

Gabriel Sánchez Cano

Inleiding webprogrammeren

HTML 5 & CSS 3

```
835 function inspectPrefiltersOrTransports( structure, options, originalO
836
837     var inspected = {},
838         seekingTransport = ( structure === transports );
839
840     function inspect( dataType ) {
841         var selected;
842         inspected[ dataType ] = true;
843         jQuery.each( structure[ dataType ] || [], function( _, prefil
844             var dataTypeOrTransport = prefilterOrFactory( options, or
845                 if ( typeof dataTypeOrTransport === "string" &&
846                     !seekingTransport && !inspected[ dataTypeOrTransport
847
848                 options.dataTypes.unshift( dataTypeOrTransport );
849                 inspect( dataTypeOrTransport );
850                 return false;
851             } else if ( seekingTransport ) {
852                 return !( selected = dataTypeOrTransport );
853             }
854         } );
855     return selected;
856 }
857
858 return inspect( options.dataTypes[ 0 ] ) || !inspected[ "*" ] &&
859 }
860
861 function ajaxExtend( target, src ) {
862     var key, deep,
863         flatOptions = jQuery.ajaxSettings.flatOptions || {};
864
865     for ( key in src ) {
866         if ( src[ key ] !== undefined ) {
867             ( flatOptions[ key ] ? target : ( deep || ( deep = {} ) ) )
868         }
869     }
870
871     return target;
872 }
873
874 return target;
```

Brinkman Uitgeverij

MBO-ICT

Inhoud

- 1 Basis HTML5 en CSS3—1**
- 1.1 Inleiding 1
 - Evolutie van HTML* 1
 - Lay-out en design van een webpagina* 1
 - Van concept tot realisatie* 2
 - De index of homepagina* 2
 - Mappenstructuur van het project* 3
 - HTML-editors* 3
 - Geneste elementen* 5
 - Browsers* 6
 - Meta-elementen* 8
 - Semantische elementen* 10
 - Hiërarchie van elementen* 11
 - id of class?* 12
- 1.2 CSS3 13
 - Inleiding* 14
 - CSS-syntaxis* 14
 - Stijlen coderen* 17
 - <head>-stijlen* 17
 - In-line-stijlen* 18
 - De stylesheet* 22
- 1.3 Webkleuren 29
- 1.4 Navigatie 33
 - Codering van de navigatie* 33
 - Ontwerp van de navigatie* 36
- 1.5 Logo 42
 - Codering van het logo* 42
 - Design van het logo* 44
- 1.6 Hoofd-content 46
 - Design van de hoofd-content* 47
 - Browser-prefix* 48
- 1.7 Positionering 51
 - relatieve positie* 52
 - Lagen* 56
- 1.8 Codering en design van de footer 60
 - Symbolen in HTML* 60

- 1.9 De playlist-pagina 62
 - Codering van playlist-pagina* 63
 - Wat zijn events?* 65
- 1.10 De formulierpagina 76
- 1.11 De contactpagina 86
 - Codering van de contactpagina* 86
 - Design van de contactpagina* 87
- 1.12 Responsive design 88
 - Grid-frameworks* 89
 - Gridless* 89
 - Voorbeeld van niet vloeiend design* 89
 - Voorbeeld van vloeiend design* 90
 - Responsive browser-knop* 90
 - @media query* 91
- 1.13 Material design 95
 - Materiaaleigenschappen* 96
 - Materiaal met objecten* 97
 - Licht en schaduw van objecten* 98
 - Materiaalomvorming* 98
 - Beweging en betekenis* 99
 - Eigenschappen van objecten en schaduwen* 100
 - Hiërarchie van objecten* 101
 - Interactie* 101
 - Beweging* 101
- 1.14 Een material-design-sjabloon 104
 - Hoofdmenu* 105
 - Het header-element* 106
 - Translate3d* 109
 - Het ~-teken* 110
 - Tabs* 112
 - Codering van de cards* 114
- 1.15 Material-design-project 117
 - De website XtremTravel* 117
 - De welkomspagina* 118
 - De windsurfing-pagina* 118
 - De mountainclimbing-pagina* 119
 - De snorkelenpagina* 119

2 JavaScript—121

- 2.1 Inleiding JavaScript 121
 - JavaScript-console* 123
 - Taalcomponenten* 124
 - JavaScript-variabelen* 126
- 2.2 Datatypes 131
 - Het datatype string* 132
 - De methode typeof* 132
 - Het datatype: boolean* 133
 - Het datatype: number (integer)* 133
 - Het datatype: number (floating-point)* 133
 - parseInt()* 134
 - parseFloat()* 134
 - parseInt()* 135
 - toString()* 135
- 2.3 Het Array-object 137
 - Een Array-object declareren* 138
 - Array-elementen declareren* 138
 - Array-methodes* 141
 - Array-pointers* 142
- 2.4 Objecten 152
- 2.5 Het Date-object 156
 - Creëer een Date-object* 156
 - Creëer een Date-object met parameters* 157
 - Creëer een Date-object met string* 158
 - Creëer een Date-object met getallen* 159
- 2.6 Date-methodes 160
 - getFullYear* 161
- 2.7 Beslissingsstructuren 162
 - De opdracht if* 162
 - De clause else* 164
 - AND-vergelijking* 164
 - De clause else if* 166
 - De ternary-operator (?:)* 167
- 2.8 Stroomdiagrammen 169
- 2.9 Switch 171
- 2.10 Eigen functies 173
 - Functie of methode* 174
 - Externe functies* 176
 - De tag <script source>* 177
 - Functies dynamisch uitvoeren* 180
 - Scope van variabelen* 181
 - De array arguments[]* 182

- 2.11 Lussen 184
 - De for()-lus* 185
 - De functie charAt()* 187
- 2.12 De for(in)-lus 190
- 2.13 De while()-lus 193
 - De do-while()-lus* 196
- 2.14 Het Math-object 197
 - Math.abs(x)* 198
 - Math.ceil(x)* 198
 - Math.floor(x)* 198
 - Math.max()* 199
 - Math.min()* 199
 - Math.round()* 200
 - Math.random()* 200
- 2.15 String-methodes 201
 - indexOf()* 202
 - Length* 204
 - charAt()* 204
 - split()* 204
 - substring()* 205
 - substr()* 206
 - trim()* 206
 - toLowerCase()* en *toUpperCase()* 206
- 2.16 Algoritmes 207
 - Het algoritme binary-search* 208
 - Een voorbeeld* 208
 - Pseudocode* 209
- 2.17 Het Document Object Model 213
 - getElementById* 214
 - DOM-events (gebeurtenis)* 214
 - De functie setTimeout()* 215
 - De functie clearTimeout()* 216
 - Mouse-events* 217
 - Form-events* 218
 - Keyboard-events* 222
- 2.18 Cookies 223
 - Hoe werk je met cookies?* 224
 - JavaScript met cookies* 225
 - Browser en cookies* 228
- 2.19 Een winkelwagentje in JavaScript 229

- 3 OOP in JavaScript—235**
 - 3.1 Het UML class-diagram 235
 - Class met properties* 236
 - Class met methodes* 236
 - Subclass* 236
 - 3.2 Het factory pattern 237
 - Destructor* 239
 - Object-properties wijzigen* 239
 - 3.3 Constructor-pattern 240
 - 3.4 Prototype-pattern 242
 - Objecten kunnen berichten naar elkaar sturen* 243
 - 3.5 Inheritance (overerven) 244
 - Overschrijven van parent-methodes* 246
 - Delete-function* 247
 - 3.6 Een OOP videoPlayer 248
 - Rekenmachine algoritme* 252
 - 3.7 Speciale functions 253
 - Externe methodes aan objecten koppelen* 254
 - Call* 254
 - Apply* 256
 - Bind* 256
 - 3.8 Anonieme functions 259
 - Callback-functions* 259
 - Zelfuitvoerende functions* 260
 - Return-functions* 260
 - Function-closures* 261
 - Function-heredity* 262
 - 3.9 JSON-object-literals 264
 - Objecten als properties van het object window* 265
 - Geneste JSON-object-literals* 266
 - Chaining properties* 266
 - Literal-objecten bundelen* 267
 - 3.10 Lokale opslagcapaciteit (local storage) 268
 - Local storage lezen* 269
 - 3.11 OOP webshop 270
 - Herbruikbare code* 272

4 jQuery—277

- 4.1 Basis van mobiele applicaties 277
 - Wat is een app?* 278
 - Native apps* 278
 - Webapps* 278
 - Hybride apps* 279
 - Native apps, webapps of hybride apps?* 279
 - Frameworks voor mobiele applicaties* 279
 - Een lokale testomgeving opzetten* 280
 - De webserver Apache* 280
 - Opera Mobile Emulator* 281
 - De ontwikkelomgeving* 282
 - De webapp Hello World* 282
 - Het element <meta viewport>* 284
 - Content Delivery Network (CDN)* 284
 - Data-role* 284
- 4.2 Een testomgeving in de cloud opzetten 286
- 4.3 De webapp FlashTix 287
 - De homepage* 288
 - Functioneel ontwerp van de homepage* 288
 - Header van de homepage* 289
 - Content van de homepage* 291
 - Footer van de homepage* 292
 - Design van de homepage* 293
 - Het attribuut data-icon* 295
 - Het attribuut data-iconpos* 295
- 4.4 De lastminute-pagina 296
 - Functioneel ontwerp* 296
 - Codering van de lastminute-pagina* 297
 - Content van de lastminute-pagina* 298
- 4.5 De biopagina 300
 - Functioneel ontwerp* 300
 - Codering van de biopagina* 301
 - Het element <video>* 302
 - Het element <source>* 303
 - Mediabestandsformaten* 303
- 4.6 De bestelpagina 304
 - Functioneel ontwerp van de bestelpagina* 305
 - Codering van de bestelpagina* 305
 - Het attribuut input type="range"* 307
 - Data-role="fieldcontain"* 307
- 4.7 Je eigen lastminute-webapp 310
 - Functioneel ontwerp van de homepage* 311

- 4.8 jQuery programmeren 312
 - De jQuery-wrapper* 313
 - jQuery-chains* 313
 - Use cases van de homepage* 313
 - DOM-elementen laden* 314
 - Content toevoegen: prepend() en append()* 314
 - Verwijderen: remove()* 317
- 4.9 Events en event-handlers 318
 - De functie bind()* 319
 - Events inspecteren* 321
 - De functie unbind()* 322
 - De functie one()* 323
 - De functie live()* 323
 - De functie die()* 323
 - De functie trigger()* 323
- 4.10 Het canvas van HTML5 324
 - Het canvasraster* 325
 - De canvas-context* 325
 - Tekenen in het canvas* 325
 - De canvasfunctie gradient()* 326
 - De fillStyle* 326
 - De functie fillRect()* 327
 - De functie strokeRect()* 327
 - De functie clearRect()* 327
 - Canvas: font en shadow* 327
 - Bogen en cirkels* 328
 - Interactief tekenen* 328
 - Canvasanimaties* 329
- 4.11 Use cases van de lastminute-pagina 331
 - Child-selectors* 332
 - De functie addClass()* 333
 - De functie CSS()* 333
 - Eigenschap-selectors []* 333
 - De functie show()* 334
 - De functie hide()* 334
 - De wildcard-selectors van CSS* 335
 - De beginwaardeselector [^=]* 335
 - De eindwaardeselector [\$=]* 336
 - De wildcard-selector [*=]* 336
 - De functie size()* 336
 - De functie first()* 336
 - De functie last()* 336
 - De functie html()* 336
 - De functie html('content')* 337

- Elementen verplaatsen* 337
- De functies after() en before()* 337
- De each (iterator)* 338
- De functie insertBefore()* 338
- 4.12 Use cases van de biopagina 339
 - data-role= "tabs"* 340
 - data-role= "navbar"* 340
 - class= "ui-body-d"* 340
 - De zoek-bar* 341
 - jQuery-plug-ins* 341
- 4.13 Use cases van de bestelpagina 343
 - Wat is AJAX?* 343
 - De functie \$.ajax()* 344
 - Het server-script* 346
- 4.14 De e-ticketspagina 346
 - Use cases van de e-ticketspagina* 348
- 4.15 De agendapagina 348
 - Functioneel ontwerp van de agendapagina* 349
 - Use cases van de agendapagina* 350
 - Codering van de agendapagina* 350
 - De knop Item toevoegen* 350
 - De knop Selecteren* 352
 - Local storage* 353
 - De agenda opnieuw laden* 354
 - De knop Verwijderen* 354
- 4.16 De locatiepagina 355
 - Codering van de locatiepagina* 356
 - Header en footer van de locatiepagina* 357
 - Use cases van de locatiepagina* 357
 - Content van de locatiepagina* 358
 - Google-markers* 359