thì mới sử dụng được.

Variable examples

```
pm.globals.set('myVariable', MY_VALUE);
pm.glo bal s.g et( 'my Var iab le');
pm.glo bal s.u nse t(' myV ari able');
pm.glo bal s.c lear();
pm.col lec tio nVa ria ble s.s et( 'my Var iable',
MY_VALUE);
pm.col lec tio nVa ria ble s.g et( 'my Var iab -
le');
pm.col lec tio nVa ria ble s.u nse t(' myV ari -
able');
pm.col lec tio nVa ria ble s.c lear();
pm.env iro nme nt.s et ('m yVa ria ble', MY_VALUE);
pm.env iro nme nt.g et ('m yVa ria ble');
pm.env iro nme nt.u ns et( 'my Var iab le');
pm.env iro nme nt.c le ar();
pm.env iro nme nt.name //show name of enviro nment
pm.var iab les.se t(' myV ari able', MY_VALUE);
pm.var iab les.ge t(' myV ari able');
pm.var iab les.un set ('m yVa ria ble');
pm.var iab les.cl ear();
pm.ite rat ion Dat a.g et( 'my Var iab le');
pm.var iab les.re pla ceI n("m yVa ria ble ");
```

Useful methods

```
// Get (n) random items from an array
_.samp le( array, [n])
let course s_arr = ["pr ogr amm ing ", " mat h",
" eco nom ics ", " sta tis tic "];
let courses = _.samp le( cou rse s_arr, 2); //
["ma th", " eco nom ics "]
//Get random number in a range start -> end
_.rand om(0, 200) //120
// Duplicate (n) times:
let a = _.times(3, _.rand om(0, 10)); //[ 6, 7, 2
]
//Make random string:
Math.r and om( ).t oSt rin g(3 6).s ub str ing(2)
//3f4n 66lsrq7
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com