

**JSON**

- ♣ JavaScript Object Notation (JSON) er et data udvekslingsformat
- ♣ JSON er let at læse, da det er en delmængde af JavaScript
- ♣ JSON er ren tekst og dermed uafhængig programmeringssprog
- ♣ JSON er et udbredt og simpelt alternativ til XML

# JSON values

❖ JSON anvender flg. delmængde af JavaScript:

objekt → {...}

array → [...]

string → "..." (ikke '...')

number

boolean

null

❖ Property navne skal også skrives med "..."

❖ Funktioner, metoder, variable og andre udtryk er ikke tilladt

```
[
  {
    "id": 1,
    "name": "Leanne Graham",
    "username": "Bret",
    "email": "Sincere@april.biz",
    "address": {
      "street": "Kulas Light",
      "suite": "Apt. 556",
      "city": "Gwenborough",
      "zipcode": "92998-3874",
      "geo": {
        "lat": "-37.3159",
        "lng": "81.1496"
      }
    },
    "phone": "1-770-736-8031 x56442",
    "website": "hildegard.org",
    "company": {
      "name": "Romaguera-Crona",
      "catchPhrase": "Multi-layered client-server neural-net",
      "bs": "harness real-time e-markets"
    }
  },
  {
    "id": 2,
    "name": "Ervin Howell",
    ...
  },
  ...
]
```

# JSON metoder og Eksempel: json.js

- ❖ `JSON.parse(JSON)` returnerer et JavaScript objekt svarende til JSON tekststrengen
- ❖ `JSON.stringify(objekt)` serialiserer et JavaScript objekt til en JSON tekststreng

```
let json = '{"x":1,"a":[true, null, "test"]}';  
let objekt = JSON.parse(json);  
console.log('%o', objekt);  
// => { x: 1, a: [ true, null, 'test', [length]: 3 ] }
```

```
objekt = {x: 1, m: function() {}, a: [true, undefined, 'test']};  
json = JSON.stringify(objekt);  
console.log(json); // => '{"x":1,"a":[true,null,"test"]}'
```