**JSON**

*JSON* je skracenica od (*JavaScript Object Notation*) i on je oblik podatka koji je lako dijeljiv izmedju ralicitih programskih jezika. To je oblik teksta koji je lako čitljiv za ljude i sastoji se od para atributa i njegove vrijednosti. Internte media tip za JSON je *application/json,* a *file* ekstenzija je *.json .* JSON oblik podatka je orginalno stvorio Douglas Crockford.

Tipovi podataka koji se mogu čuvati i pisati u JSON formatu su brojevi (*number*),stim da JSON ne pravi razliku izmedju broja i broja sa decimalnim zarezom (*float*), nizove tekstualnih karaktera (*string*) koji su razdvojeni sa duplim navodnicima ( “ “ ) i koji podrzavaju sintaksu sa kosom linijom unazad (*backslash \* ) za izbjegavanje karaktera.

*Boolean* vrijednost istinito ili lazno (*true* ili *false*), JSON podrzava i niz (*array*) koji je uredjena lista,a koja se moze sastojati od više elemenata različith tipova podataka. Nizovi koriste četvrtaste zagrade ( [ ] ) i vrijednosti su razdovjite zarezom.

Objekti su neuredjene kolekcije koje se sastoje od parova imena i njihovih vrijednosti,imena su kljucevi niza u obliku stringa. Objekti se označavaju sa vitičastim zagradama ( { } ) i sa zarezom je razdvojit svaki par,dok su imena razdvojita sa dvotačkom ( : ) od svoje vrijednosti. *Null* je prazan tip podatka i označava se sa *null.*

Prazan prostor ili *whitespace* je dozvoljen i ignorise se oko i izmedju elemenata ali se ne ignorise unutar string imena ili vrijednosti. Medjutim samo četri specifična karaktera se smatraju za prazan prostor,a to su *space,* horizontalni *tab, line feed* i *carriage return.*

*JSON* ne podržava i ne dozvoljava bilo kakav komentar.

Primjer *JSON* objekta sa svim tipovima podataka.

{

"firstName": "John",

"lastName": "Smith",

"isAlive": **true**,

"age": 25,

"address": {

"streetAddress": "21 2nd Street",

"city": "New York",

"state": "NY",

"postalCode": "10021-3100"

},

"phoneNumbers": [

{

"type": "home",

"number": "212 555-1234"

},

{

"type": "office",

"number": "646 555-4567"

}

],

"children": [],

"spouse": **null**

}

Izvor Wikipedia <https://en.wikipedia.org/wiki/JSON#Example>

*JSON* dokumenti se mogu enkodirati sa *UTF-8, UTF-16* ili *UTF-32*, ali uobičajeni način enkodiranja je *UTF-8*. Ovi načini enkodiranja podrzavaju sve *Unicode* karaktere.

**CANVAS**

*Canvas* je novi *HTML5 tag* element koji omogućava dinamičko, skriptabilno prikazivanje 2D objekata i *bitmap* slika (definirani prostor na displeju i boja za svaki piksel ili bit). *Canvas* je inicijalno predstavljen od strane *Apple-a* u svrhu korištenja u njihovom *Mac OS X Webkit* komponenti 2004.god. pomoću koje su pokretali aplikacije kao što su *Dashboard* i *Safari* pretrživač.

*HTML <canvas>* element se koristi za crtanje grafike uz pomoc nekog skriptnog jezika,najčešće *JavaScript-a. Canvas* element je samo kontejner za grafiku. Da bi nesto nacrtali moramo koristiti neki skriptni jezik. *Canvas* ima nekoliko metoda za crtanje putanja, kvadrata, krugova, tekstova i dodavanja slika.

*Canvas* po uobičajenim postavkama nema granicu niti sadržaj, zato moramo uvijek navesti *id* atribut koji nam služi za referenciranje u nekom skriptnom jeziku i atribut širine i visine da definiramo veličinu *canvas*-a koje su po uobičajenim postavkama namještene na 200 piksela za širinu i 150 piksela za visinu. *Canvas* ima zapravo dvije veličine,prva je veličina samog *canvas* elementa,a druga veličina površine za crtanje. *CSS* atributi za širinu i visinu se odnose samo na veličinu *canvas* elementa, ali ne i na veličinu za crtanje.

*Canvas* primjer (crtanje crvenog pravougaonika)

**<canvas** id="example" width="200" height="200"**>**

This text is displayed if your browser does not support HTML5 Canvas.

**</canvas>**

**<script>**

**var** example = document.getElementById('example');

**var** context = example.getContext('2d');

context.fillStyle = 'red';

context.fillRect(30, 30, 50, 50);

**</script>**

Izvor Wikipedia <https://en.wikipedia.org/wiki/Canvas_element#Example>

*Canvas* primjer (crtanje linearnog gradijenta)

**<script>**

var c = document.getElementById("example");  
var ctx = c.getContext("2d");  
  
// Create gradient  
var grd = ctx.createLinearGradient(0,0,200,0);  
grd.addColorStop(0,"red");  
grd.addColorStop(1,"white");  
  
// Fill with gradient  
ctx.fillStyle = grd;  
ctx.fillRect(10,10,150,80);

**</script>**

Izvor w3schools <http://www.w3schools.com/html/html5_canvas.asp>

Za razliku od *SVG*-a koji je vektorski objekat i grafika, *canvas* seiscrtava momentalno u pretrživaču i po zvršetku iscrtavanja sistem “zaboravlja” crtež,tako da ako se promjeni pozicija ili nesto drugo,čitav crtež se mora izpočtka nacrtati.

***CSS* mjerne jedinice**

Trenutno postoji petnaest *CSS* mjernih jedinica koje podrzavaju pretrazivači. Ove mjeren jedinice mozemo podijeliti u dvije grupe,a to su apsolutne mijerne jedinice i relativne. U apsolutne mjerne jedinice ubrajamo:

* *Pixels (px)*
* *Inches (in)*
* *Centimeters (cm)*
* *Millimeters (mm)*
* *Points (pt)*
* *Picas (pc)*

Relativne mjeren jedinice su:

* *Percentages (%)*
* *Font-sizes (em & rem)*
* *Charachter-sizes (ex & ch)*
* *Viewport Dimensions (vw & vh)*
* *Viewport Max (vmax)*
* *Viewport Min (vmin)*

Relativne mjerene jedinice dobijaju svoju veličinu u zavisnosti od nečega drugog. Postoje tri stvari na osnovu kojih relativne jedinice baziraju svoju veličinu,a to su roditeljske dimenzije (%), trenutno deklarisani *font* atributi (em, rem, ex, ch) i velicina *viewport*-a.

Procenti (%)

<div class=”example”>

<div class=”parent”>

<div class=”child”></div>

</div>

</div>

<style>

.parent {

width: 400px;

height: 400px;

background: hsla(196, 100%, 55%, 0.3);

}

.child {

width: 50%;

height: 50%;

background: hsla(59, 100%, 55%, 0.3);

}

</style>

Postoje četri mjerene jedinice koje su bazirane na *viewport* veličini,a to *su vw (viewport width), vh (viewport height), vmin (viewport minimum), vmax (viewport maximum).*

*Viewport width (vw)* i *viewport height (vh)* dijele *viewport* tj. veličinu *display*-a na mrezu od 100x100 jedinica.

<div class=”example”>

<div class=”vw”></div>

</div>

<style>

.vw {

/\*Sa obrnutim vrjednostima dobijamo responsive\*/

width: 50vw; /\*50vh\*/

height: 50vh; /\*50vw\*/

background: hsla(196, 100%, 55%, 0.3);

}

</style>

*Viewport Max* i *Viewport Min* (oznacavaju viewport maximum ili minimum) , isto dobijaju svoje mijere kao vw ili vh i dijele velicinu displaya na 100x100 mrezu,ali referenciraju ili širinu ili visinu u zavisnosti šta je veće ili manje,i u zavisnosti šta od ovih jedinica koristimo.

<div class=”example”>

<div class=”vmin”></div>

</div>

<style>

.vmin {

width: 50vmin; /\*50vmax\*/

height: 400px;

background: hsla(196, 100%, 55%, 0.3);

}

</style>

Postoje četri *font* relativne mjerene jedinice za širinu,a to su *ex (“x” height)* i *ch (“0” width)* i *em (font-size)* i *rem (:root font-size)*.

Mjerna jedinica *ex* je baziran na visini ‘*x*’ karaktera od font porodice koju koristimo.

<div class=”example”>

<div class=”ex”>x 0</div>

</div>

<style>

.ex {

width: 100%;

height: 1ex;

background: hsla(196, 100%, 55%, 0.3);

}

</style>

Mjerna jedinica *ch* je bazirana na širini broja 0.

<div class=”example”>

<div class=”cx”>x 0</div>

</div>

<style>

.ch {

width: 100%;

height: 1ch;

background: hsla(196, 100%, 55%, 0.3);

}

</style>

Mjerene jedinice *em* i *rem* su bazirane na font veličini. *Em* se odnosi na velicinu fonta koju koristi roditeljski element.

<div class=”example”>

<div class=”em”></div>

</div>

<style>

.example {

font-size: 10px;

}

.em {

width: 1em;

height: 1em;

background: hsla(196, 100%, 55%, 0.3);

}

</style>

Mjerna jedinica *rem* je bazirana na velicini fonta korjenskog elementa.

<div class=”example”>

<div class=”rem”></div>

</div>

<style>

:root {

font-size: 10px;

}

.example {

font-size: 10px;

}

.rem {

width: 1rem;

height: 1rem;

background: hsla(196, 100%, 55%, 0.3);

}

</style>

Apsolutne mijerne jedinice su dobile svoje ime zato sto ne baziraju svoju veličinu u odnosu na roditeljski element,ili veličinu fonta,vec je baziraju na stvarni mjernim jedinicama (cm,mm,in itd.).

<div class=”example”>

<div class=”in”>3 Inches</div><br><br>

<div class=”cm”>3 Centimeters</div><br><br>

<div class=”mm”>3 Milimeters</div><br><br>

<div class=”pt”>3 Points</div><br><br>

<div class=”pc”>3 Picas</div><br><br>

<div class=”px”>3 Pixels</div><br><br>

</div>

<style>

.example > div {

padding: 20px 0;

font-size: 10px;

white-space: nowrap;

background: hsla(196, 100%, 55%, 0.3);

}

.in { width: 3in; }

.cm { width: 3cm; }

.mm { width: 3mm; }

.pt { width: 3pt; }

.pc { width: 3pc; }

.px {

width: 300px;

height: 300px;

}

</style>

Izvor Youtube https://www.youtube.com/watch?v=qrduUUdxBSY

**Boostrap Forms**

Pravljenje forme *u Bootstrap CSS front-end framework*-u je veoma lako uz vrlo jednostavan *HTML markup* i vec gotove golobalne predefinisane *bootstrap* klase. Postoje tri tipa boostrap formi,a to su:

1. vertikalna forma koja je (*default*) tj. uobicajena
2. horizontalna forma
3. *in-line* forma

Da bi napravili osnovnu ali i svaku drugu boostrap formu,moramo uraditi sledece

* Dodati *HTML* atribut *role* sa vrijednošću ‘form’ u *HTML form* tag *<form role=”form”>* (Ovaj atribut omogućava bolji pristup ljudima koji koriste *screen readers*)
* Okružiti tag *labels* i *input* (ulazna) polja forme u *div* tag sa klasom *‘.form-group’* (ovo je potrebno za optimalni razmak)
* Dodtai klasu *‘.from-control’* svim tekstualni ulaznim poljima forme *<textarea>, <input>, <select>*

1. Vertikalna ili osnovna forma

<form role="form">  
  <div class="form-group">  
    <label for="email">Email address:</label>  
    <input type="email" class="form-control" id="email">  
  </div>  
  <div class="form-group">  
    <label for="pwd">Password:</label>  
    <input type="password" class="form-control" id="pwd">  
  </div>  
  <div class="checkbox">  
    <label><input type="checkbox"> Remember me</label>  
  </div>  
  <button type="submit" class="btn btn-default">Submit</button>  
</form>

1. Horizontalna forma

Horizontalna forma izdvaja se od ostalih oblika forma,kako u veličini *HTML markup* tako i u prezentaciji forme. Da bi napravili formu koja koristi horizontalni izgled potrebno je uraditi sledeće:

* + Potrebno je dodati klasu *‘.form-horizontal’* u *form* element
  + Okružiti tag *labels* i *input* (ulazna) polja forme u *div* tag sa klasom ‘.form-group’
  + Dodati klasu *‘.control-label’* u *label* element

<form class="form-horizontal" role="form">  
  <div class="form-group">  
    <label class="control-label col-sm-2" for="email">Email:</label>  
    <div class="col-sm-10">  
      <input type="email" class="form-control" id="email" placeholder="Enter email">  
    </div>  
  </div>  
  <div class="form-group">  
    <label class="control-label col-sm-2" for="pwd">Password:</label>  
    <div class="col-sm-10">   
      <input type="password" class="form-control" id="pwd" placeholder="Enter password">  
    </div>  
  </div>  
  <div class="form-group">   
    <div class="col-sm-offset-2 col-sm-10">  
      <div class="checkbox">  
        <label><input type="checkbox"> Remember me</label>  
      </div>  
    </div>  
  </div>  
  <div class="form-group">   
    <div class="col-sm-offset-2 col-sm-10">  
      <button type="submit" class="btn btn-default">Submit</button>  
    </div>  
  </div>  
</form>

1. *Inline* Forma

Da bi kreirali formu gdje su svi elementi *inline*,lijevo poravnati i pored njih su imana ulaznih polja *label*. Ovo se samo odnosi na forme gdje je veličina *viewporta* tj. veličina *display*-a minimalno 768px široka. Da bi napravili *inline* formu potrebno je u *form* element dodati klasu *‘.form-inline’.*

<form class="form-inline" role="form">  
  <div class="form-group">  
    <label for="email">Email address:</label>  
    <input type="email" class="form-control" id="email">  
  </div>  
  <div class="form-group">  
    <label for="pwd">Password:</label>  
    <input type="password" class="form-control" id="pwd">  
  </div>  
  <div class="checkbox">  
    <label><input type="checkbox"> Remember me</label>  
  </div>  
  <button type="submit" class="btn btn-default">Submit</button>  
</form>

Izvor w3schools http://www.w3schools.com/bootstrap/bootstrap\_forms.asp