**WebSite Name: Car House.**

Summary:

Car house website is all about cars and its showroom located in Boston. This website makes it easy for the customers to order car online which would be delivered at their door easily in redefined timeline. This website has various features like add to cart, videos are attached, signup form, user validation for login,etc.

Technologies used:

Html5

CSS3

Bootstrap

Javascript

JQuery

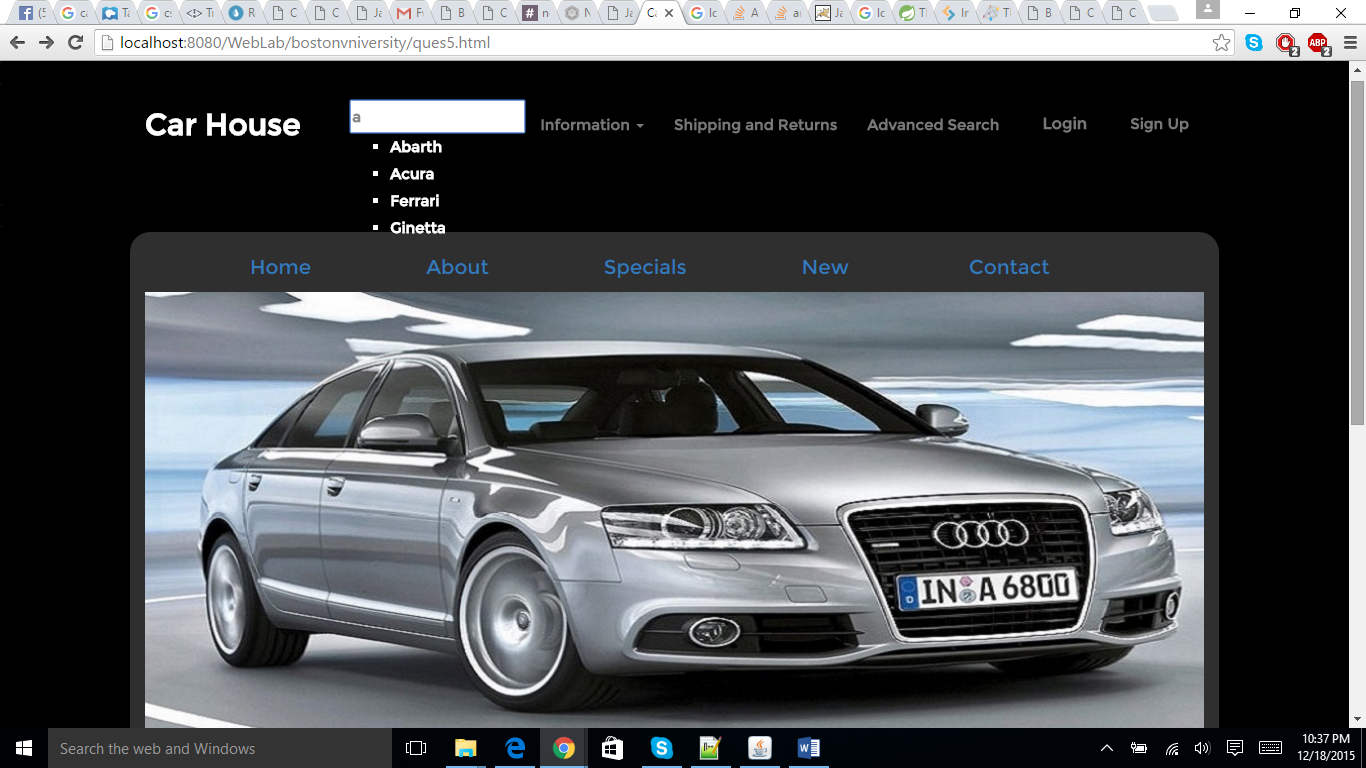
AngularJS

AJAX

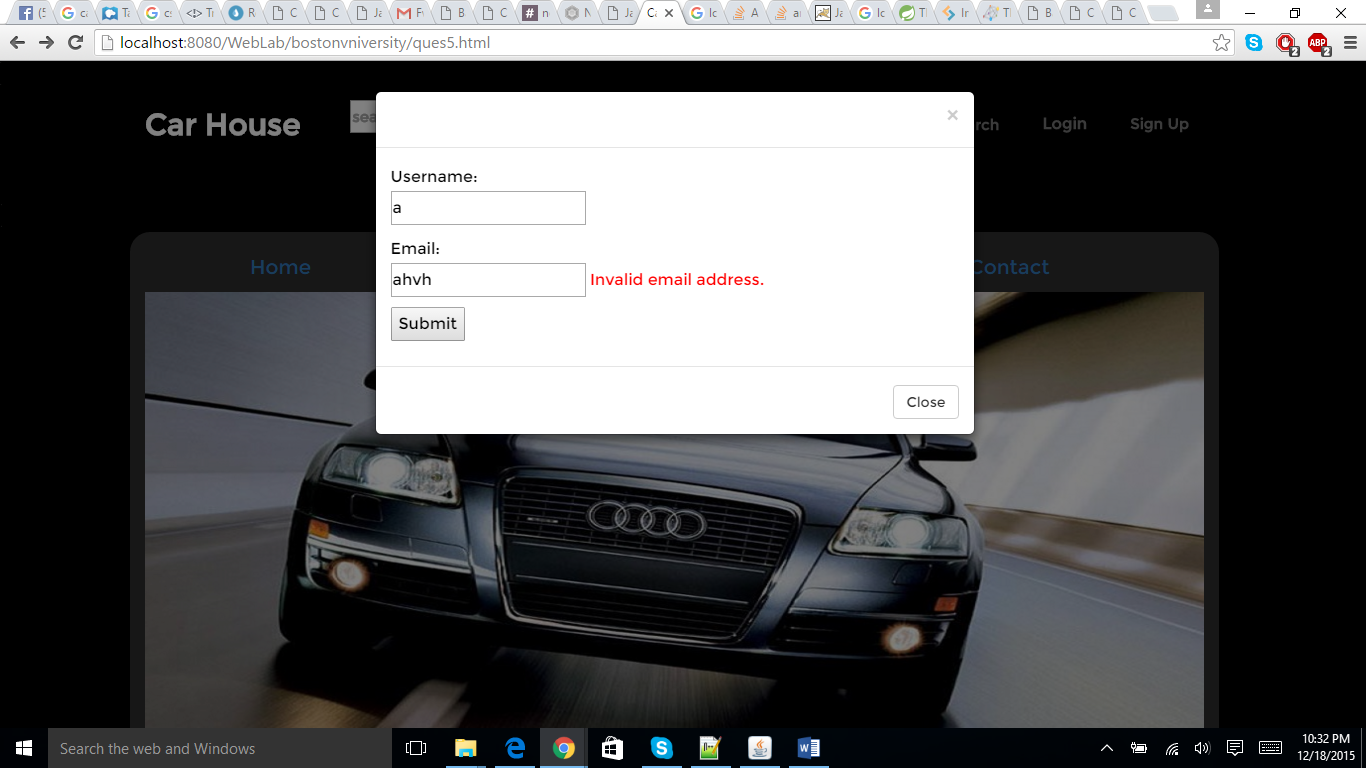
Screenshots:

1: Search box using AJAX

2: carousel for car pictures.

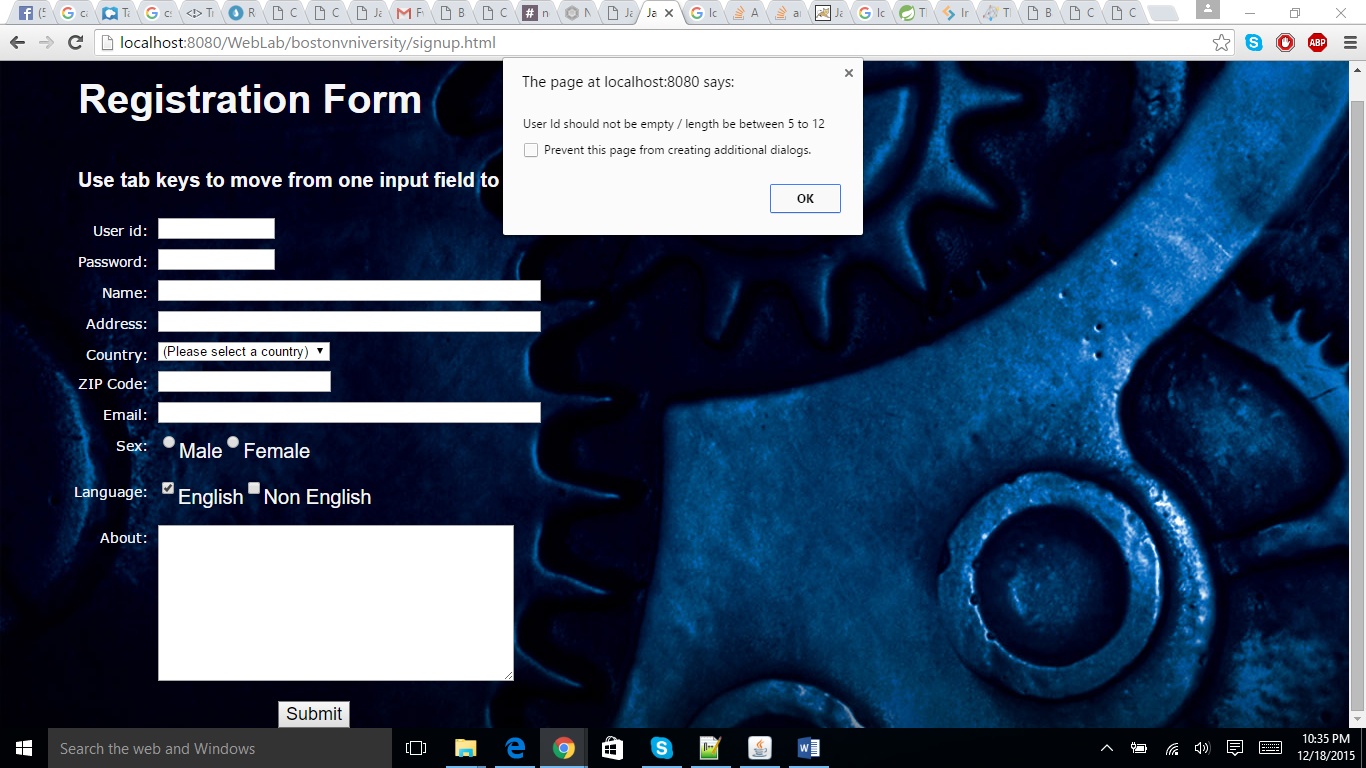


1:Login using AngularJS



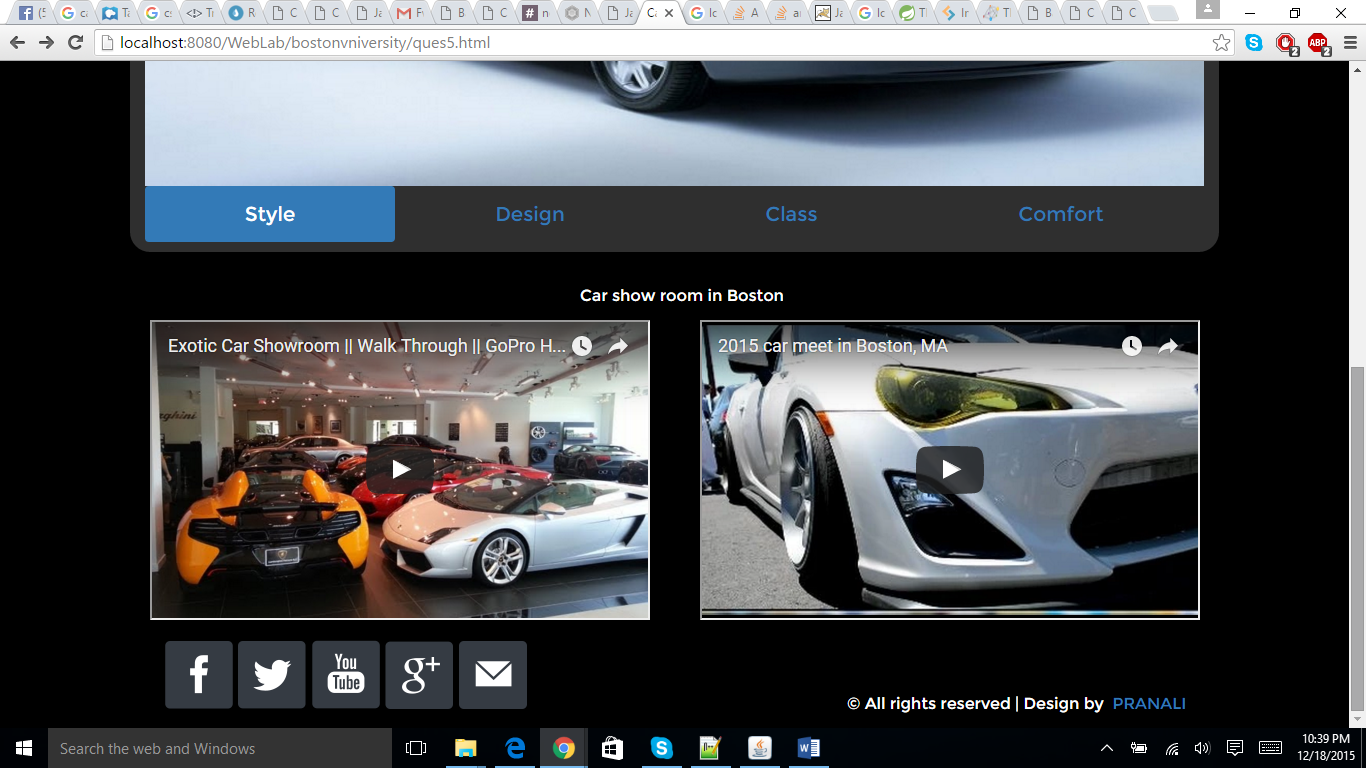
1:Form validation using JavaScript.

2:Background image attached



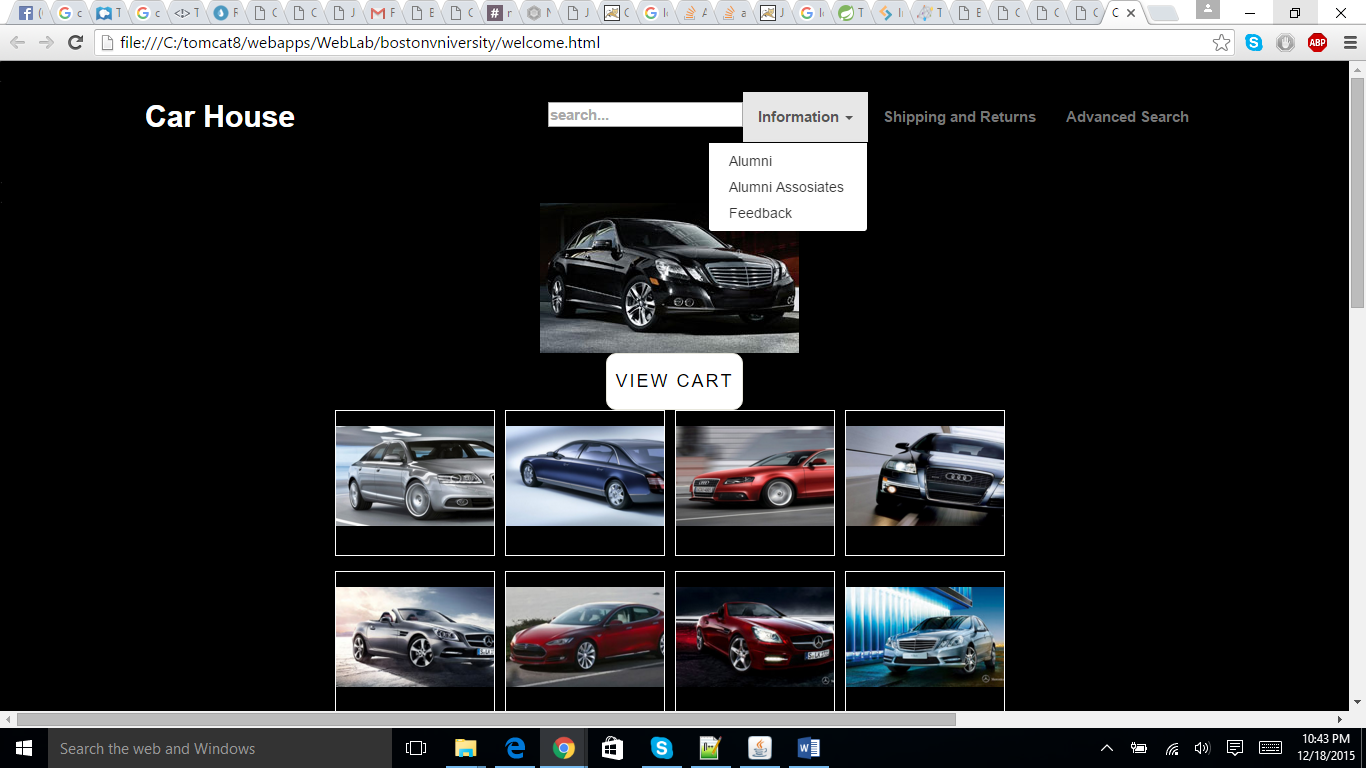
1: videos attached using HTML5

2: Connected to different media using different images.

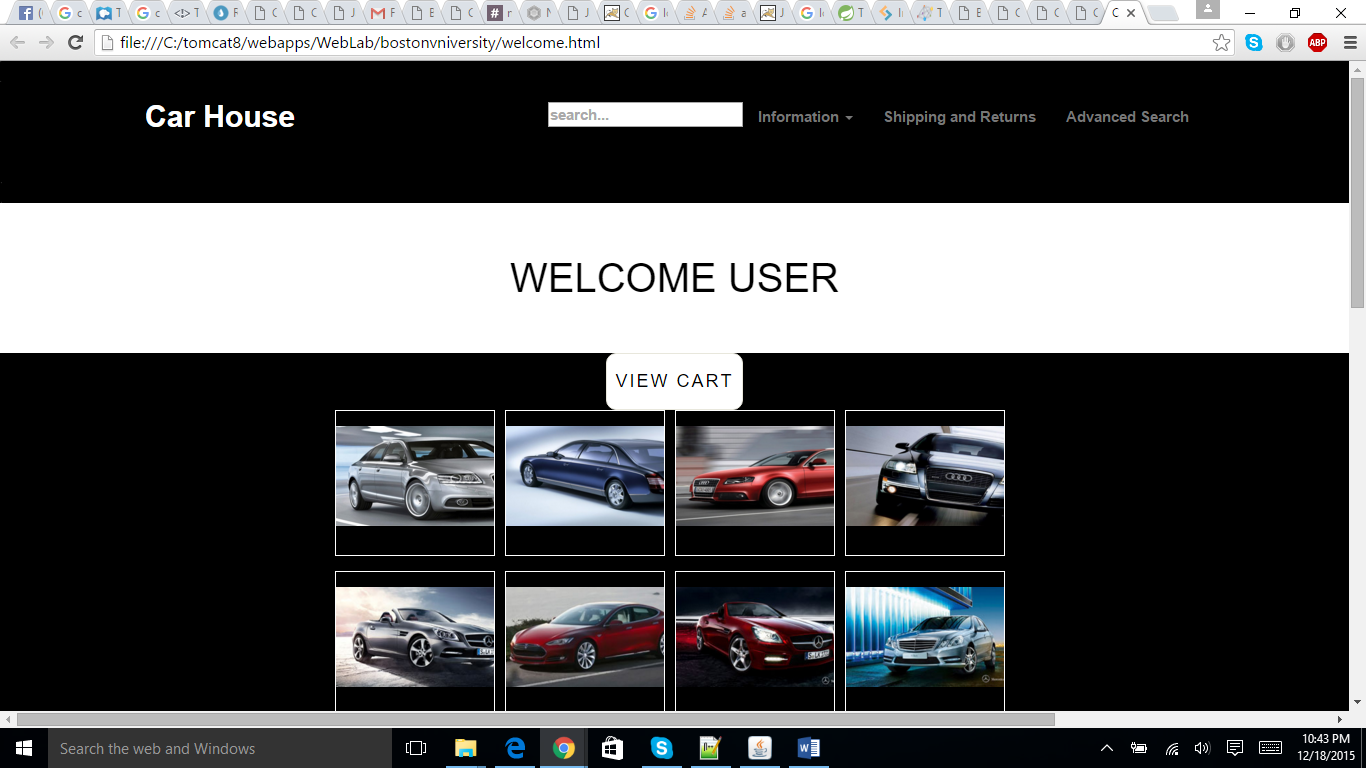


1:Information tab populated using dropdown.

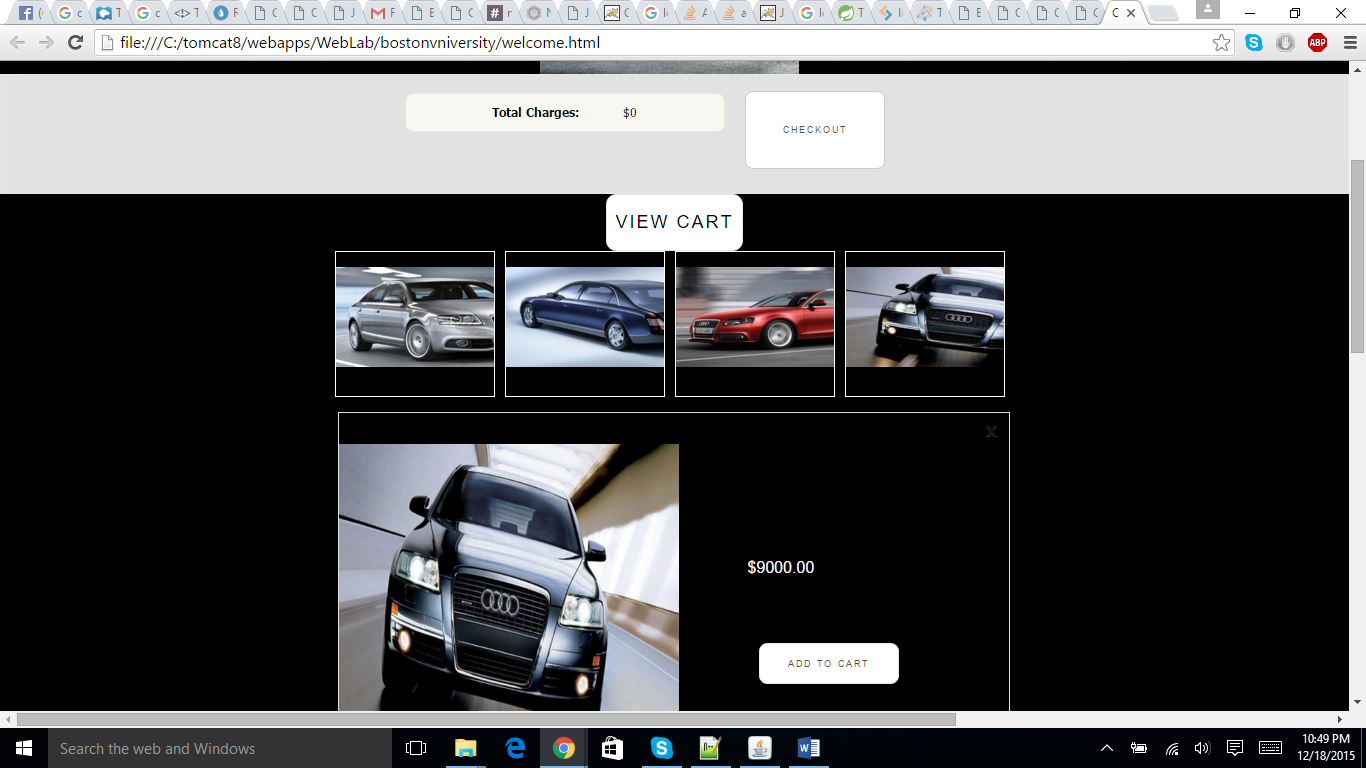
2: animation added to the car using css3(before)



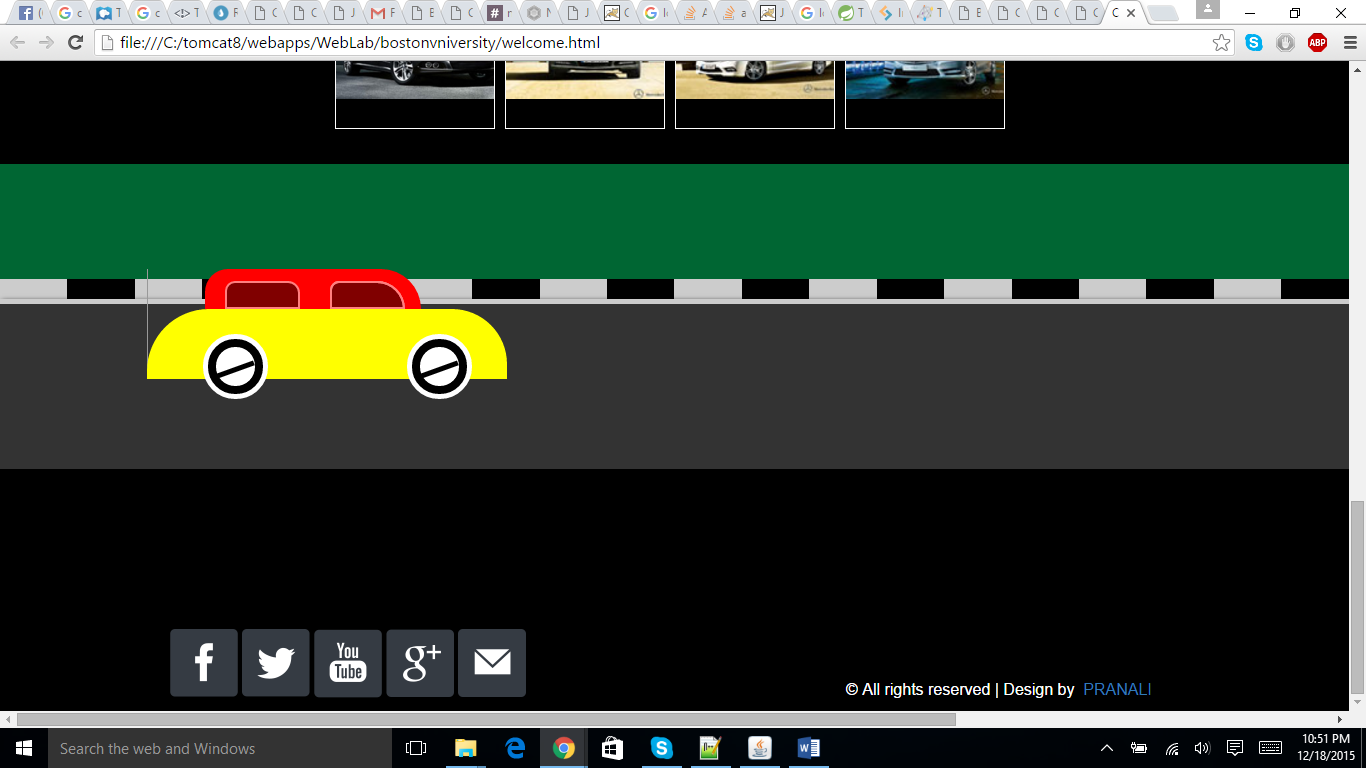
1: Welcome user comes when car is hovered(after)



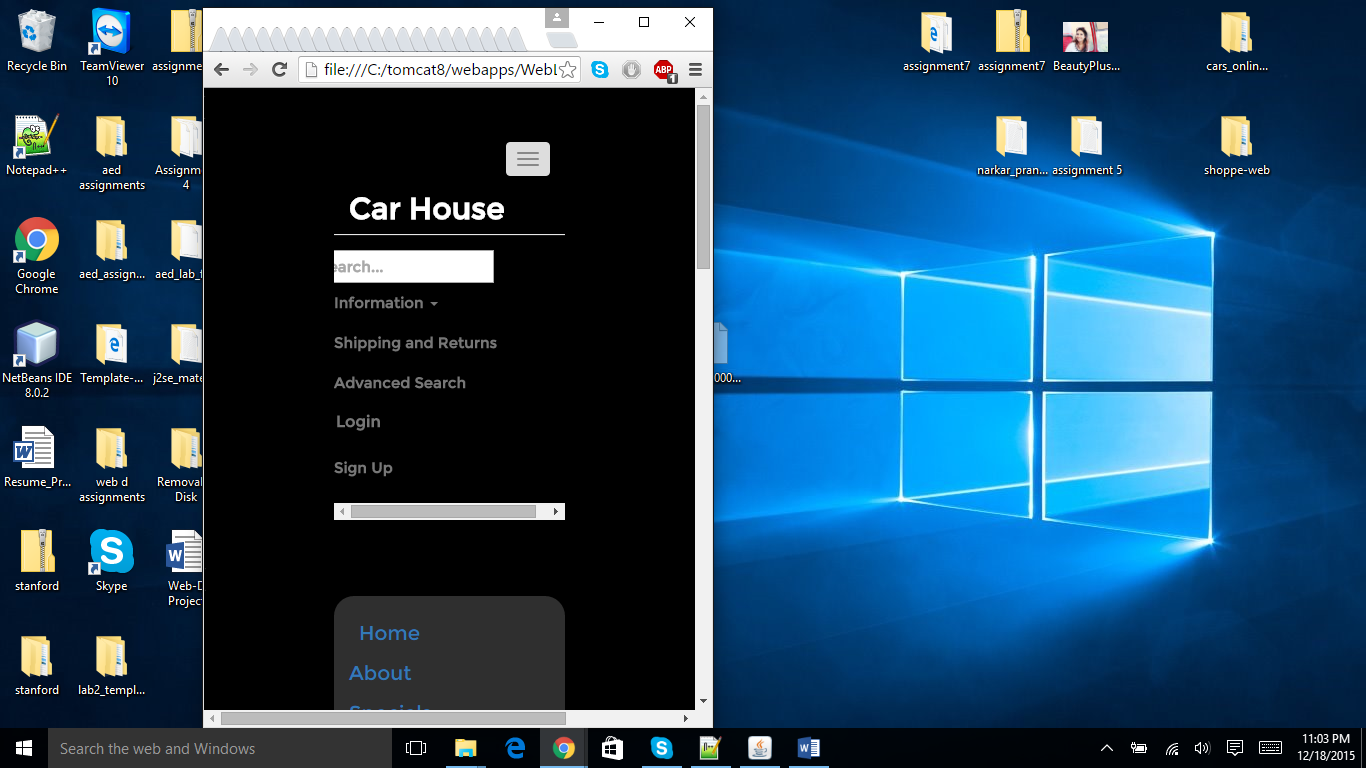
1:Add to cart feature using JQuery

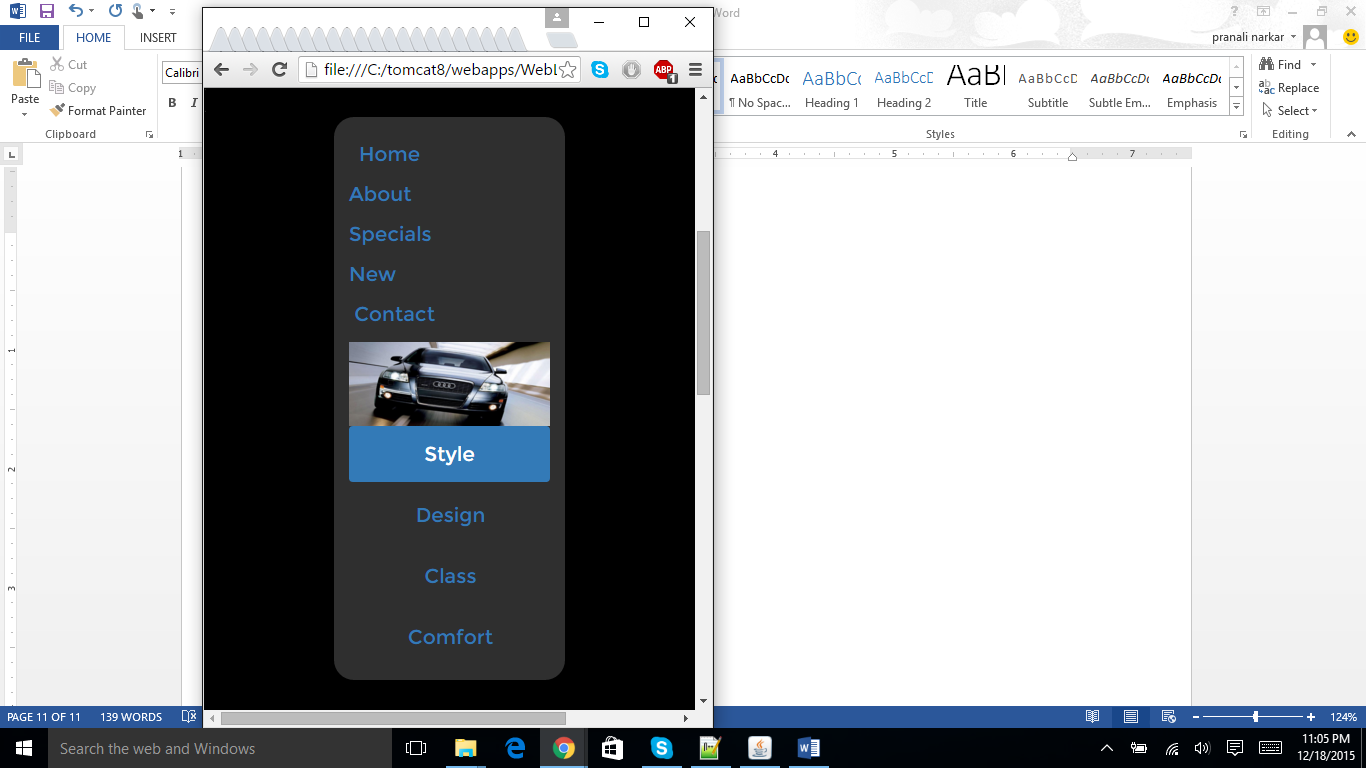


1:Animation added using html5 and css3



1:Page is made responsive.





Cars.html

<!DOCTYPE html>

<html lang="en">

<head>

<title>Car house</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href='http://fonts.googleapis.com/css?family=Work+Sans:400,600,700' rel='stylesheet' type='text/css'>

<meta name="keywords" content="example, JavaScript Form Validation, Sample registration form"/>

<meta name="description" content="This document is an example of JavaScript Form Validation using a sample registration form."/>

<link rel='stylesheet' href='js-form-validation.css' type='text/css' />

<script src="sample-registration-form-validation.js"></script>

<link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css">

<link href="http://fonts.googleapis.com/css?family=Montserrat" rel="stylesheet">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>

<script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script>

<script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>

<style>

body {

font: 20px Montserrat, sans-serif;

line-height: 1.8;

color: #ffffff;

background-color :#000000;

}

p {font-size: 16px;

color:#ffffff;}

.margin {margin-bottom: 45px;}

.bg-1 {

background-color: #1abc9c;

color: blue;

}

.bg-2 {

background-color: #474e5d;

color: blue;

}

.bg-3 {

background-color: #ffffff;

color: blue;

}

.bg-4 {

background-color: #2f2f2f;

color: blue;

}

.hero-feature {

margin-bottom: 30px;

}

#myNavbar{color:#888888;}

.container{

margin-left : 150px;

margin-right : 150px;

float : left;

}

.container-fluid {

padding-top: 10px;

padding-bottom: 10px;

margin-left:130px;

margin-right:130px;

margin-bottom :30px;

border-radius : 1em;

}

.navbar {

padding-top: 0;

padding-bottom: 0;

border: 2px solid black;

border-radius: 30px;

margin-bottom: 0;

font-size:15px;

font-weight : bold;

display : inline;

}

.navbar-nav li a:hover {

color: #1abc9c !important;

}

.thumbnail > img{

height :150px;

width : 1000px;

}

.caption{

color : red;}

p{

color : black;

}

.img-rounded{

border-radius : 1em;

}

.footer-right{

float : right;

color : gray;

float :center;

margin-bottom:100px;

margin-top:-25px}

#videosList {

max-width: 1400px;

margin: 0 auto;

padding: 0px 80px; }

@media (max-width: 1250px) {

#videosList {

padding: 0px 50px; } }

@media (max-width: 900px) {

#videosList {

padding: 60px 20px; } }

#videosList .video {

width: 50%;

display: inline-block;

float: left;

position: relative;

overflow: hidden; }

@media (max-width: 500px) {

#videosList .video {

width: 100%; } }

#videosList .video .videoSlate {

width: 100%;

height: 0;

padding: 60% 0 0 0;

-webkit-transition: 5000ms 50ms;

-moz-transition: 5000ms 50ms;

transition: 5000ms 50ms; }

#videosList .video .videoSlate:after {

content: ' ';

position: absolute;

top: 0;

left: 0;

display: block;

width: 100%;

height: 100%;

background: rgba(0, 0, 0, 0.3);

-webkit-transition: 500ms 50ms;

-moz-transition: 500ms 50ms;

transition: 500ms 50ms; }

#videosList .video .videoSlate video {

top: 0;

left: 0;

width: 100%;

height: 100%;

position: absolute; }

#videosList .video .videoListCopy {

display: inline-block;

text-align: center;

width: 100%;

z-index: 20; }

</style>

</head>

<body>

<nav class="navbar navbar-default">

<div class="container-fluid" ">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="#" style="color: #ffffff; font-size : 30px; padding:-5px">Car House</a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<div>

<ul class="nav navbar-nav" style="list-style:none;">

<li class="height"> <form>

<input type="text" onKeyUp="ajaxFunction(this.value)" placeholder="search..." />

<div id="cars" style="position:absolute; color:#ffffff; list-style:none" ></div>

</form></li>

<li class="dropdown"><a class="dropdown-toggle" data-toggle="dropdown" href="#">Information <span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a href="#">Accessories</a></li>

<li><a href="#">Models</a></li>

</ul>

</li>

<li><a href="#">Shipping and Returns</a></li>

<li><a href="#">Advanced Search</a></li>

<li id="listed" style="margin-top:-9.7888px; margin-left:-11px" ><a href="#">

<button type="button" style="background-color:black ;font-weight:bold; font-size:1.07em;" class="btn btn-large" data-toggle="modal" data-target="#myModalSignup">Login</button></a>

</li>

<li style="margin-top:-1.5px;font-weight:bold;"><a href="signup.html">Sign Up</a></li>

</ul>

</nav>

<div class="modal fade" id="myModalSignup" tabindex="-1" role="dialog" aria-labelledby="myModalLabel">

<div class="modal-dialog" role="document">

<div class="modal-content">

<div class="modal-header">

<button type="button" class="close" data-dismiss="modal">&times;</button>

<h4 class="modal-title">Change the number</h4>

</div>

<div class="modal-body">

<form ng-app="myApp" name="myForm" novalidate>

<p>Username:<br>

<input type="text" name="user" ng-model="user" id="name" required>

<span style="color:red" ng-show="myForm.user.$dirty && myForm.user.$invalid">

<span ng-show="myForm.user.$error.required">Username is required.</span>

</span>

</p>

<p>Email:<br>

<input type="email" name="email" ng-model="email" id="email" required>

<span style="color:red" ng-show="myForm.email.$dirty && myForm.email.$invalid">

<span ng-show="myForm.email.$error.email">Invalid email address.</span>

</span>

</p>

<p>

<input type="submit"

ng-disabled="myForm.user.$dirty && myForm.user.$invalid ||

myForm.email.$dirty && myForm.email.$invalid">

</p>

</form>

</div>

<!--angular validation ends here-->

<div class="modal-footer">

<button type="button" class="btn btn-default" data-dismiss="modal">Close</button>

</div>

</div>

</div>

</div>

<div class="container-fluid bg-4 ">

<ul class="nav navbar-nav" style="padding-top : -50px; text-align: center; margin-right : 100px; margin top: -50px;">

<li><a href="#" style="margin-right:85px; margin-left:10px">Home</a></li>

<li><a href="#" style="margin-right:130px;">About</a></li>

<li><a href="#" style="margin-right:110px;">Specials</a></li>

<li><a href="#"style="margin-right:120px;">New</a></li>

<li><a href="#" style=" float:right;">Contact</a></li>

</ul>

<div id="carousel1" class="carousel slide" data-ride="carousel">

<div class="carousel-inner">

<div class="item active">

<img alt="Bootstrap template" src="banner4.jpg">

<div class="carousel-caption">

</div>

</div>

<div class="item">

<img alt="Bootstrap template" src="banner1.jpg">

<div class="carousel-caption">

<div class="col-lg-12 text-center v-center" style="font-size: 39pt;">

<a href="#"><span class="avatar"><i class="fa fa-google-plus"></i></span></a>

<a href="#"><span class="avatar"><i class="fa fa-linkedin"></i></span></a>

<a href="#"><span class="avatar"><i class="fa fa-facebook"></i></span></a>

<a href="#"><span class="avatar"><i class="fa fa-github"></i></span></a>

</div>

</div>

</div>

<div class="item">

<img alt="Bootstrap template" src="banner2.jpg">

<div class="carousel-caption">

</div>

</div>

<div class="item">

<img alt="Bootstrap template" src="banner3.jpg">

<div class="carousel-caption">

</div>

</div>

</div>

<ul class="nav nav-pills nav-justified">

<li data-target="#carousel1" data-slide-to="0" class="active">

<a href="#">Style</a>

</li>

<li data-target="#carousel1" data-slide-to="1">

<a href="#">Design</a>

</li>

<li data-target="#carousel1" data-slide-to="2">

<a href="#">Class</a>

</li>

<li data-target="#carousel1" data-slide-to="3">

<a href="#">Comfort</a>

</li>

</ul>

</div>

</div>

<p style="margin-left:580px; color:#ffffff;">Car show room in Boston</p>

<section><iframe width="500" height="300" src="https://www.youtube.com/embed/4aC9tUuRCrM" style="margin-left : 150px; float:left; frameborder="0" allowfullscreen="" float:left;></iframe>

<iframe width="500" height="300" src="https://www.youtube.com/embed/4fqAKeMEGfw" style="margin-left : 50px; float:left; frameborder="0" allowfullscreen="" float:left;></iframe>

<a href="https://www.youtube.com/embed/XrlRy4r\_b3U" target="\_blank" class="btn btn-1"></a> </section>

<div class="footer">

<div class="container">

<div class="socialize" style="float:left ; margin-top:20px">

<div>

<a href="https://www.facebook.com/pranali.narkar"><img src="images/fb.png" width="68" height="68" class="facebook" alt="fb" onmouseover=hover(this) onmouseout=unhover(this)></a>

<a href="#"><img src="images/twitter.png" width="68" height="68" class="twitter" alt="twitter" onmouseover=hover(this) onmouseout=unhover(this)></a>

<a href="#"><img src="images/youtube.png" width="68" height="69" class="youtube" alt="youtube" onmouseover=hover(this) onmouseout=unhover(this)></a>

<a href="#"><img src="images/g+.png" width="68" height="68" class="google" alt="g+" onmouseover=hover(this) onmouseout=unhover(this)></a>

<a href="mailto:pranali.narkar017@gmail.com"><img src="images/message.png" width="68" height="68" class="message" alt="message" onmouseover=hover(this) onmouseout=unhover(this)></a>

</div>

<!--end of social imgs-->

<script>

function hover(element) {

var s = (element.src).replace(".png", "hover.png");

element.setAttribute("src", s);}

function unhover(element) {

var s = (element.src).replace("hover.png", ".png");

element.setAttribute("src", s);

}

</script>

</div>

<div class="copy" style="float:left ; margin-top:70px; margin-left:320px">

<p class="link" style="color:#ffffff;">&copy; All rights reserved | Design by&nbsp; <a href="#">PRANALI</a></p>

</div>

</div>

</div>

<script>

var app = angular.module('myApp', []);

var INTEGER\_REGEXP = /^\-?\d+$/;

app.directive('integer', function() {

return {

require: 'ngModel',

link: function(scope, elm, attrs, ctrl) {

ctrl.$validators.integer = function(modelValue, viewValue) {

if (ctrl.$isEmpty(modelValue)) {

// consider empty models to be valid

return true;

}

if (INTEGER\_REGEXP.test(viewValue)) {

// it is valid

return true;

}

// it is invalid

return false;

};

}

};

});

app.directive('username', function($q, $timeout) {

return {

require: 'ngModel',

link: function(scope, elm, attrs, ctrl) {

var usernames = ['Jim', 'John', 'Jill', 'Jackie'];

ctrl.$asyncValidators.username = function(modelValue, viewValue) {

if (ctrl.$isEmpty(modelValue)) {

// consider empty model valid

return $q.when();

}

var def = $q.defer();

$timeout(function() {

// Mock a delayed response

if (usernames.indexOf(modelValue) === -1) {

// The username is available

def.resolve();

} else {

def.reject();

}

}, 2000);

return def.promise;

};

}

};

});

</script>

<script type="text/javascript">

function ajaxFunction(cars) //functionName could be anything

{

//1. Create XmlHttpRequest Object

//This could be written into an external .js file that could be used within other pages as well.

var xmlHttp;

try // Firefox, Opera 8.0+, Safari

{

xmlHttp=new XMLHttpRequest();

}

catch (e)

{

try // Internet Explorer

{

xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");

}

catch (e)

{

try

{

xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");

}

catch (e)

{

alert("Your browser does not support AJAX!");

return false;

}

}

}

//2. Call the server-side script

//cars is the function parameter passed when the JS function is called

xmlHttp.open("GET", "cars.jsp?q=" + cars, true); //q is the name of the parameter to be used in JSP

xmlHttp.send(); //leave blank, or pass null. Not used with GET requests. when using post we need to pass the parameters

//3. Check the server-data is ready

xmlHttp.onreadystatechange=function()

{

if(xmlHttp.readyState==4)

{

//4. Manipulate the DOM

var cars = document.getElementById("cars");

var serverData = xmlHttp.responseText;

cars.innerHTML = serverData;

}

}

}

</script>

</body>

</html>

Signup.html

<!DOCTYPE html>

<html lang="en"><head>

<meta charset="utf-8">

<title>JavaScript Form Validation using a sample registration form</title>

<meta name="keywords" content="example, JavaScript Form Validation, Sample registration form" />

<meta name="description" content="This document is an example of JavaScript Form Validation using a sample registration form. " />

<link rel='stylesheet' href='js-form-validation.css' type='text/css' />

<script src="sample-registration-form-validation.js"></script>

<style>

body {

font: 20px Montserrat, sans-serif;

line-height: 1.8;

color: #f5f6f7;

background-color :#000000;

background-image: url("http://rpgmaker.net/media/content/users/4854/locker/NLBackground.jpg");

}

h1 {

margin-left: 70px;

}

form li {

list-style: none;

margin-bottom: 5px;

}

form ul li label{

float: left;

clear: left;

width: 100px;

text-align: right;

margin-right: 10px;

font-family:Verdana, Arial, Helvetica, sans-serif;

font-size:14px;

}

form ul li input, select, span {

float: left;

margin-bottom: 10px;

}

form textarea {

float: left;

width: 350px;

height: 150px;

}

[type="submit"] {

clear: left;

margin: 20px 0 0 230px;

font-size:18px

}

p {

margin-left: 70px;

font-weight: bold;

}

</style>

</head>

<body onload="document.registration.userid.focus();">

<h1>Registration Form</h1>

<p>Use tab keys to move from one input field to the next.</p>

<form name='registration' onSubmit="return formValidation();">

<ul style="text-align:centre;">

<li><label for="userid">User id:</label></li>

<li><input type="text" name="userid" size="12" /></li>

<li><label for="passid">Password:</label></li>

<li><input type="password" name="passid" size="12" /></li>

<li><label for="username">Name:</label></li>

<li><input type="text" name="username" size="50" /></li>

<li><label for="address">Address:</label></li>

<li><input type="text" name="address" size="50" /></li>

<li><label for="country">Country:</label></li>

<li><select name="country">

<option selected="" value="Default">(Please select a country)</option>

<option value="AF">Australia</option>

<option value="AL">Canada</option>

<option value="DZ">India</option>

<option value="AS">Russia</option>

<option value="AD">USA</option>

</select></li>

<li><label for="zip">ZIP Code:</label></li>

<li><input type="text" name="zip" /></li>

<li><label for="email">Email:</label></li>

<li><input type="text" name="email" size="50" /></li>

<li><label id="gender">Sex:</label></li>

<li><input type="radio" name="sex" value="Male" /><span>Male</span></li>

<li><input type="radio" name="sex" value="Female" /><span>Female</span></li>

<li><label>Language:</label></li>

<li><input type="checkbox" name="lan" value="en" checked /><span>English</span></li>

<li><input type="checkbox" name="lan" value="noen" /><span>Non English</span></li>

<li><label for="desc">About:</label></li>

<li><textarea name="desc" id="desc"></textarea></li>

<li><input type="submit" name="submit" value="Submit" /></li>

</ul>

</form>

<script>

function formValidation()

{

var uid = document.registration.userid;

var passid = document.registration.passid;

var uname = document.registration.username;

var uadd = document.registration.address;

var ucountry = document.registration.country;

var uzip = document.registration.zip;

var uemail = document.registration.email;

var umsex = document.registration.msex;

var ufsex = document.registration.fsex; if(userid\_validation(uid,5,12))

{

if(passid\_validation(passid,7,12))

{

if(allLetter(uname))

{

if(alphanumeric(uadd))

{

if(countryselect(ucountry))

{

if(allnumeric(uzip))

{

if(ValidateEmail(uemail))

{

if(validsex(umsex,ufsex))

{

}

}

}

}

}

}

}

}

return false;

} function userid\_validation(uid,mx,my)

{

var uid\_len = uid.value.length;

if (uid\_len == 0 || uid\_len >= my || uid\_len < mx)

{

alert("User Id should not be empty / length be between "+mx+" to "+my);

uid.focus();

return false;

}

return true;

}

function passid\_validation(passid,mx,my)

{

var passid\_len = passid.value.length;

if (passid\_len == 0 ||passid\_len >= my || passid\_len < mx)

{

alert("Password should not be empty / length be between "+mx+" to "+my);

passid.focus();

return false;

}

return true;

}

function allLetter(uname)

{

var letters = /^[A-Za-z]+$/;

if(uname.value.match(letters))

{

return true;

}

else

{

alert('Username must have alphabet characters only');

uname.focus();

return false;

}

}

function alphanumeric(uadd)

{

var letters = /[0-9a-zA-Z]+$/;

if(uadd.value.match(letters))

{

return true;

}

else

{

alert('User address must have alphanumeric characters only');

uadd.focus();

return false;

}

}

function countryselect(ucountry)

{

if(ucountry.value == "Default")

{

alert('Select your country from the list');

ucountry.focus();

return false;

}

else

{

return true;

}

}

function allnumeric(uzip)

{

var numbers = /^[0-9]+$/;

if(uzip.value.match(numbers))

{

return true;

}

else

{

alert('ZIP code must have numeric characters only');

uzip.focus();

return false;

}

}

function ValidateEmail(uemail)

{

var mailformat = /^\w+([\.-]?\w+)\*@\w+([\.-]?\w+)\*(\.\w{2,3})+$/;

if(uemail.value.match(mailformat))

{

return true;

}

else

{

alert("You have entered an invalid email address!");

uemail.focus();

return false;

}

} function validsex(umsex,ufsex)

{

x=0;

if(umsex.checked)

{

x++;

} if(ufsex.checked)

{

x++;

}

if(x==0)

{

alert('Select Male/Female');

umsex.focus();

return false;

}

else

{

alert('Form Succesfully Submitted');

window.location.reload()

window.location.href = "welcome.html";

return true;

}

}

</script>

</body>

</html>

Welcome.html

<!DOCTYPE html>

<html lang="en">

<head>

<title>Car house</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link href='http://fonts.googleapis.com/css?family=Work+Sans:400,600,700' rel='stylesheet' type='text/css'>

<meta name="keywords" content="example, JavaScript Form Validation, Sample registration form"/>

<meta name="description" content="This document is an example of JavaScript Form Validation using a sample registration form."/>

<link rel='stylesheet' href='js-form-validation.css' type='text/css' />

<script src="sample-registration-form-validation.js"></script>

<link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css">

<link href="http://fonts.googleapis.com/css?family=Montserrat" rel="stylesheet">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>

<script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<script type="text/javascript" src="jquery-1.3.2.js"></script>

<script type="text/javascript" src="jquery.livequery.js"></script>

<link href="css.css" rel="stylesheet" />

<style>

body{ background:#000000;

color:#ffffff;

}

\*{ margin:0; padding:0;}

.top\_bg{ height:115px; width:100%; background:#063;}

.side\_bg{ width:100%; height:10px; background:#CCC; left:0; top:110px;}

.side\_bg\_black{ background:#000; height:10px; width:5%; margin-left:5%; float:left;}

.side\_bg.bot{top:280px; z-index:999;}

.car{ width:360px; height:140px; top:-200px; position:relative; left:-360px; animation:carmove 6s infinite;

-webkit-animation:carmove 6s infinite;}

.wheel{ background:#000; border-radius:36px; width:65px; height:65px; left: 56px; position: absolute; top: 65px; border:5px solid #fff;}

.wheel\_cap{ background:#FFF; border-radius:32px; margin:15%; height:70%; width:70%; position:relative;}

.wheel\_bar{ height:5px; background:#000; position:absolute; top:20px; width:100%;animation:mymove 0.1s infinite;

-webkit-animation:mymove 0.1s infinite;}

.wheel.front{left: 260px;}

.car\_top{ width:50%; height:40px; background:#F00; border-radius: 29px 50px 0 0; left: 58px; position: absolute; width: 60%;}

.car\_bottom{ width:10%; height:70px; background:#FF0; border-radius: 62px 55px 0 0; top:40px; left: 0; position: absolute; width: 100%;}

.window{ width:35%; background:#000; opacity:0.5; height:28px; position:absolute; top:12px; left:20px; border-radius: 10px 12px 0 0; border:2px solid #fff;}

.window.front{border-radius: 10px 30px 0 0; left: 125px; width: 35%;}

.seprator{ width:1px; height:110px; background:#999; top:0; left:47.1%;}

.road{ width:100%; background:#333; height:170px; top:120px; box-shadow:2px 2px 7px #000;}

.divider{ background:#CCC; height:5px; top:70px; width:100%;}

.navigation{ background:#000; width:20%; height:100px; position:absolute; bottom:0; margin-left:40%; border-radius:50% 50% 0 0; border-top:5px solid #930; border-left:5px solid #930; border-right:5px solid #930;}

.reverse, .forward{ font-family:Arial, Helvetica, sans-serif; background:#F00; color:#FFF; font-size:14px; padding:5px; border-radius:10px; position:absolute; left:20px; top:50px; cursor:pointer;}

.forward{ left:auto; right:20px;}

@keyframes mymove

{

0% {transform:rotate(0deg);}

15% {transform:rotate(30deg);}

35% {transform:rotate(60deg);}

50% {transform:rotate(90deg);}

65% {transform:rotate(120deg);}

80% {transform:rotate(150deg);}

100%{transform:rotate(180deg);}

}

@-webkit-keyframes mymove

{

0% {-webkit-transform:rotate(0deg);}

15% {-webkit-transform:rotate(30deg);}

35% {-webkit-transform:rotate(60deg);}

50% {-webkit-transform:rotate(90deg);}

65% {-webkit-transform:rotate(120deg);}

80% {-webkit-transform:rotate(150deg);}

100%{-webkit-transform:rotate(180deg);}

}

@keyframes carmove

{

0% {left:-20%;}

100% {left:100%;}

}

@-webkit-keyframes carmove

{

0% {left:-20%;}

100% {left:100%;}

}

p {font-size: 16px;

color:#ffffff;}

.margin {margin-bottom: 45px;}

.bg-1 {

background-color: #1abc9c;

color: #ffffff;

}

.bg-2 {

background-color: #474e5d;

color: #ffffff;

}

.bg-3 {

background-color: #ffffff;

color: #555555;

}

.bg-4 {

background-color: #2f2f2f;

color: black;

}

.hero-feature {

margin-bottom: 30px;

}

#myNavbar{color:black;}

.container{

margin-left : 150px;

margin-right : 150px;

float : left;

}

.container-fluid {

padding-top: 10px;

padding-bottom: 10px;

margin-left:130px;

margin-right:130px;

margin-bottom :30px;

border-radius : 1em;

}

.navbar {

padding-top: 0;

padding-bottom: 0;

border: 2px solid black;

border-radius: 30px;

margin-bottom: 0;

font-size:15px;

font-weight : bold;

display : inline;

}

.navbar-nav li a:hover {

color: #1abc9c !important;

}

.price{color:#ffffff;}

.thumbnail > img{

height :150px;

width : 1000px;

}

.caption{

color : red;}

p{

color : black;

}

.img-rounded{

border-radius : 1em;

}

.footer-right{

float : right;

color : gray;

float :center;

margin-bottom:100px;

margin-top:-25px}

.card-container {

cursor: pointer;

height: 150px;

perspective: 600;

position: relative;

width: 100%;

}

.card {

height: 100%;

transform-style: preserve-3d;

transition: all 1s ease-in-out;

width: 100%;

}

.card:hover {

transform: rotateY(180deg);

}

.card .side {

backface-visibility: hidden;

height: 100%;

position: absolute;

overflow: hidden;

width: 100%;

}

.card .back {

background: #ffffff;

color: #000000;

font-size:40px;

line-height: 150px;

text-align: center;

transform: rotateY(180deg);

}

</style>

<script type="text/javascript">

$(document).ready(function() {

var Arrays=new Array();

$('.add-to-cart-button').click(function(){

var thisID = $(this).parent().parent().attr('id').replace('detail-','');

var itemname = $(this).parent().find('.item\_name').html();

var itemprice = $(this).parent().find('.price').html();

if(include(Arrays,thisID))

{

var price = $('#each-'+thisID).children(".shopp-price").find('em').html();

var quantity = $('#each-'+thisID).children(".shopp-quantity").html();

quantity = parseInt(quantity)+parseInt(1);

var total = parseInt(itemprice)\*parseInt(quantity);

$('#each-'+thisID).children(".shopp-price").find('em').html(total);

$('#each-'+thisID).children(".shopp-quantity").html(quantity);

var prev\_charges = $('.cart-total span').html();

prev\_charges = parseInt(prev\_charges)-parseInt(price);

prev\_charges = parseInt(prev\_charges)+parseInt(total);

$('.cart-total span').html(prev\_charges);

$('#total-hidden-charges').val(prev\_charges);

}

else

{

Arrays.push(thisID);

var prev\_charges = $('.cart-total span').html();

prev\_charges = parseInt(prev\_charges)+parseInt(itemprice);

$('.cart-total span').html(prev\_charges);

$('#total-hidden-charges').val(prev\_charges);

var Height = $('#cart\_wrapper').height();

$('#cart\_wrapper').css({height:Height+parseInt(45)});

$('#cart\_wrapper .cart-info').append('<div class="shopp" id="each-'+thisID+'"><div class="label">'+itemname+'</div><div class="shopp-price"> $<em>'+itemprice+'</em></div><span class="shopp-quantity">1</span><img src="remove.png" class="remove" /><br class="all" /></div>');

}

});

$('.remove').livequery('click', function() {

var deduct = $(this).parent().children(".shopp-price").find('em').html();

var prev\_charges = $('.cart-total span').html();

var thisID = $(this).parent().attr('id').replace('each-','');

var pos = getpos(Arrays,thisID);

Arrays.splice(pos,1,"0")

prev\_charges = parseInt(prev\_charges)-parseInt(deduct);

$('.cart-total span').html(prev\_charges);

$('#total-hidden-charges').val(prev\_charges);

$(this).parent().remove();

});

$('#Submit').livequery('click', function() {

var totalCharge = $('#total-hidden-charges').val();

$('#cart\_wrapper').html('Total Charges: $'+totalCharge);

return false;

});

// this is for 2nd row's li offset from top. It means how much offset you want to give them with animation

var single\_li\_offset = 200;

var current\_opened\_box = -1;

$('#wrap li').click(function() {

var thisID = $(this).attr('id');

var $this = $(this);

var id = $('#wrap li').index($this);

if(current\_opened\_box == id) // if user click a opened box li again you close the box and return back

{

$('#wrap .detail-view').slideUp('slow');

return false;

}

$('#cart\_wrapper').slideUp('slow');

$('#wrap .detail-view').slideUp('slow');

// save this id. so if user click a opened box li again you close the box.

current\_opened\_box = id;

var targetOffset = 0;

// below conditions assumes that there are four li in one row and total rows are 4. How ever if you want to increase the rows you have to increase else-if conditions and if you want to increase li in one row, then you have to increment all value below. (if(id<=3)), if(id<=7) etc

if(id<=3)

targetOffset = 0;

else if(id<=7)

targetOffset = single\_li\_offset;

else if(id<=11)

targetOffset = single\_li\_offset\*2;

else if(id<=15)

targetOffset = single\_li\_offset\*3;

$("html:not(:animated),body:not(:animated)").animate({scrollTop: targetOffset}, 800,function(){

$('#wrap #detail-'+thisID).slideDown(500);

return false;

});

});

$('.close a').click(function() {

$('#wrap .detail-view').slideUp('slow');

});

$('.closeCart').click(function() {

$('#cart\_wrapper').slideUp();

});

$('#show\_cart').click(function() {

$('#cart\_wrapper').slideToggle('slow');

});

});

function include(arr, obj) {

for(var i=0; i<arr.length; i++) {

if (arr[i] == obj) return true;

}

}

function getpos(arr, obj) {

for(var i=0; i<arr.length; i++) {

if (arr[i] == obj) return i;

}

}

</script>

</head>

<body>

<nav class="navbar navbar-default">

<div class="container-fluid" ">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="#" style="color:#ffffff; font-size : 30px; padding:-5px">Car House</a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<div>

<ul class="nav navbar-nav" style="list-style:none;">

<li class="height" style="margin-top:10px"> <form>

<input type="text" onKeyUp="ajaxFunction(this.value)" placeholder="search..." />

<div id="cars" style="position:absolute; color:black; list-style:none; " ></div>

</form></li>

<li class="dropdown"><a class="dropdown-toggle" data-toggle="dropdown" href="#">Information <span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a href="#">Alumni</a></li>

<li><a href="#">Alumni Assosiates</a></li>

<li><a href="#">Feedback</a></li>

</ul>

</li>

<li><a href="#">Shipping and Returns</a></li>

<li><a href="#">Advanced Search</a></li>

</ul>

</div>

</div>

</div>

</nav>

<div class="card-container">

<div class="card">

<div class="side" style="margin-left:540px"><img src="pic20.jpg" alt="WELCOME USER"></div>

<div class="side back">WELCOME USER</div>

</div>

</div>

<div align="left" style="min-height:800px;">

<div id="cart\_wrapper" align="center">

<form action="#" id="cart\_form" name="cart\_form" Style="color:black;">

<div class="cart-info"></div>

<div class="cart-total">

<b>Total Charges:&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</b> $<span>0</span>

<input type="hidden" name="total-hidden-charges" id="total-hidden-charges" value="0" />

</div>

<button type="submit" id="Submit">CheckOut</button>

</form>

</div>

<div id="wrap" align="center">

<a id="show\_cart" href="javascript:void(0)" style="color:black; border-radius:10px;"><h4>View Cart</h3></a>

<ul style="color:black;">

<li id="1">

<img src="product\_img/1.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="2">

<img src="product\_img/2.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="3">

<img src="product\_img/3.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="4">

<img src="product\_img/4.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<!-- Detail Boxes for above four li -->

<div class="detail-view" id="detail-1">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/1.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping, merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-2">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/2.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$7000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-3">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/3.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$5000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-4">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/4.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$9000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<!---->

<li id="5">

<img src="product\_img/5.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="6">

<img src="product\_img/6.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="7">

<img src="product\_img/7.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="8">

<img src="product\_img/8.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<!-- Detail Boxes for above four li -->

<div class="detail-view" id="detail-5">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/5.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-6">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/6.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-7">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/7.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-8">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/8.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<!---->

<li id="9">

<img src="product\_img/9.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="10">

<img src="product\_img/10.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="11">

<img src="product\_img/11.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="12">

<img src="product\_img/12.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<!-- Detail Boxes for above four li -->

<div class="detail-view" id="detail-9">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/9.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-10">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/10.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<label class='item\_name'>Green Large Shirt </label>

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-11">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/11.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-12">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/12.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<!---->

<li id="13">

<img src="product\_img/13.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="14">

<img src="product\_img/14.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="15">

<img src="product\_img/15.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<li id="16">

<img src="product\_img/16.jpg" class="items" height="100" alt="" />

<br clear="all" />

</li>

<!-- Detail Boxes for above four li -->

<div class="detail-view" id="detail-13">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/13.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-14">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/14.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-15">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/15.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<div class="detail-view" id="detail-16">

<div class="close" align="right">

<a href="javascript:void(0)">x</a>

</div>

<img src="product\_img/16.jpg" class="detail\_images" width="340" height="310" alt="" />

<div class="detail\_info">

<br clear="all" />

<p>

shopping , merchandise, consumerism, gift:

<br clear="all" /><br clear="all" />

$<span class="price">$8000.00</span>

</p>

<br clear="all" />

<button class="add-to-cart-button">Add to Cart</button>

</div>

</div>

<br clear="all" />

</ul>

<br clear="all" />

</div>

<div class="top\_bg"></div>

<div class="side\_bg">

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

</div>

<div class="side\_bg bot">

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

<div class="side\_bg\_black"></div>

</div>

<div class="road">

<div class="divider"></div>

</div>

<div class="car">

<div class="car\_top">

<div class="window"></div>

<div class="window front"></div>

</div>

<div class="car\_bottom"></div>

<div class="wheel">

<div class="wheel\_cap">

<div class="wheel\_bar"></div>

</div>

</div>

<div class="wheel front">

<div class="wheel\_cap">

<div class="wheel\_bar"></div>

</div>

</div>

<div class="seprator"></div>

</div>

</div>

<div class="socialize" style="float:left ; margin-top:20px; margin-left:170px">

<div>

<a href="https://www.facebook.com/pranali.narkar"><img src="images/fb.png" width="68" height="68" class="facebook" alt="fb" onmouseover=hover(this) onmouseout=unhover(this)></a>

<a href="#"><img src="images/twitter.png" width="68" height="68" class="twitter" alt="twitter" onmouseover=hover(this) onmouseout=unhover(this)></a>

<a href="#"><img src="images/youtube.png" width="68" height="69" class="youtube" alt="youtube" onmouseover=hover(this) onmouseout=unhover(this)></a>

<a href="#"><img src="images/g+.png" width="68" height="68" class="google" alt="g+" onmouseover=hover(this) onmouseout=unhover(this)></a>

<a href="mailto:pranali.narkar017@gmail.com"><img src="images/message.png" width="68" height="68" class="message" alt="message" onmouseover=hover(this) onmouseout=unhover(this)></a>

</div>

<!--end of social imgs-->

<script>

function hover(element) {

var s = (element.src).replace(".png", "hover.png");

element.setAttribute("src", s);}

function unhover(element) {

var s = (element.src).replace("hover.png", ".png");

element.setAttribute("src", s);

}

</script>

</div>

<div class="copy" style="float:left ; margin-top:70px; margin-left:320px">

<p class="link" style="color:#ffffff;">&copy; All rights reserved | Design by&nbsp; <a href="#">PRANALI</a></p>

</div>

<script type="text/javascript">

function ajaxFunction(cars) //functionName could be anything

{

//1. Create XmlHttpRequest Object

//This could be written into an external .js file that could be used within other pages as well.

var xmlHttp;

try // Firefox, Opera 8.0+, Safari

{

xmlHttp=new XMLHttpRequest();

}

catch (e)

{

try // Internet Explorer

{

xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");

}

catch (e)

{

try

{

xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");

}

catch (e)

{

alert("Your browser does not support AJAX!");

return false;

}

}

}

//2. Call the server-side script

//cars is the function parameter passed when the JS function is called

xmlHttp.open("GET", "cars.jsp?q=" + cars, true); //q is the name of the parameter to be used in JSP

xmlHttp.send(); //leave blank, or pass null. Not used with GET requests. when using post we need to pass the parameters

//3. Check the server-data is ready

xmlHttp.onreadystatechange=function()

{

if(xmlHttp.readyState==4)

{

//4. Manipulate the DOM

var cars = document.getElementById("cars");

var serverData = xmlHttp.responseText;

cars.innerHTML = serverData;

}

}

}

</script>

</body>

</html>

Css

html, body{

margin:0;

padding:0;

border:0;

outline:0;

}

#wrap{ width:100%; min-height:900px; top:0px; position:relative; bottom:0px; }

#wrap ul{ margin:0px; padding:0px; width: 700px;text-align:center; }

#wrap .detail-view {

/\* background: none repeat scroll 0 0 #F3F4EE;\*/

border: 1px solid #E2E2E2;

border-top: 1px solid #E2E2E2;

left: 0;

height:380px;

overflow: hidden;

clear:both;

display:none;

margin-left:13px;

margin-bottom:15px;

width: 96%;

}

#wrap .detail-view .close{ text-align:right; width:98%; margin:5px; }

#wrap .close a{ padding:6px; height:10px; width:20px; color:#525049; }

#wrap .detail-view .detail\_images{ float:left}

#wrap .detail-view .detail\_info{

float:right;

font-family: "Helvetica Neue",Helvetica,"Nimbus Sans L",Arial,sans-serif;

color:#525049;

margin-right:20px;

margin-top:30px;

text-align:justify;

width:250px;

font-size:12px;

}

#wrap .detail-view .detail\_info label{ font-size:12px;text-transform:uppercase; letter-spacing:1px; line-height:60px;}

#wrap .detail-view .detail\_info p{ height:110px;}

a#show\_cart{

background: none repeat scroll 0 0 #FFFFFF;

border: 1px solid #E8E7DC;

cursor: pointer;

display:block;

display: inline-block;

font: 9px/21px "Helvetica Neue",Helvetica,"Nimbus Sans L",Arial,sans-serif;

letter-spacing: 2px;

color:#525049;

padding:8px;

text-decoration:none;

text-transform: uppercase;

}

.add-to-cart-button{

background: none repeat scroll 0 0 #FFFFFF;

border: 1px solid #E8E7DC;

cursor: pointer;

display: inline-block;

font: 9px/21px "Helvetica Neue",Helvetica,"Nimbus Sans L",Arial,sans-serif;

letter-spacing: 2px;

padding-top: 10px;color:#525049;

margin-top:15px;

text-transform: uppercase;

}

.add-to-cart-button:hover {

background: none repeat scroll 0 0 #F8F8F3;

}

.shopp{background: none repeat scroll 0 0 #F8F8F3;}

#wrap ul li{

list-style-type:none;

height:146px;

width:160px;

margin-left:10px;

margin-bottom:15px;

float:left;

padding:15px 0px 0px 0px;

font-family:"LubalGraphBdBTBold",Tahoma;

font-size:2em;

border:solid #fff 1px;

overflow:hidden;

}

.footer{ height:400px; background:#E2E2E2}

#wrap ul li:hover{ border:solid #f3f4ee 1px; }

#wrap ul li div{

height:31px;

text-align:center;

width:160px;

margin-top:10px;

position:relative;

bottom:0px;

padding-top:6px;

padding-bottom:4px;

background:#f3f4ee ;

font: 12px/21px "Helvetica Neue",Helvetica,"Nimbus Sans L",Arial,sans-serif;

opacity:0.8;

color: #525049 ;

text-shadow: 0px 2px 3px #555;

}

img#cart{bottom:0px;position:fixed; margin-left:30px; /\* keep the bar on top \*/}

#wrap ul li { cursor:pointer;}

#cart\_wrapper {

border:solid #E8E7DC 1px;

min-height:120px;

width:100%;

padding-top:15px;

display:none;

background:#E2E2E2;

font: 12px/21px "Helvetica Neue",Helvetica,"Nimbus Sans L",Arial,sans-serif;

position:relative

}

#Submit {

height: 78px;

float:left;

}

.closeCart{ cursor:pointer;}

button {

background: none repeat scroll 0 0 #FFFFFF;

border: 1px solid #E8E7DC;

width:140px;

cursor: pointer;

display: inline-block;

font: 9px/21px "Helvetica Neue",Helvetica,"Nimbus Sans L",Arial,sans-serif;

letter-spacing: 2px;

padding-top: 12px;color:#525049;

margin-top:1px;

border:solid #ccc 1px; padding:8px;

-webkit-border-radius: 8px;

-moz-border-radius: 8px;

margin-left:20px;

text-transform: uppercase;

}

button:hover {

background: none repeat scroll 0 0 #F8F8F3;

}

.cart-total{background: none repeat scroll 0 0 #F8F8F3;}

.shopp,.cart-total{

border:solid #E8E7DC 1px; padding:8px;

-webkit-border-radius: 8px;

-moz-border-radius: 8px; font-size:12px;

background:url(remove.png) center right no-repeat 5px;

border-radius: 8px;

font-family:"LubalGraphBdBTBold",Tahoma;

margin-top:3px;

width:320px;

float:left;

}

#cart\_form{ width:570px; padding-left:15px;}

div.shopp span{ float:left;}

div.shopp div.label{ width:130px; float:left; }

div.shopp div.shopp-price{ width:70px; float:left;}

.quantity{ float:left; margin-top:-3px; width:20px;}

img.remove{float:right;cursor:pointer;}

.cart-total b{width:130px;}

Cars.jsp

<%

String [] cars = new String[10];

cars[0] = "Abarth";

cars[1] = "Acura";

cars[2] = "Audi";

cars[3] = "Bentley";

cars[4] = "BMW";

cars[5] = "Corvette";

cars[6] = "Chevrolet";

cars[7] = "Donkervoort";

cars[8] = "Ferrari";

cars[9] = "Ginetta";

//read user input

String car = request.getParameter("q"); //q is the name of the parameter from AJAX call

//I am using UL/LI to return the data to xmlHttp object. Could be anything.

String result = "<UL>";

for (String currentcar : cars) //new for loop

{

if (currentcar.contains(car))

{

result += "<LI>" + currentcar + "</LI>";

}

}

result += "</UL>";

out.println(result); //send this to xmlHttp object

%>

Jquery library used:

/\*! Copyright (c) 2008 Brandon Aaron (http://brandonaaron.net)

\* Dual licensed under the MIT (http://www.opensource.org/licenses/mit-license.php)

\* and GPL (http://www.opensource.org/licenses/gpl-license.php) licenses.

\*

\* Version: 1.0.3

\* Requires jQuery 1.1.3+

\* Docs: http://docs.jquery.com/Plugins/livequery

\*/

(function($) {

$.extend($.fn, {

livequery: function(type, fn, fn2) {

var self = this, q;

// Handle different call patterns

if ($.isFunction(type))

fn2 = fn, fn = type, type = undefined;

// See if Live Query already exists

$.each( $.livequery.queries, function(i, query) {

if ( self.selector == query.selector && self.context == query.context &&

type == query.type && (!fn || fn.$lqguid == query.fn.$lqguid) && (!fn2 || fn2.$lqguid == query.fn2.$lqguid) )

// Found the query, exit the each loop

return (q = query) && false;

});

// Create new Live Query if it wasn't found

q = q || new $.livequery(this.selector, this.context, type, fn, fn2);

// Make sure it is running

q.stopped = false;

// Run it immediately for the first time

q.run();

// Contnue the chain

return this;

},

expire: function(type, fn, fn2) {

var self = this;

// Handle different call patterns

if ($.isFunction(type))

fn2 = fn, fn = type, type = undefined;

// Find the Live Query based on arguments and stop it

$.each( $.livequery.queries, function(i, query) {

if ( self.selector == query.selector && self.context == query.context &&

(!type || type == query.type) && (!fn || fn.$lqguid == query.fn.$lqguid) && (!fn2 || fn2.$lqguid == query.fn2.$lqguid) && !this.stopped )

$.livequery.stop(query.id);

});

// Continue the chain

return this;

}

});

$.livequery = function(selector, context, type, fn, fn2) {

this.selector = selector;

this.context = context || document;

this.type = type;

this.fn = fn;

this.fn2 = fn2;

this.elements = [];

this.stopped = false;

// The id is the index of the Live Query in $.livequery.queries

this.id = $.livequery.queries.push(this)-1;

// Mark the functions for matching later on

fn.$lqguid = fn.$lqguid || $.livequery.guid++;

if (fn2) fn2.$lqguid = fn2.$lqguid || $.livequery.guid++;

// Return the Live Query

return this;

};

$.livequery.prototype = {

stop: function() {

var query = this;

if ( this.type )

// Unbind all bound events

this.elements.unbind(this.type, this.fn);

else if (this.fn2)

// Call the second function for all matched elements

this.elements.each(function(i, el) {

query.fn2.apply(el);

});

// Clear out matched elements

this.elements = [];

// Stop the Live Query from running until restarted

this.stopped = true;

},

run: function() {

// Short-circuit if stopped

if ( this.stopped ) return;

var query = this;

var oEls = this.elements,

els = $(this.selector, this.context),

nEls = els.not(oEls);

// Set elements to the latest set of matched elements

this.elements = els;

if (this.type) {

// Bind events to newly matched elements

nEls.bind(this.type, this.fn);

// Unbind events to elements no longer matched

if (oEls.length > 0)

$.each(oEls, function(i, el) {

if ( $.inArray(el, els) < 0 )

$.event.remove(el, query.type, query.fn);

});

}

else {

// Call the first function for newly matched elements

nEls.each(function() {

query.fn.apply(this);

});

// Call the second function for elements no longer matched

if ( this.fn2 && oEls.length > 0 )

$.each(oEls, function(i, el) {

if ( $.inArray(el, els) < 0 )

query.fn2.apply(el);

});

}

}

};

$.extend($.livequery, {

guid: 0,

queries: [],

queue: [],

running: false,

timeout: null,

checkQueue: function() {

if ( $.livequery.running && $.livequery.queue.length ) {

var length = $.livequery.queue.length;

// Run each Live Query currently in the queue

while ( length-- )

$.livequery.queries[ $.livequery.queue.shift() ].run();

}

},

pause: function() {

// Don't run anymore Live Queries until restarted

$.livequery.running = false;

},

play: function() {

// Restart Live Queries

$.livequery.running = true;

// Request a run of the Live Queries

$.livequery.run();

},

registerPlugin: function() {

$.each( arguments, function(i,n) {

// Short-circuit if the method doesn't exist

if (!$.fn[n]) return;

// Save a reference to the original method

var old = $.fn[n];

// Create a new method

$.fn[n] = function() {

// Call the original method

var r = old.apply(this, arguments);

// Request a run of the Live Queries

$.livequery.run();

// Return the original methods result

return r;

}

});

},

run: function(id) {

if (id != undefined) {

// Put the particular Live Query in the queue if it doesn't already exist

if ( $.inArray(id, $.livequery.queue) < 0 )

$.livequery.queue.push( id );

}

else

// Put each Live Query in the queue if it doesn't already exist

$.each( $.livequery.queries, function(id) {

if ( $.inArray(id, $.livequery.queue) < 0 )

$.livequery.queue.push( id );

});

// Clear timeout if it already exists

if ($.livequery.timeout) clearTimeout($.livequery.timeout);

// Create a timeout to check the queue and actually run the Live Queries

$.livequery.timeout = setTimeout($.livequery.checkQueue, 20);

},

stop: function(id) {

if (id != undefined)

// Stop are particular Live Query

$.livequery.queries[ id ].stop();

else

// Stop all Live Queries

$.each( $.livequery.queries, function(id) {

$.livequery.queries[ id ].stop();

});

}

});

// Register core DOM manipulation methods

$.livequery.registerPlugin('append', 'prepend', 'after', 'before', 'wrap', 'attr', 'removeAttr', 'addClass', 'removeClass', 'toggleClass', 'empty', 'remove');

// Run Live Queries when the Document is ready

$(function() { $.livequery.play(); });

// Save a reference to the original init method

var init = $.prototype.init;

// Create a new init method that exposes two new properties: selector and context

$.prototype.init = function(a,c) {

// Call the original init and save the result

var r = init.apply(this, arguments);

// Copy over properties if they exist already

if (a && a.selector)

r.context = a.context, r.selector = a.selector;

// Set properties

if ( typeof a == 'string' )

r.context = c || document, r.selector = a;

// Return the result

return r;

};

// Give the init function the jQuery prototype for later instantiation (needed after Rev 4091)

$.prototype.init.prototype = $.prototype;

})(jQuery);

JQuery Library used:

/\*!

\* jQuery JavaScript Library v1.3.2

\* http://jquery.com/

\*

\* Copyright (c) 2009 John Resig

\* Dual licensed under the MIT and GPL licenses.

\* http://docs.jquery.com/License

\*

\* Date: 2009-02-19 17:34:21 -0500 (Thu, 19 Feb 2009)

\* Revision: 6246

\*/

(function(){

var

// Will speed up references to window, and allows munging its name.

window = this,

// Will speed up references to undefined, and allows munging its name.

undefined,

// Map over jQuery in case of overwrite

\_jQuery = window.jQuery,

// Map over the $ in case of overwrite

\_$ = window.$,

jQuery = window.jQuery = window.$ = function( selector, context ) {

// The jQuery object is actually just the init constructor 'enhanced'

return new jQuery.fn.init( selector, context );

},

// A simple way to check for HTML strings or ID strings

// (both of which we optimize for)

quickExpr = /^[^<]\*(<(.|\s)+>)[^>]\*$|^#([\w-]+)$/,

// Is it a simple selector

isSimple = /^.[^:#\[\.,]\*$/;

jQuery.fn = jQuery.prototype = {

init: function( selector, context ) {

// Make sure that a selection was provided

selector = selector || document;

// Handle $(DOMElement)

if ( selector.nodeType ) {

this[0] = selector;

this.length = 1;

this.context = selector;

return this;

}

// Handle HTML strings

if ( typeof selector === "string" ) {

// Are we dealing with HTML string or an ID?

var match = quickExpr.exec( selector );

// Verify a match, and that no context was specified for #id

if ( match && (match[1] || !context) ) {

// HANDLE: $(html) -> $(array)

if ( match[1] )

selector = jQuery.clean( [ match[1] ], context );

// HANDLE: $("#id")

else {

var elem = document.getElementById( match[3] );

// Handle the case where IE and Opera return items

// by name instead of ID

if ( elem && elem.id != match[3] )

return jQuery().find( selector );

// Otherwise, we inject the element directly into the jQuery object

var ret = jQuery( elem || [] );

ret.context = document;

ret.selector = selector;

return ret;

}

// HANDLE: $(expr, [context])

// (which is just equivalent to: $(content).find(expr)

} else

return jQuery( context ).find( selector );

// HANDLE: $(function)

// Shortcut for document ready

} else if ( jQuery.isFunction( selector ) )

return jQuery( document ).ready( selector );

// Make sure that old selector state is passed along

if ( selector.selector && selector.context ) {

this.selector = selector.selector;

this.context = selector.context;

}

return this.setArray(jQuery.isArray( selector ) ?

selector :

jQuery.makeArray(selector));

},

// Start with an empty selector

selector: "",

// The current version of jQuery being used

jquery: "1.3.2",

// The number of elements contained in the matched element set

size: function() {

return this.length;

},

// Get the Nth element in the matched element set OR

// Get the whole matched element set as a clean array

get: function( num ) {

return num === undefined ?

// Return a 'clean' array

Array.prototype.slice.call( this ) :

// Return just the object

this[ num ];

},

// Take an array of elements and push it onto the stack

// (returning the new matched element set)

pushStack: function( elems, name, selector ) {

// Build a new jQuery matched element set

var ret = jQuery( elems );

// Add the old object onto the stack (as a reference)

ret.prevObject = this;

ret.context = this.context;

if ( name === "find" )

ret.selector = this.selector + (this.selector ? " " : "") + selector;

else if ( name )

ret.selector = this.selector + "." + name + "(" + selector + ")";

// Return the newly-formed element set

return ret;

},

// Force the current matched set of elements to become

// the specified array of elements (destroying the stack in the process)

// You should use pushStack() in order to do this, but maintain the stack

setArray: function( elems ) {

// Resetting the length to 0, then using the native Array push

// is a super-fast way to populate an object with array-like properties

this.length = 0;

Array.prototype.push.apply( this, elems );

return this;

},

// Execute a callback for every element in the matched set.

// (You can seed the arguments with an array of args, but this is

// only used internally.)

each: function( callback, args ) {

return jQuery.each( this, callback, args );

},

// Determine the position of an element within

// the matched set of elements

index: function( elem ) {

// Locate the position of the desired element

return jQuery.inArray(

// If it receives a jQuery object, the first element is used

elem && elem.jquery ? elem[0] : elem

, this );

},

attr: function( name, value, type ) {

var options = name;

// Look for the case where we're accessing a style value

if ( typeof name === "string" )

if ( value === undefined )

return this[0] && jQuery[ type || "attr" ]( this[0], name );

else {

options = {};

options[ name ] = value;

}

// Check to see if we're setting style values

return this.each(function(i){

// Set all the styles

for ( name in options )

jQuery.attr(

type ?

this.style :

this,

name, jQuery.prop( this, options[ name ], type, i, name )

);

});

},

css: function( key, value ) {

// ignore negative width and height values

if ( (key == 'width' || key == 'height') && parseFloat(value) < 0 )

value = undefined;

return this.attr( key, value, "curCSS" );

},

text: function( text ) {

if ( typeof text !== "object" && text != null )

return this.empty().append( (this[0] && this[0].ownerDocument || document).createTextNode( text ) );

var ret = "";

jQuery.each( text || this, function(){

jQuery.each( this.childNodes, function(){

if ( this.nodeType != 8 )

ret += this.nodeType != 1 ?

this.nodeValue :

jQuery.fn.text( [ this ] );

});

});

return ret;

},

wrapAll: function( html ) {

if ( this[0] ) {

// The elements to wrap the target around

var wrap = jQuery( html, this[0].ownerDocument ).clone();

if ( this[0].parentNode )

wrap.insertBefore( this[0] );

wrap.map(function(){

var elem = this;

while ( elem.firstChild )

elem = elem.firstChild;

return elem;

}).append(this);

}

return this;

},

wrapInner: function( html ) {

return this.each(function(){

jQuery( this ).contents().wrapAll( html );

});

},

wrap: function( html ) {

return this.each(function(){

jQuery( this ).wrapAll( html );

});

},

append: function() {

return this.domManip(arguments, true, function(elem){

if (this.nodeType == 1)

this.appendChild( elem );

});

},

prepend: function() {

return this.domManip(arguments, true, function(elem){

if (this.nodeType == 1)

this.insertBefore( elem, this.firstChild );

});

},

before: function() {

return this.domManip(arguments, false, function(elem){

this.parentNode.insertBefore( elem, this );

});

},

after: function() {

return this.domManip(arguments, false, function(elem){

this.parentNode.insertBefore( elem, this.nextSibling );

});

},

end: function() {

return this.prevObject || jQuery( [] );

},

// For internal use only.

// Behaves like an Array's method, not like a jQuery method.

push: [].push,

sort: [].sort,

splice: [].splice,

find: function( selector ) {

if ( this.length === 1 ) {

var ret = this.pushStack( [], "find", selector );

ret.length = 0;

jQuery.find( selector, this[0], ret );

return ret;

} else {

return this.pushStack( jQuery.unique(jQuery.map(this, function(elem){

return jQuery.find( selector, elem );

})), "find", selector );

}

},

clone: function( events ) {

// Do the clone

var ret = this.map(function(){

if ( !jQuery.support.noCloneEvent && !jQuery.isXMLDoc(this) ) {

// IE copies events bound via attachEvent when

// using cloneNode. Calling detachEvent on the

// clone will also remove the events from the orignal

// In order to get around this, we use innerHTML.

// Unfortunately, this means some modifications to

// attributes in IE that are actually only stored

// as properties will not be copied (such as the

// the name attribute on an input).

var html = this.outerHTML;

if ( !html ) {

var div = this.ownerDocument.createElement("div");

div.appendChild( this.cloneNode(true) );

html = div.innerHTML;

}

return jQuery.clean([html.replace(/ jQuery\d+="(?:\d+|null)"/g, "").replace(/^\s\*/, "")])[0];

} else

return this.cloneNode(true);

});

// Copy the events from the original to the clone

if ( events === true ) {

var orig = this.find("\*").andSelf(), i = 0;

ret.find("\*").andSelf().each(function(){

if ( this.nodeName !== orig[i].nodeName )

return;

var events = jQuery.data( orig[i], "events" );

for ( var type in events ) {

for ( var handler in events[ type ] ) {

jQuery.event.add( this, type, events[ type ][ handler ], events[ type ][ handler ].data );

}

}

i++;

});

}

// Return the cloned set

return ret;

},

filter: function( selector ) {

return this.pushStack(

jQuery.isFunction( selector ) &&

jQuery.grep(this, function(elem, i){

return selector.call( elem, i );

}) ||

jQuery.multiFilter( selector, jQuery.grep(this, function(elem){

return elem.nodeType === 1;

}) ), "filter", selector );

},

closest: function( selector ) {

var pos = jQuery.expr.match.POS.test( selector ) ? jQuery(selector) : null,

closer = 0;

return this.map(function(){

var cur = this;

while ( cur && cur.ownerDocument ) {

if ( pos ? pos.index(cur) > -1 : jQuery(cur).is(selector) ) {

jQuery.data(cur, "closest", closer);

return cur;

}

cur = cur.parentNode;

closer++;

}

});

},

not: function( selector ) {

if ( typeof selector === "string" )

// test special case where just one selector is passed in

if ( isSimple.test( selector ) )

return this.pushStack( jQuery.multiFilter( selector, this, true ), "not", selector );

else

selector = jQuery.multiFilter( selector, this );

var isArrayLike = selector.length && selector[selector.length - 1] !== undefined && !selector.nodeType;

return this.filter(function() {

return isArrayLike ? jQuery.inArray( this, selector ) < 0 : this != selector;

});

},

add: function( selector ) {

return this.pushStack( jQuery.unique( jQuery.merge(

this.get(),

typeof selector === "string" ?

jQuery( selector ) :

jQuery.makeArray( selector )

)));

},

is: function( selector ) {

return !!selector && jQuery.multiFilter( selector, this ).length > 0;

},

hasClass: function( selector ) {

return !!selector && this.is( "." + selector );

},

val: function( value ) {

if ( value === undefined ) {

var elem = this[0];

if ( elem ) {

if( jQuery.nodeName( elem, 'option' ) )

return (elem.attributes.value || {}).specified ? elem.value : elem.text;

// We need to handle select boxes special

if ( jQuery.nodeName( elem, "select" ) ) {

var index = elem.selectedIndex,

values = [],

options = elem.options,

one = elem.type == "select-one";

// Nothing was selected

if ( index < 0 )

return null;

// Loop through all the selected options

for ( var i = one ? index : 0, max = one ? index + 1 : options.length; i < max; i++ ) {

var option = options[ i ];

if ( option.selected ) {

// Get the specifc value for the option

value = jQuery(option).val();

// We don't need an array for one selects

if ( one )

return value;

// Multi-Selects return an array

values.push( value );

}

}

return values;

}

// Everything else, we just grab the value

return (elem.value || "").replace(/\r/g, "");

}

return undefined;

}

if ( typeof value === "number" )

value += '';

return this.each(function(){

if ( this.nodeType != 1 )

return;

if ( jQuery.isArray(value) && /radio|checkbox/.test( this.type ) )

this.checked = (jQuery.inArray(this.value, value) >= 0 ||

jQuery.inArray(this.name, value) >= 0);

else if ( jQuery.nodeName( this, "select" ) ) {

var values = jQuery.makeArray(value);

jQuery( "option", this ).each(function(){

this.selected = (jQuery.inArray( this.value, values ) >= 0 ||

jQuery.inArray( this.text, values ) >= 0);

});

if ( !values.length )

this.selectedIndex = -1;

} else

this.value = value;

});

},

html: function( value ) {

return value === undefined ?

(this[0] ?

this[0].innerHTML.replace(/ jQuery\d+="(?:\d+|null)"/g, "") :

null) :

this.empty().append( value );

},

replaceWith: function( value ) {

return this.after( value ).remove();

},

eq: function( i ) {

return this.slice( i, +i + 1 );

},

slice: function() {

return this.pushStack( Array.prototype.slice.apply( this, arguments ),

"slice", Array.prototype.slice.call(arguments).join(",") );

},

map: function( callback ) {

return this.pushStack( jQuery.map(this, function(elem, i){

return callback.call( elem, i, elem );

}));

},

andSelf: function() {

return this.add( this.prevObject );

},

domManip: function( args, table, callback ) {

if ( this[0] ) {

var fragment = (this[0].ownerDocument || this[0]).createDocumentFragment(),

scripts = jQuery.clean( args, (this[0].ownerDocument || this[0]), fragment ),

first = fragment.firstChild;

if ( first )

for ( var i = 0, l = this.length; i < l; i++ )

callback.call( root(this[i], first), this.length > 1 || i > 0 ?

fragment.cloneNode(true) : fragment );

if ( scripts )

jQuery.each( scripts, evalScript );

}

return this;

function root( elem, cur ) {

return table && jQuery.nodeName(elem, "table") && jQuery.nodeName(cur, "tr") ?

(elem.getElementsByTagName("tbody")[0] ||

elem.appendChild(elem.ownerDocument.createElement("tbody"))) :

elem;

}

}

};

// Give the init function the jQuery prototype for later instantiation

jQuery.fn.init.prototype = jQuery.fn;

function evalScript( i, elem ) {

if ( elem.src )

jQuery.ajax({

url: elem.src,

async: false,

dataType: "script"

});

else

jQuery.globalEval( elem.text || elem.textContent || elem.innerHTML || "" );

if ( elem.parentNode )

elem.parentNode.removeChild( elem );

}

function now(){

return +new Date;

}

jQuery.extend = jQuery.fn.extend = function() {

// copy reference to target object

var target = arguments[0] || {}, i = 1, length = arguments.length, deep = false, options;

// Handle a deep copy situation

if ( typeof target === "boolean" ) {

deep = target;

target = arguments[1] || {};

// skip the boolean and the target

i = 2;

}

// Handle case when target is a string or something (possible in deep copy)

if ( typeof target !== "object" && !jQuery.isFunction(target) )

target = {};

// extend jQuery itself if only one argument is passed

if ( length == i ) {

target = this;

--i;

}

for ( ; i < length; i++ )

// Only deal with non-null/undefined values

if ( (options = arguments[ i ]) != null )

// Extend the base object

for ( var name in options ) {

var src = target[ name ], copy = options[ name ];

// Prevent never-ending loop

if ( target === copy )

continue;

// Recurse if we're merging object values

if ( deep && copy && typeof copy === "object" && !copy.nodeType )

target[ name ] = jQuery.extend( deep,

// Never move original objects, clone them

src || ( copy.length != null ? [ ] : { } )

, copy );

// Don't bring in undefined values

else if ( copy !== undefined )

target[ name ] = copy;

}

// Return the modified object

return target;

};

// exclude the following css properties to add px

var exclude = /z-?index|font-?weight|opacity|zoom|line-?height/i,

// cache defaultView

defaultView = document.defaultView || {},

toString = Object.prototype.toString;

jQuery.extend({

noConflict: function( deep ) {

window.$ = \_$;

if ( deep )

window.jQuery = \_jQuery;

return jQuery;

},

// See test/unit/core.js for details concerning isFunction.

// Since version 1.3, DOM methods and functions like alert

// aren't supported. They return false on IE (#2968).

isFunction: function( obj ) {

return toString.call(obj) === "[object Function]";

},

isArray: function( obj ) {

return toString.call(obj) === "[object Array]";

},

// check if an element is in a (or is an) XML document

isXMLDoc: function( elem ) {

return elem.nodeType === 9 && elem.documentElement.nodeName !== "HTML" ||

!!elem.ownerDocument && jQuery.isXMLDoc( elem.ownerDocument );

},

// Evalulates a script in a global context

globalEval: function( data ) {

if ( data && /\S/.test(data) ) {

// Inspired by code by Andrea Giammarchi

// http://webreflection.blogspot.com/2007/08/global-scope-evaluation-and-dom.html

var head = document.getElementsByTagName("head")[0] || document.documentElement,

script = document.createElement("script");

script.type = "text/javascript";

if ( jQuery.support.scriptEval )

script.appendChild( document.createTextNode( data ) );

else

script.text = data;

// Use insertBefore instead of appendChild to circumvent an IE6 bug.

// This arises when a base node is used (#2709).

head.insertBefore( script, head.firstChild );

head.removeChild( script );

}

},

nodeName: function( elem, name ) {

return elem.nodeName && elem.nodeName.toUpperCase() == name.toUpperCase();

},

// args is for internal usage only

each: function( object, callback, args ) {

var name, i = 0, length = object.length;

if ( args ) {

if ( length === undefined ) {

for ( name in object )

if ( callback.apply( object[ name ], args ) === false )

break;

} else

for ( ; i < length; )

if ( callback.apply( object[ i++ ], args ) === false )

break;

// A special, fast, case for the most common use of each

} else {

if ( length === undefined ) {

for ( name in object )

if ( callback.call( object[ name ], name, object[ name ] ) === false )

break;

} else

for ( var value = object[0];

i < length && callback.call( value, i, value ) !== false; value = object[++i] ){}

}

return object;

},

prop: function( elem, value, type, i, name ) {

// Handle executable functions

if ( jQuery.isFunction( value ) )

value = value.call( elem, i );

// Handle passing in a number to a CSS property

return typeof value === "number" && type == "curCSS" && !exclude.test( name ) ?

value + "px" :

value;

},

className: {

// internal only, use addClass("class")

add: function( elem, classNames ) {

jQuery.each((classNames || "").split(/\s+/), function(i, className){

if ( elem.nodeType == 1 && !jQuery.className.has( elem.className, className ) )

elem.className += (elem.className ? " " : "") + className;

});

},

// internal only, use removeClass("class")

remove: function( elem, classNames ) {

if (elem.nodeType == 1)

elem.className = classNames !== undefined ?

jQuery.grep(elem.className.split(/\s+/), function(className){

return !jQuery.className.has( classNames, className );

}).join(" ") :

"";

},

// internal only, use hasClass("class")

has: function( elem, className ) {

return elem && jQuery.inArray( className, (elem.className || elem).toString().split(/\s+/) ) > -1;

}

},

// A method for quickly swapping in/out CSS properties to get correct calculations

swap: function( elem, options, callback ) {

var old = {};

// Remember the old values, and insert the new ones

for ( var name in options ) {

old[ name ] = elem.style[ name ];

elem.style[ name ] = options[ name ];

}

callback.call( elem );

// Revert the old values

for ( var name in options )

elem.style[ name ] = old[ name ];

},

css: function( elem, name, force, extra ) {

if ( name == "width" || name == "height" ) {

var val, props = { position: "absolute", visibility: "hidden", display:"block" }, which = name == "width" ? [ "Left", "Right" ] : [ "Top", "Bottom" ];

function getWH() {

val = name == "width" ? elem.offsetWidth : elem.offsetHeight;

if ( extra === "border" )

return;

jQuery.each( which, function() {

if ( !extra )

val -= parseFloat(jQuery.curCSS( elem, "padding" + this, true)) || 0;

if ( extra === "margin" )

val += parseFloat(jQuery.curCSS( elem, "margin" + this, true)) || 0;

else

val -= parseFloat(jQuery.curCSS( elem, "border" + this + "Width", true)) || 0;

});

}

if ( elem.offsetWidth !== 0 )

getWH();

else

jQuery.swap( elem, props, getWH );

return Math.max(0, Math.round(val));

}

return jQuery.curCSS( elem, name, force );

},

curCSS: function( elem, name, force ) {

var ret, style = elem.style;

// We need to handle opacity special in IE

if ( name == "opacity" && !jQuery.support.opacity ) {

ret = jQuery.attr( style, "opacity" );

return ret == "" ?

"1" :

ret;

}

// Make sure we're using the right name for getting the float value

if ( name.match( /float/i ) )

name = styleFloat;

if ( !force && style && style[ name ] )

ret = style[ name ];

else if ( defaultView.getComputedStyle ) {

// Only "float" is needed here

if ( name.match( /float/i ) )

name = "float";

name = name.replace( /([A-Z])/g, "-$1" ).toLowerCase();

var computedStyle = defaultView.getComputedStyle( elem, null );

if ( computedStyle )

ret = computedStyle.getPropertyValue( name );

// We should always get a number back from opacity

if ( name == "opacity" && ret == "" )

ret = "1";

} else if ( elem.currentStyle ) {

var camelCase = name.replace(/\-(\w)/g, function(all, letter){

return letter.toUpperCase();

});

ret = elem.currentStyle[ name ] || elem.currentStyle[ camelCase ];

// From the awesome hack by Dean Edwards

// http://erik.eae.net/archives/2007/07/27/18.54.15/#comment-102291

// If we're not dealing with a regular pixel number

// but a number that has a weird ending, we need to convert it to pixels

if ( !/^\d+(px)?$/i.test( ret ) && /^\d/.test( ret ) ) {

// Remember the original values

var left = style.left, rsLeft = elem.runtimeStyle.left;

// Put in the new values to get a computed value out

elem.runtimeStyle.left = elem.currentStyle.left;

style.left = ret || 0;

ret = style.pixelLeft + "px";

// Revert the changed values

style.left = left;

elem.runtimeStyle.left = rsLeft;

}

}

return ret;

},

clean: function( elems, context, fragment ) {

context = context || document;

// !context.createElement fails in IE with an error but returns typeof 'object'

if ( typeof context.createElement === "undefined" )

context = context.ownerDocument || context[0] && context[0].ownerDocument || document;

// If a single string is passed in and it's a single tag

// just do a createElement and skip the rest

if ( !fragment && elems.length === 1 && typeof elems[0] === "string" ) {

var match = /^<(\w+)\s\*\/?>$/.exec(elems[0]);

if ( match )

return [ context.createElement( match[1] ) ];

}

var ret = [], scripts = [], div = context.createElement("div");

jQuery.each(elems, function(i, elem){

if ( typeof elem === "number" )

elem += '';

if ( !elem )

return;

// Convert html string into DOM nodes

if ( typeof elem === "string" ) {

// Fix "XHTML"-style tags in all browsers

elem = elem.replace(/(<(\w+)[^>]\*?)\/>/g, function(all, front, tag){

return tag.match(/^(abbr|br|col|img|input|link|meta|param|hr|area|embed)$/i) ?

all :

front + "></" + tag + ">";

});

// Trim whitespace, otherwise indexOf won't work as expected

var tags = elem.replace(/^\s+/, "").substring(0, 10).toLowerCase();

var wrap =

// option or optgroup

!tags.indexOf("<opt") &&

[ 1, "<select multiple='multiple'>", "</select>" ] ||

!tags.indexOf("<leg") &&

[ 1, "<fieldset>", "</fieldset>" ] ||

tags.match(/^<(thead|tbody|tfoot|colg|cap)/) &&

[ 1, "<table>", "</table>" ] ||

!tags.indexOf("<tr") &&

[ 2, "<table><tbody>", "</tbody></table>" ] ||

// <thead> matched above

(!tags.indexOf("<td") || !tags.indexOf("<th")) &&

[ 3, "<table><tbody><tr>", "</tr></tbody></table>" ] ||

!tags.indexOf("<col") &&

[ 2, "<table><tbody></tbody><colgroup>", "</colgroup></table>" ] ||

// IE can't serialize <link> and <script> tags normally

!jQuery.support.htmlSerialize &&

[ 1, "div<div>", "</div>" ] ||

[ 0, "", "" ];

// Go to html and back, then peel off extra wrappers

div.innerHTML = wrap[1] + elem + wrap[2];

// Move to the right depth

while ( wrap[0]-- )

div = div.lastChild;

// Remove IE's autoinserted <tbody> from table fragments

if ( !jQuery.support.tbody ) {

// String was a <table>, \*may\* have spurious <tbody>

var hasBody = /<tbody/i.test(elem),

tbody = !tags.indexOf("<table") && !hasBody ?

div.firstChild && div.firstChild.childNodes :

// String was a bare <thead> or <tfoot>

wrap[1] == "<table>" && !hasBody ?

div.childNodes :

[];

for ( var j = tbody.length - 1; j >= 0 ; --j )

if ( jQuery.nodeName( tbody[ j ], "tbody" ) && !tbody[ j ].childNodes.length )

tbody[ j ].parentNode.removeChild( tbody[ j ] );

}

// IE completely kills leading whitespace when innerHTML is used

if ( !jQuery.support.leadingWhitespace && /^\s/.test( elem ) )

div.insertBefore( context.createTextNode( elem.match(/^\s\*/)[0] ), div.firstChild );

elem = jQuery.makeArray( div.childNodes );

}

if ( elem.nodeType )

ret.push( elem );

else

ret = jQuery.merge( ret, elem );

});

if ( fragment ) {

for ( var i = 0; ret[i]; i++ ) {

if ( jQuery.nodeName( ret[i], "script" ) && (!ret[i].type || ret[i].type.toLowerCase() === "text/javascript") ) {

scripts.push( ret[i].parentNode ? ret[i].parentNode.removeChild( ret[i] ) : ret[i] );

} else {

if ( ret[i].nodeType === 1 )

ret.splice.apply( ret, [i + 1, 0].concat(jQuery.makeArray(ret[i].getElementsByTagName("script"))) );

fragment.appendChild( ret[i] );

}

}

return scripts;

}

return ret;

},

attr: function( elem, name, value ) {

// don't set attributes on text and comment nodes

if (!elem || elem.nodeType == 3 || elem.nodeType == 8)

return undefined;

var notxml = !jQuery.isXMLDoc( elem ),

// Whether we are setting (or getting)

set = value !== undefined;

// Try to normalize/fix the name

name = notxml && jQuery.props[ name ] || name;

// Only do all the following if this is a node (faster for style)

// IE elem.getAttribute passes even for style

if ( elem.tagName ) {

// These attributes require special treatment

var special = /href|src|style/.test( name );

// Safari mis-reports the default selected property of a hidden option

// Accessing the parent's selectedIndex property fixes it

if ( name == "selected" && elem.parentNode )

elem.parentNode.selectedIndex;

// If applicable, access the attribute via the DOM 0 way

if ( name in elem && notxml && !special ) {

if ( set ){

// We can't allow the type property to be changed (since it causes problems in IE)

if ( name == "type" && jQuery.nodeName( elem, "input" ) && elem.parentNode )

throw "type property can't be changed";

elem[ name ] = value;

}

// browsers index elements by id/name on forms, give priority to attributes.

if( jQuery.nodeName( elem, "form" ) && elem.getAttributeNode(name) )

return elem.getAttributeNode( name ).nodeValue;

// elem.tabIndex doesn't always return the correct value when it hasn't been explicitly set

// http://fluidproject.org/blog/2008/01/09/getting-setting-and-removing-tabindex-values-with-javascript/

if ( name == "tabIndex" ) {

var attributeNode = elem.getAttributeNode( "tabIndex" );

return attributeNode && attributeNode.specified

? attributeNode.value

: elem.nodeName.match(/(button|input|object|select|textarea)/i)

? 0

: elem.nodeName.match(/^(a|area)$/i) && elem.href

? 0

: undefined;

}

return elem[ name ];

}

if ( !jQuery.support.style && notxml && name == "style" )

return jQuery.attr( elem.style, "cssText", value );

if ( set )

// convert the value to a string (all browsers do this but IE) see #1070

elem.setAttribute( name, "" + value );

var attr = !jQuery.support.hrefNormalized && notxml && special

// Some attributes require a special call on IE

? elem.getAttribute( name, 2 )

: elem.getAttribute( name );

// Non-existent attributes return null, we normalize to undefined

return attr === null ? undefined : attr;

}

// elem is actually elem.style ... set the style

// IE uses filters for opacity

if ( !jQuery.support.opacity && name == "opacity" ) {

if ( set ) {

// IE has trouble with opacity if it does not have layout

// Force it by setting the zoom level

elem.zoom = 1;

// Set the alpha filter to set the opacity

elem.filter = (elem.filter || "").replace( /alpha\([^)]\*\)/, "" ) +

(parseInt( value ) + '' == "NaN" ? "" : "alpha(opacity=" + value \* 100 + ")");

}

return elem.filter && elem.filter.indexOf("opacity=") >= 0 ?

(parseFloat( elem.filter.match(/opacity=([^)]\*)/)[1] ) / 100) + '':

"";

}

name = name.replace(/-([a-z])/ig, function(all, letter){

return letter.toUpperCase();

});

if ( set )

elem[ name ] = value;

return elem[ name ];

},

trim: function( text ) {

return (text || "").replace( /^\s+|\s+$/g, "" );

},

makeArray: function( array ) {

var ret = [];

if( array != null ){

var i = array.length;

// The window, strings (and functions) also have 'length'

if( i == null || typeof array === "string" || jQuery.isFunction(array) || array.setInterval )

ret[0] = array;

else

while( i )

ret[--i] = array[i];

}

return ret;

},

inArray: function( elem, array ) {

for ( var i = 0, length = array.length; i < length; i++ )

// Use === because on IE, window == document

if ( array[ i ] === elem )

return i;

return -1;

},

merge: function( first, second ) {

// We have to loop this way because IE & Opera overwrite the length

// expando of getElementsByTagName

var i = 0, elem, pos = first.length;

// Also, we need to make sure that the correct elements are being returned

// (IE returns comment nodes in a '\*' query)

if ( !jQuery.support.getAll ) {

while ( (elem = second[ i++ ]) != null )

if ( elem.nodeType != 8 )

first[ pos++ ] = elem;

} else

while ( (elem = second[ i++ ]) != null )

first[ pos++ ] = elem;

return first;

},

unique: function( array ) {

var ret = [], done = {};

try {

for ( var i = 0, length = array.length; i < length; i++ ) {

var id = jQuery.data( array[ i ] );

if ( !done[ id ] ) {

done[ id ] = true;

ret.push( array[ i ] );

}

}

} catch( e ) {

ret = array;

}

return ret;

},

grep: function( elems, callback, inv ) {

var ret = [];

// Go through the array, only saving the items

// that pass the validator function

for ( var i = 0, length = elems.length; i < length; i++ )

if ( !inv != !callback( elems[ i ], i ) )

ret.push( elems[ i ] );

return ret;

},

map: function( elems, callback ) {

var ret = [];

// Go through the array, translating each of the items to their

// new value (or values).

for ( var i = 0, length = elems.length; i < length; i++ ) {

var value = callback( elems[ i ], i );

if ( value != null )

ret[ ret.length ] = value;

}

return ret.concat.apply( [], ret );

}

});

// Use of jQuery.browser is deprecated.

// It's included for backwards compatibility and plugins,

// although they should work to migrate away.

var userAgent = navigator.userAgent.toLowerCase();

// Figure out what browser is being used

jQuery.browser = {

version: (userAgent.match( /.+(?:rv|it|ra|ie)[\/: ]([\d.]+)/ ) || [0,'0'])[1],

safari: /webkit/.test( userAgent ),

opera: /opera/.test( userAgent ),

msie: /msie/.test( userAgent ) && !/opera/.test( userAgent ),

mozilla: /mozilla/.test( userAgent ) && !/(compatible|webkit)/.test( userAgent )

};

jQuery.each({

parent: function(elem){return elem.parentNode;},

parents: function(elem){return jQuery.dir(elem,"parentNode");},

next: function(elem){return jQuery.nth(elem,2,"nextSibling");},

prev: function(elem){return jQuery.nth(elem,2,"previousSibling");},

nextAll: function(elem){return jQuery.dir(elem,"nextSibling");},

prevAll: function(elem){return jQuery.dir(elem,"previousSibling");},

siblings: function(elem){return jQuery.sibling(elem.parentNode.firstChild,elem);},

children: function(elem){return jQuery.sibling(elem.firstChild);},

contents: function(elem){return jQuery.nodeName(elem,"iframe")?elem.contentDocument||elem.contentWindow.document:jQuery.makeArray(elem.childNodes);}

}, function(name, fn){

jQuery.fn[ name ] = function( selector ) {

var ret = jQuery.map( this, fn );

if ( selector && typeof selector == "string" )

ret = jQuery.multiFilter( selector, ret );

return this.pushStack( jQuery.unique( ret ), name, selector );

};

});

jQuery.each({

appendTo: "append",

prependTo: "prepend",

insertBefore: "before",

insertAfter: "after",

replaceAll: "replaceWith"

}, function(name, original){

jQuery.fn[ name ] = function( selector ) {

var ret = [], insert = jQuery( selector );

for ( var i = 0, l = insert.length; i < l; i++ ) {

var elems = (i > 0 ? this.clone(true) : this).get();

jQuery.fn[ original ].apply( jQuery(insert[i]), elems );

ret = ret.concat( elems );

}

return this.pushStack( ret, name, selector );

};

});

jQuery.each({

removeAttr: function( name ) {

jQuery.attr( this, name, "" );

if (this.nodeType == 1)

this.removeAttribute( name );

},

addClass: function( classNames ) {

jQuery.className.add( this, classNames );

},

removeClass: function( classNames ) {

jQuery.className.remove( this, classNames );

},

toggleClass: function( classNames, state ) {

if( typeof state !== "boolean" )

state = !jQuery.className.has( this, classNames );

jQuery.className[ state ? "add" : "remove" ]( this, classNames );

},

remove: function( selector ) {

if ( !selector || jQuery.filter( selector, [ this ] ).length ) {

// Prevent memory leaks

jQuery( "\*", this ).add([this]).each(function(){

jQuery.event.remove(this);

jQuery.removeData(this);

});

if (this.parentNode)

this.parentNode.removeChild( this );

}

},

empty: function() {

// Remove element nodes and prevent memory leaks

jQuery(this).children().remove();

// Remove any remaining nodes

while ( this.firstChild )

this.removeChild( this.firstChild );

}

}, function(name, fn){

jQuery.fn[ name ] = function(){

return this.each( fn, arguments );

};

});

// Helper function used by the dimensions and offset modules

function num(elem, prop) {

return elem[0] && parseInt( jQuery.curCSS(elem[0], prop, true), 10 ) || 0;

}

var expando = "jQuery" + now(), uuid = 0, windowData = {};

jQuery.extend({

cache: {},

data: function( elem, name, data ) {

elem = elem == window ?

windowData :

elem;

var id = elem[ expando ];

// Compute a unique ID for the element

if ( !id )

id = elem[ expando ] = ++uuid;

// Only generate the data cache if we're

// trying to access or manipulate it

if ( name && !jQuery.cache[ id ] )

jQuery.cache[ id ] = {};

// Prevent overriding the named cache with undefined values

if ( data !== undefined )

jQuery.cache[ id ][ name ] = data;

// Return the named cache data, or the ID for the element

return name ?

jQuery.cache[ id ][ name ] :

id;

},

removeData: function( elem, name ) {

elem = elem == window ?

windowData :

elem;

var id = elem[ expando ];

// If we want to remove a specific section of the element's data

if ( name ) {

if ( jQuery.cache[ id ] ) {

// Remove the section of cache data

delete jQuery.cache[ id ][ name ];

// If we've removed all the data, remove the element's cache

name = "";

for ( name in jQuery.cache[ id ] )

break;

if ( !name )

jQuery.removeData( elem );

}

// Otherwise, we want to remove all of the element's data

} else {

// Clean up the element expando

try {

delete elem[ expando ];

} catch(e){

// IE has trouble directly removing the expando

// but it's ok with using removeAttribute

if ( elem.removeAttribute )

elem.removeAttribute( expando );

}

// Completely remove the data cache

delete jQuery.cache[ id ];

}

},

queue: function( elem, type, data ) {

if ( elem ){

type = (type || "fx") + "queue";

var q = jQuery.data( elem, type );

if ( !q || jQuery.isArray(data) )

q = jQuery.data( elem, type, jQuery.makeArray(data) );

else if( data )

q.push( data );

}

return q;

},

dequeue: function( elem, type ){

var queue = jQuery.queue( elem, type ),

fn = queue.shift();

if( !type || type === "fx" )

fn = queue[0];

if( fn !== undefined )

fn.call(elem);

}

});

jQuery.fn.extend({

data: function( key, value ){

var parts = key.split(".");

parts[1] = parts[1] ? "." + parts[1] : "";

if ( value === undefined ) {

var data = this.triggerHandler("getData" + parts[1] + "!", [parts[0]]);

if ( data === undefined && this.length )

data = jQuery.data( this[0], key );

return data === undefined && parts[1] ?

this.data( parts[0] ) :

data;

} else

return this.trigger("setData" + parts[1] + "!", [parts[0], value]).each(function(){

jQuery.data( this, key, value );

});

},

removeData: function( key ){

return this.each(function(){

jQuery.removeData( this, key );

});

},

queue: function(type, data){

if ( typeof type !== "string" ) {

data = type;

type = "fx";

}

if ( data === undefined )

return jQuery.queue( this[0], type );

return this.each(function(){

var queue = jQuery.queue( this, type, data );

if( type == "fx" && queue.length == 1 )

queue[0].call(this);

});

},

dequeue: function(type){

return this.each(function(){

jQuery.dequeue( this, type );

});

}

});/\*!

\* Sizzle CSS Selector Engine - v0.9.3

\* Copyright 2009, The Dojo Foundation

\* Released under the MIT, BSD, and GPL Licenses.

\* More information: http://sizzlejs.com/

\*/

(function(){

var chunker = /((?:\((?:\([^()]+\)|[^()]+)+\)|\[(?:\[[^[\]]\*\]|['"][^'"]\*['"]|[^[\]'"]+)+\]|\\.|[^ >+~,(\[\\]+)+|[>+~])(\s\*,\s\*)?/g,

done = 0,

toString = Object.prototype.toString;

var Sizzle = function(selector, context, results, seed) {

results = results || [];

context = context || document;

if ( context.nodeType !== 1 && context.nodeType !== 9 )

return [];

if ( !selector || typeof selector !== "string" ) {

return results;

}

var parts = [], m, set, checkSet, check, mode, extra, prune = true;

// Reset the position of the chunker regexp (start from head)

chunker.lastIndex = 0;

while ( (m = chunker.exec(selector)) !== null ) {

parts.push( m[1] );

if ( m[2] ) {

extra = RegExp.rightContext;

break;

}

}

if ( parts.length > 1 && origPOS.exec( selector ) ) {

if ( parts.length === 2 && Expr.relative[ parts[0] ] ) {

set = posProcess( parts[0] + parts[1], context );

} else {

set = Expr.relative[ parts[0] ] ?

[ context ] :

Sizzle( parts.shift(), context );

while ( parts.length ) {

selector = parts.shift();

if ( Expr.relative[ selector ] )

selector += parts.shift();

set = posProcess( selector, set );

}

}

} else {

var ret = seed ?

{ expr: parts.pop(), set: makeArray(seed) } :

Sizzle.find( parts.pop(), parts.length === 1 && context.parentNode ? context.parentNode : context, isXML(context) );

set = Sizzle.filter( ret.expr, ret.set );

if ( parts.length > 0 ) {

checkSet = makeArray(set);

} else {

prune = false;

}

while ( parts.length ) {

var cur = parts.pop(), pop = cur;

if ( !Expr.relative[ cur ] ) {

cur = "";

} else {

pop = parts.pop();

}

if ( pop == null ) {

pop = context;

}

Expr.relative[ cur ]( checkSet, pop, isXML(context) );

}

}

if ( !checkSet ) {

checkSet = set;

}

if ( !checkSet ) {

throw "Syntax error, unrecognized expression: " + (cur || selector);

}

if ( toString.call(checkSet) === "[object Array]" ) {

if ( !prune ) {

results.push.apply( results, checkSet );

} else if ( context.nodeType === 1 ) {

for ( var i = 0; checkSet[i] != null; i++ ) {

if ( checkSet[i] && (checkSet[i] === true || checkSet[i].nodeType === 1 && contains(context, checkSet[i])) ) {

results.push( set[i] );

}

}

} else {

for ( var i = 0; checkSet[i] != null; i++ ) {

if ( checkSet[i] && checkSet[i].nodeType === 1 ) {

results.push( set[i] );

}

}

}

} else {

makeArray( checkSet, results );

}

if ( extra ) {

Sizzle( extra, context, results, seed );

if ( sortOrder ) {

hasDuplicate = false;

results.sort(sortOrder);

if ( hasDuplicate ) {

for ( var i = 1; i < results.length; i++ ) {

if ( results[i] === results[i-1] ) {

results.splice(i--, 1);

}

}

}

}

}

return results;

};

Sizzle.matches = function(expr, set){

return Sizzle(expr, null, null, set);

};

Sizzle.find = function(expr, context, isXML){

var set, match;

if ( !expr ) {

return [];

}

for ( var i = 0, l = Expr.order.length; i < l; i++ ) {

var type = Expr.order[i], match;

if ( (match = Expr.match[ type ].exec( expr )) ) {

var left = RegExp.leftContext;

if ( left.substr( left.length - 1 ) !== "\\" ) {

match[1] = (match[1] || "").replace(/\\/g, "");

set = Expr.find[ type ]( match, context, isXML );

if ( set != null ) {

expr = expr.replace( Expr.match[ type ], "" );

break;

}

}

}

}

if ( !set ) {

set = context.getElementsByTagName("\*");

}

return {set: set, expr: expr};

};

Sizzle.filter = function(expr, set, inplace, not){

var old = expr, result = [], curLoop = set, match, anyFound,

isXMLFilter = set && set[0] && isXML(set[0]);

while ( expr && set.length ) {

for ( var type in Expr.filter ) {

if ( (match = Expr.match[ type ].exec( expr )) != null ) {

var filter = Expr.filter[ type ], found, item;

anyFound = false;

if ( curLoop == result ) {

result = [];

}

if ( Expr.preFilter[ type ] ) {

match = Expr.preFilter[ type ]( match, curLoop, inplace, result, not, isXMLFilter );

if ( !match ) {

anyFound = found = true;

} else if ( match === true ) {

continue;

}

}

if ( match ) {

for ( var i = 0; (item = curLoop[i]) != null; i++ ) {

if ( item ) {

found = filter( item, match, i, curLoop );

var pass = not ^ !!found;

if ( inplace && found != null ) {

if ( pass ) {

anyFound = true;

} else {

curLoop[i] = false;

}

} else if ( pass ) {

result.push( item );

anyFound = true;

}

}

}

}

if ( found !== undefined ) {

if ( !inplace ) {

curLoop = result;

}

expr = expr.replace( Expr.match[ type ], "" );

if ( !anyFound ) {

return [];

}

break;

}

}

}

// Improper expression

if ( expr == old ) {

if ( anyFound == null ) {

throw "Syntax error, unrecognized expression: " + expr;

} else {

break;

}

}

old = expr;

}

return curLoop;

};

var Expr = Sizzle.selectors = {

order: [ "ID", "NAME", "TAG" ],

match: {

ID: /#((?:[\w\u00c0-\uFFFF\_-]|\\.)+)/,

CLASS: /\.((?:[\w\u00c0-\uFFFF\_-]|\\.)+)/,

NAME: /\[name=['"]\*((?:[\w\u00c0-\uFFFF\_-]|\\.)+)['"]\*\]/,

ATTR: /\[\s\*((?:[\w\u00c0-\uFFFF\_-]|\\.)+)\s\*(?:(\S?=)\s\*(['"]\*)(.\*?)\3|)\s\*\]/,

TAG: /^((?:[\w\u00c0-\uFFFF\\*\_-]|\\.)+)/,

CHILD: /:(only|nth|last|first)-child(?:\((even|odd|[\dn+-]\*)\))?/,

POS: /:(nth|eq|gt|lt|first|last|even|odd)(?:\((\d\*)\))?(?=[^-]|$)/,

PSEUDO: /:((?:[\w\u00c0-\uFFFF\_-]|\\.)+)(?:\((['"]\*)((?:\([^\)]+\)|[^\2\(\)]\*)+)\2\))?/

},

attrMap: {

"class": "className",

"for": "htmlFor"

},

attrHandle: {

href: function(elem){

return elem.getAttribute("href");

}

},

relative: {

"+": function(checkSet, part, isXML){

var isPartStr = typeof part === "string",

isTag = isPartStr && !/\W/.test(part),

isPartStrNotTag = isPartStr && !isTag;

if ( isTag && !isXML ) {

part = part.toUpperCase();

}

for ( var i = 0, l = checkSet.length, elem; i < l; i++ ) {

if ( (elem = checkSet[i]) ) {

while ( (elem = elem.previousSibling) && elem.nodeType !== 1 ) {}

checkSet[i] = isPartStrNotTag || elem && elem.nodeName === part ?

elem || false :

elem === part;

}

}

if ( isPartStrNotTag ) {

Sizzle.filter( part, checkSet, true );

}

},

">": function(checkSet, part, isXML){

var isPartStr = typeof part === "string";

if ( isPartStr && !/\W/.test(part) ) {

part = isXML ? part : part.toUpperCase();

for ( var i = 0, l = checkSet.length; i < l; i++ ) {

var elem = checkSet[i];

if ( elem ) {

var parent = elem.parentNode;

checkSet[i] = parent.nodeName === part ? parent : false;

}

}

} else {

for ( var i = 0, l = checkSet.length; i < l; i++ ) {

var elem = checkSet[i];

if ( elem ) {

checkSet[i] = isPartStr ?

elem.parentNode :

elem.parentNode === part;

}

}

if ( isPartStr ) {

Sizzle.filter( part, checkSet, true );

}

}

},

"": function(checkSet, part, isXML){

var doneName = done++, checkFn = dirCheck;

if ( !part.match(/\W/) ) {

var nodeCheck = part = isXML ? part : part.toUpperCase();

checkFn = dirNodeCheck;

}

checkFn("parentNode", part, doneName, checkSet, nodeCheck, isXML);

},

"~": function(checkSet, part, isXML){

var doneName = done++, checkFn = dirCheck;

if ( typeof part === "string" && !part.match(/\W/) ) {

var nodeCheck = part = isXML ? part : part.toUpperCase();

checkFn = dirNodeCheck;

}

checkFn("previousSibling", part, doneName, checkSet, nodeCheck, isXML);

}

},

find: {

ID: function(match, context, isXML){

if ( typeof context.getElementById !== "undefined" && !isXML ) {

var m = context.getElementById(match[1]);

return m ? [m] : [];

}

},

NAME: function(match, context, isXML){

if ( typeof context.getElementsByName !== "undefined" ) {

var ret = [], results = context.getElementsByName(match[1]);

for ( var i = 0, l = results.length; i < l; i++ ) {

if ( results[i].getAttribute("name") === match[1] ) {

ret.push( results[i] );

}

}

return ret.length === 0 ? null : ret;

}

},

TAG: function(match, context){

return context.getElementsByTagName(match[1]);

}

},

preFilter: {

CLASS: function(match, curLoop, inplace, result, not, isXML){

match = " " + match[1].replace(/\\/g, "") + " ";

if ( isXML ) {

return match;

}

for ( var i = 0, elem; (elem = curLoop[i]) != null; i++ ) {

if ( elem ) {

if ( not ^ (elem.className && (" " + elem.className + " ").indexOf(match) >= 0) ) {

if ( !inplace )

result.push( elem );

} else if ( inplace ) {

curLoop[i] = false;

}

}

}

return false;

},

ID: function(match){

return match[1].replace(/\\/g, "");

},

TAG: function(match, curLoop){

for ( var i = 0; curLoop[i] === false; i++ ){}

return curLoop[i] && isXML(curLoop[i]) ? match[1] : match[1].toUpperCase();

},

CHILD: function(match){

if ( match[1] == "nth" ) {

// parse equations like 'even', 'odd', '5', '2n', '3n+2', '4n-1', '-n+6'

var test = /(-?)(\d\*)n((?:\+|-)?\d\*)/.exec(

match[2] == "even" && "2n" || match[2] == "odd" && "2n+1" ||

!/\D/.test( match[2] ) && "0n+" + match[2] || match[2]);

// calculate the numbers (first)n+(last) including if they are negative

match[2] = (test[1] + (test[2] || 1)) - 0;

match[3] = test[3] - 0;

}

// TODO: Move to normal caching system

match[0] = done++;

return match;

},

ATTR: function(match, curLoop, inplace, result, not, isXML){

var name = match[1].replace(/\\/g, "");

if ( !isXML && Expr.attrMap[name] ) {

match[1] = Expr.attrMap[name];

}

if ( match[2] === "~=" ) {

match[4] = " " + match[4] + " ";

}

return match;

},

PSEUDO: function(match, curLoop, inplace, result, not){

if ( match[1] === "not" ) {

// If we're dealing with a complex expression, or a simple one

if ( match[3].match(chunker).length > 1 || /^\w/.test(match[3]) ) {

match[3] = Sizzle(match[3], null, null, curLoop);

} else {

var ret = Sizzle.filter(match[3], curLoop, inplace, true ^ not);

if ( !inplace ) {

result.push.apply( result, ret );

}

return false;

}

} else if ( Expr.match.POS.test( match[0] ) || Expr.match.CHILD.test( match[0] ) ) {

return true;

}

return match;

},

POS: function(match){

match.unshift( true );

return match;

}

},

filters: {

enabled: function(elem){

return elem.disabled === false && elem.type !== "hidden";

},

disabled: function(elem){

return elem.disabled === true;

},

checked: function(elem){

return elem.checked === true;

},

selected: function(elem){

// Accessing this property makes selected-by-default

// options in Safari work properly

elem.parentNode.selectedIndex;

return elem.selected === true;

},

parent: function(elem){

return !!elem.firstChild;

},

empty: function(elem){

return !elem.firstChild;

},

has: function(elem, i, match){

return !!Sizzle( match[3], elem ).length;

},

header: function(elem){

return /h\d/i.test( elem.nodeName );

},

text: function(elem){

return "text" === elem.type;

},

radio: function(elem){

return "radio" === elem.type;

},

checkbox: function(elem){

return "checkbox" === elem.type;

},

file: function(elem){

return "file" === elem.type;

},

password: function(elem){

return "password" === elem.type;

},

submit: function(elem){

return "submit" === elem.type;

},

image: function(elem){

return "image" === elem.type;

},

reset: function(elem){

return "reset" === elem.type;

},

button: function(elem){

return "button" === elem.type || elem.nodeName.toUpperCase() === "BUTTON";

},

input: function(elem){

return /input|select|textarea|button/i.test(elem.nodeName);

}

},

setFilters: {

first: function(elem, i){

return i === 0;

},

last: function(elem, i, match, array){

return i === array.length - 1;

},

even: function(elem, i){

return i % 2 === 0;

},

odd: function(elem, i){

return i % 2 === 1;

},

lt: function(elem, i, match){

return i < match[3] - 0;

},

gt: function(elem, i, match){

return i > match[3] - 0;

},

nth: function(elem, i, match){

return match[3] - 0 == i;

},

eq: function(elem, i, match){

return match[3] - 0 == i;

}

},

filter: {

PSEUDO: function(elem, match, i, array){

var name = match[1], filter = Expr.filters[ name ];

if ( filter ) {

return filter( elem, i, match, array );

} else if ( name === "contains" ) {

return (elem.textContent || elem.innerText || "").indexOf(match[3]) >= 0;

} else if ( name === "not" ) {

var not = match[3];

for ( var i = 0, l = not.length; i < l; i++ ) {

if ( not[i] === elem ) {

return false;

}

}

return true;

}

},

CHILD: function(elem, match){

var type = match[1], node = elem;

switch (type) {

case 'only':

case 'first':

while (node = node.previousSibling) {

if ( node.nodeType === 1 ) return false;

}

if ( type == 'first') return true;

node = elem;

case 'last':

while (node = node.nextSibling) {

if ( node.nodeType === 1 ) return false;

}

return true;

case 'nth':

var first = match[2], last = match[3];

if ( first == 1 && last == 0 ) {

return true;

}

var doneName = match[0],

parent = elem.parentNode;

if ( parent && (parent.sizcache !== doneName || !elem.nodeIndex) ) {

var count = 0;

for ( node = parent.firstChild; node; node = node.nextSibling ) {

if ( node.nodeType === 1 ) {

node.nodeIndex = ++count;

}

}

parent.sizcache = doneName;

}

var diff = elem.nodeIndex - last;

if ( first == 0 ) {

return diff == 0;

} else {

return ( diff % first == 0 && diff / first >= 0 );

}

}

},

ID: function(elem, match){

return elem.nodeType === 1 && elem.getAttribute("id") === match;

},

TAG: function(elem, match){

return (match === "\*" && elem.nodeType === 1) || elem.nodeName === match;

},

CLASS: function(elem, match){

return (" " + (elem.className || elem.getAttribute("class")) + " ")

.indexOf( match ) > -1;

},

ATTR: function(elem, match){

var name = match[1],

result = Expr.attrHandle[ name ] ?

Expr.attrHandle[ name ]( elem ) :

elem[ name ] != null ?

elem[ name ] :

elem.getAttribute( name ),

value = result + "",

type = match[2],

check = match[4];

return result == null ?

type === "!=" :

type === "=" ?

value === check :

type === "\*=" ?

value.indexOf(check) >= 0 :

type === "~=" ?

(" " + value + " ").indexOf(check) >= 0 :

!check ?

value && result !== false :

type === "!=" ?

value != check :

type === "^=" ?

value.indexOf(check) === 0 :

type === "$=" ?

value.substr(value.length - check.length) === check :

type === "|=" ?

value === check || value.substr(0, check.length + 1) === check + "-" :

false;

},

POS: function(elem, match, i, array){

var name = match[2], filter = Expr.setFilters[ name ];

if ( filter ) {

return filter( elem, i, match, array );

}

}

}

};

var origPOS = Expr.match.POS;

for ( var type in Expr.match ) {

Expr.match[ type ] = RegExp( Expr.match[ type ].source + /(?![^\[]\*\])(?![^\(]\*\))/.source );

}

var makeArray = function(array, results) {

array = Array.prototype.slice.call( array );

if ( results ) {

results.push.apply( results, array );

return results;

}

return array;

};

// Perform a simple check to determine if the browser is capable of

// converting a NodeList to an array using builtin methods.

try {

Array.prototype.slice.call( document.documentElement.childNodes );

// Provide a fallback method if it does not work

} catch(e){

makeArray = function(array, results) {

var ret = results || [];

if ( toString.call(array) === "[object Array]" ) {

Array.prototype.push.apply( ret, array );

} else {

if ( typeof array.length === "number" ) {

for ( var i = 0, l = array.length; i < l; i++ ) {

ret.push( array[i] );

}

} else {

for ( var i = 0; array[i]; i++ ) {

ret.push( array[i] );

}

}

}

return ret;

};

}

var sortOrder;

if ( document.documentElement.compareDocumentPosition ) {

sortOrder = function( a, b ) {

var ret = a.compareDocumentPosition(b) & 4 ? -1 : a === b ? 0 : 1;

if ( ret === 0 ) {

hasDuplicate = true;

}

return ret;

};

} else if ( "sourceIndex" in document.documentElement ) {

sortOrder = function( a, b ) {

var ret = a.sourceIndex - b.sourceIndex;

if ( ret === 0 ) {

hasDuplicate = true;

}

return ret;

};

} else if ( document.createRange ) {

sortOrder = function( a, b ) {

var aRange = a.ownerDocument.createRange(), bRange = b.ownerDocument.createRange();

aRange.selectNode(a);

aRange.collapse(true);

bRange.selectNode(b);

bRange.collapse(true);

var ret = aRange.compareBoundaryPoints(Range.START\_TO\_END, bRange);

if ( ret === 0 ) {

hasDuplicate = true;

}

return ret;

};

}

// Check to see if the browser returns elements by name when

// querying by getElementById (and provide a workaround)

(function(){

// We're going to inject a fake input element with a specified name

var form = document.createElement("form"),

id = "script" + (new Date).getTime();

form.innerHTML = "<input name='" + id + "'/>";

// Inject it into the root element, check its status, and remove it quickly

var root = document.documentElement;

root.insertBefore( form, root.firstChild );

// The workaround has to do additional checks after a getElementById

// Which slows things down for other browsers (hence the branching)

if ( !!document.getElementById( id ) ) {

Expr.find.ID = function(match, context, isXML){

if ( typeof context.getElementById !== "undefined" && !isXML ) {

var m = context.getElementById(match[1]);

return m ? m.id === match[1] || typeof m.getAttributeNode !== "undefined" && m.getAttributeNode("id").nodeValue === match[1] ? [m] : undefined : [];

}

};

Expr.filter.ID = function(elem, match){

var node = typeof elem.getAttributeNode !== "undefined" && elem.getAttributeNode("id");

return elem.nodeType === 1 && node && node.nodeValue === match;

};

}

root.removeChild( form );

})();

(function(){

// Check to see if the browser returns only elements

// when doing getElementsByTagName("\*")

// Create a fake element

var div = document.createElement("div");

div.appendChild( document.createComment("") );

// Make sure no comments are found

if ( div.getElementsByTagName("\*").length > 0 ) {

Expr.find.TAG = function(match, context){

var results = context.getElementsByTagName(match[1]);

// Filter out possible comments

if ( match[1] === "\*" ) {

var tmp = [];

for ( var i = 0; results[i]; i++ ) {

if ( results[i].nodeType === 1 ) {

tmp.push( results[i] );

}

}

results = tmp;

}

return results;

};

}

// Check to see if an attribute returns normalized href attributes

div.innerHTML = "<a href='#'></a>";

if ( div.firstChild && typeof div.firstChild.getAttribute !== "undefined" &&

div.firstChild.getAttribute("href") !== "#" ) {

Expr.attrHandle.href = function(elem){

return elem.getAttribute("href", 2);

};

}

})();

if ( document.querySelectorAll ) (function(){

var oldSizzle = Sizzle, div = document.createElement("div");

div.innerHTML = "<p class='TEST'></p>";

// Safari can't handle uppercase or unicode characters when

// in quirks mode.

if ( div.querySelectorAll && div.querySelectorAll(".TEST").length === 0 ) {

return;

}

Sizzle = function(query, context, extra, seed){

context = context || document;

// Only use querySelectorAll on non-XML documents

// (ID selectors don't work in non-HTML documents)

if ( !seed && context.nodeType === 9 && !isXML(context) ) {

try {

return makeArray( context.querySelectorAll(query), extra );

} catch(e){}

}

return oldSizzle(query, context, extra, seed);

};

Sizzle.find = oldSizzle.find;

Sizzle.filter = oldSizzle.filter;

Sizzle.selectors = oldSizzle.selectors;

Sizzle.matches = oldSizzle.matches;

})();

if ( document.getElementsByClassName && document.documentElement.getElementsByClassName ) (function(){

var div = document.createElement("div");

div.innerHTML = "<div class='test e'></div><div class='test'></div>";

// Opera can't find a second classname (in 9.6)

if ( div.getElementsByClassName("e").length === 0 )

return;

// Safari caches class attributes, doesn't catch changes (in 3.2)

div.lastChild.className = "e";

if ( div.getElementsByClassName("e").length === 1 )

return;

Expr.order.splice(1, 0, "CLASS");

Expr.find.CLASS = function(match, context, isXML) {

if ( typeof context.getElementsByClassName !== "undefined" && !isXML ) {

return context.getElementsByClassName(match[1]);

}

};

})();

function dirNodeCheck( dir, cur, doneName, checkSet, nodeCheck, isXML ) {

var sibDir = dir == "previousSibling" && !isXML;

for ( var i = 0, l = checkSet.length; i < l; i++ ) {

var elem = checkSet[i];

if ( elem ) {

if ( sibDir && elem.nodeType === 1 ){

elem.sizcache = doneName;

elem.sizset = i;

}

elem = elem[dir];

var match = false;

while ( elem ) {

if ( elem.sizcache === doneName ) {

match = checkSet[elem.sizset];

break;

}

if ( elem.nodeType === 1 && !isXML ){

elem.sizcache = doneName;

elem.sizset = i;

}

if ( elem.nodeName === cur ) {

match = elem;

break;

}

elem = elem[dir];

}

checkSet[i] = match;

}

}

}

function dirCheck( dir, cur, doneName, checkSet, nodeCheck, isXML ) {

var sibDir = dir == "previousSibling" && !isXML;

for ( var i = 0, l = checkSet.length; i < l; i++ ) {

var elem = checkSet[i];

if ( elem ) {

if ( sibDir && elem.nodeType === 1 ) {

elem.sizcache = doneName;

elem.sizset = i;

}

elem = elem[dir];

var match = false;

while ( elem ) {

if ( elem.sizcache === doneName ) {

match = checkSet[elem.sizset];

break;

}

if ( elem.nodeType === 1 ) {

if ( !isXML ) {

elem.sizcache = doneName;

elem.sizset = i;

}

if ( typeof cur !== "string" ) {

if ( elem === cur ) {

match = true;

break;

}

} else if ( Sizzle.filter( cur, [elem] ).length > 0 ) {

match = elem;

break;

}

}

elem = elem[dir];

}

checkSet[i] = match;

}

}

}

var contains = document.compareDocumentPosition ? function(a, b){

return a.compareDocumentPosition(b) & 16;

} : function(a, b){

return a !== b && (a.contains ? a.contains(b) : true);

};

var isXML = function(elem){

return elem.nodeType === 9 && elem.documentElement.nodeName !== "HTML" ||

!!elem.ownerDocument && isXML( elem.ownerDocument );

};

var posProcess = function(selector, context){

var tmpSet = [], later = "", match,

root = context.nodeType ? [context] : context;

// Position selectors must be done after the filter

// And so must :not(positional) so we move all PSEUDOs to the end

while ( (match = Expr.match.PSEUDO.exec( selector )) ) {

later += match[0];

selector = selector.replace( Expr.match.PSEUDO, "" );

}

selector = Expr.relative[selector] ? selector + "\*" : selector;

for ( var i = 0, l = root.length; i < l; i++ ) {

Sizzle( selector, root[i], tmpSet );

}

return Sizzle.filter( later, tmpSet );

};

// EXPOSE

jQuery.find = Sizzle;

jQuery.filter = Sizzle.filter;

jQuery.expr = Sizzle.selectors;

jQuery.expr[":"] = jQuery.expr.filters;

Sizzle.selectors.filters.hidden = function(elem){

return elem.offsetWidth === 0 || elem.offsetHeight === 0;

};

Sizzle.selectors.filters.visible = function(elem){

return elem.offsetWidth > 0 || elem.offsetHeight > 0;

};

Sizzle.selectors.filters.animated = function(elem){

return jQuery.grep(jQuery.timers, function(fn){

return elem === fn.elem;

}).length;

};

jQuery.multiFilter = function( expr, elems, not ) {

if ( not ) {

expr = ":not(" + expr + ")";

}

return Sizzle.matches(expr, elems);

};

jQuery.dir = function( elem, dir ){

var matched = [], cur = elem[dir];

while ( cur && cur != document ) {

if ( cur.nodeType == 1 )

matched.push( cur );

cur = cur[dir];

}

return matched;

};

jQuery.nth = function(cur, result, dir, elem){

result = result || 1;

var num = 0;

for ( ; cur; cur = cur[dir] )

if ( cur.nodeType == 1 && ++num == result )

break;

return cur;

};

jQuery.sibling = function(n, elem){

var r = [];

for ( ; n; n = n.nextSibling ) {

if ( n.nodeType == 1 && n != elem )

r.push( n );

}

return r;

};

return;

window.Sizzle = Sizzle;

})();

/\*

\* A number of helper functions used for managing events.

\* Many of the ideas behind this code originated from

\* Dean Edwards' addEvent library.

\*/

jQuery.event = {

// Bind an event to an element

// Original by Dean Edwards

add: function(elem, types, handler, data) {

if ( elem.nodeType == 3 || elem.nodeType == 8 )

return;

// For whatever reason, IE has trouble passing the window object

// around, causing it to be cloned in the process

if ( elem.setInterval && elem != window )

elem = window;

// Make sure that the function being executed has a unique ID

if ( !handler.guid )

handler.guid = this.guid++;

// if data is passed, bind to handler

if ( data !== undefined ) {

// Create temporary function pointer to original handler

var fn = handler;

// Create unique handler function, wrapped around original handler

handler = this.proxy( fn );

// Store data in unique handler

handler.data = data;

}

// Init the element's event structure

var events = jQuery.data(elem, "events") || jQuery.data(elem, "events", {}),

handle = jQuery.data(elem, "handle") || jQuery.data(elem, "handle", function(){

// Handle the second event of a trigger and when

// an event is called after a page has unloaded

return typeof jQuery !== "undefined" && !jQuery.event.triggered ?

jQuery.event.handle.apply(arguments.callee.elem, arguments) :

undefined;

});

// Add elem as a property of the handle function

// This is to prevent a memory leak with non-native

// event in IE.

handle.elem = elem;

// Handle multiple events separated by a space

// jQuery(...).bind("mouseover mouseout", fn);

jQuery.each(types.split(/\s+/), function(index, type) {

// Namespaced event handlers

var namespaces = type.split(".");

type = namespaces.shift();

handler.type = namespaces.slice().sort().join(".");

// Get the current list of functions bound to this event

var handlers = events[type];

if ( jQuery.event.specialAll[type] )

jQuery.event.specialAll[type].setup.call(elem, data, namespaces);

// Init the event handler queue

if (!handlers) {

handlers = events[type] = {};

// Check for a special event handler

// Only use addEventListener/attachEvent if the special

// events handler returns false

if ( !jQuery.event.special[type] || jQuery.event.special[type].setup.call(elem, data, namespaces) === false ) {

// Bind the global event handler to the element

if (elem.addEventListener)

elem.addEventListener(type, handle, false);

else if (elem.attachEvent)

elem.attachEvent("on" + type, handle);

}

}

// Add the function to the element's handler list

handlers[handler.guid] = handler;

// Keep track of which events have been used, for global triggering

jQuery.event.global[type] = true;

});

// Nullify elem to prevent memory leaks in IE

elem = null;

},

guid: 1,

global: {},

// Detach an event or set of events from an element

remove: function(elem, types, handler) {

// don't do events on text and comment nodes

if ( elem.nodeType == 3 || elem.nodeType == 8 )

return;

var events = jQuery.data(elem, "events"), ret, index;

if ( events ) {

// Unbind all events for the element

if ( types === undefined || (typeof types === "string" && types.charAt(0) == ".") )

for ( var type in events )

this.remove( elem, type + (types || "") );

else {

// types is actually an event object here

if ( types.type ) {

handler = types.handler;

types = types.type;

}

// Handle multiple events seperated by a space

// jQuery(...).unbind("mouseover mouseout", fn);

jQuery.each(types.split(/\s+/), function(index, type){

// Namespaced event handlers

var namespaces = type.split(".");

type = namespaces.shift();

var namespace = RegExp("(^|\\.)" + namespaces.slice().sort().join(".\*\\.") + "(\\.|$)");

if ( events[type] ) {

// remove the given handler for the given type

if ( handler )

delete events[type][handler.guid];

// remove all handlers for the given type

else

for ( var handle in events[type] )

// Handle the removal of namespaced events

if ( namespace.test(events[type][handle].type) )

delete events[type][handle];

if ( jQuery.event.specialAll[type] )

jQuery.event.specialAll[type].teardown.call(elem, namespaces);

// remove generic event handler if no more handlers exist

for ( ret in events[type] ) break;

if ( !ret ) {

if ( !jQuery.event.special[type] || jQuery.event.special[type].teardown.call(elem, namespaces) === false ) {

if (elem.removeEventListener)

elem.removeEventListener(type, jQuery.data(elem, "handle"), false);

else if (elem.detachEvent)

elem.detachEvent("on" + type, jQuery.data(elem, "handle"));

}

ret = null;

delete events[type];

}

}

});

}

// Remove the expando if it's no longer used

for ( ret in events ) break;

if ( !ret ) {

var handle = jQuery.data( elem, "handle" );

if ( handle ) handle.elem = null;

jQuery.removeData( elem, "events" );

jQuery.removeData( elem, "handle" );

}

}

},

// bubbling is internal

trigger: function( event, data, elem, bubbling ) {

// Event object or event type

var type = event.type || event;

if( !bubbling ){

event = typeof event === "object" ?

// jQuery.Event object

event[expando] ? event :

// Object literal

jQuery.extend( jQuery.Event(type), event ) :

// Just the event type (string)

jQuery.Event(type);

if ( type.indexOf("!") >= 0 ) {

event.type = type = type.slice(0, -1);

event.exclusive = true;

}

// Handle a global trigger

if ( !elem ) {

// Don't bubble custom events when global (to avoid too much overhead)

event.stopPropagation();

// Only trigger if we've ever bound an event for it

if ( this.global[type] )

jQuery.each( jQuery.cache, function(){

if ( this.events && this.events[type] )

jQuery.event.trigger( event, data, this.handle.elem );

});

}

// Handle triggering a single element

// don't do events on text and comment nodes

if ( !elem || elem.nodeType == 3 || elem.nodeType == 8 )

return undefined;

// Clean up in case it is reused

event.result = undefined;

event.target = elem;

// Clone the incoming data, if any

data = jQuery.makeArray(data);

data.unshift( event );

}

event.currentTarget = elem;

// Trigger the event, it is assumed that "handle" is a function

var handle = jQuery.data(elem, "handle");

if ( handle )

handle.apply( elem, data );

// Handle triggering native .onfoo handlers (and on links since we don't call .click() for links)

if ( (!elem[type] || (jQuery.nodeName(elem, 'a') && type == "click")) && elem["on"+type] && elem["on"+type].apply( elem, data ) === false )

event.result = false;

// Trigger the native events (except for clicks on links)

if ( !bubbling && elem[type] && !event.isDefaultPrevented() && !(jQuery.nodeName(elem, 'a') && type == "click") ) {

this.triggered = true;

try {

elem[ type ]();

// prevent IE from throwing an error for some hidden elements

} catch (e) {}

}

this.triggered = false;

if ( !event.isPropagationStopped() ) {

var parent = elem.parentNode || elem.ownerDocument;

if ( parent )

jQuery.event.trigger(event, data, parent, true);

}

},

handle: function(event) {

// returned undefined or false

var all, handlers;

event = arguments[0] = jQuery.event.fix( event || window.event );

event.currentTarget = this;

// Namespaced event handlers

var namespaces = event.type.split(".");

event.type = namespaces.shift();

// Cache this now, all = true means, any handler

all = !namespaces.length && !event.exclusive;

var namespace = RegExp("(^|\\.)" + namespaces.slice().sort().join(".\*\\.") + "(\\.|$)");

handlers = ( jQuery.data(this, "events") || {} )[event.type];

for ( var j in handlers ) {

var handler = handlers[j];

// Filter the functions by class

if ( all || namespace.test(handler.type) ) {

// Pass in a reference to the handler function itself

// So that we can later remove it

event.handler = handler;

event.data = handler.data;

var ret = handler.apply(this, arguments);

if( ret !== undefined ){

event.result = ret;

if ( ret === false ) {

event.preventDefault();

event.stopPropagation();

}

}

if( event.isImmediatePropagationStopped() )

break;

}

}

},

props: "altKey attrChange attrName bubbles button cancelable charCode clientX clientY ctrlKey currentTarget data detail eventPhase fromElement handler keyCode metaKey newValue originalTarget pageX pageY prevValue relatedNode relatedTarget screenX screenY shiftKey srcElement target toElement view wheelDelta which".split(" "),

fix: function(event) {

if ( event[expando] )

return event;

// store a copy of the original event object

// and "clone" to set read-only properties

var originalEvent = event;

event = jQuery.Event( originalEvent );

for ( var i = this.props.length, prop; i; ){

prop = this.props[ --i ];

event[ prop ] = originalEvent[ prop ];

}

// Fix target property, if necessary

if ( !event.target )

event.target = event.srcElement || document; // Fixes #1925 where srcElement might not be defined either

// check if target is a textnode (safari)

if ( event.target.nodeType == 3 )

event.target = event.target.parentNode;

// Add relatedTarget, if necessary

if ( !event.relatedTarget && event.fromElement )

event.relatedTarget = event.fromElement == event.target ? event.toElement : event.fromElement;

// Calculate pageX/Y if missing and clientX/Y available

if ( event.pageX == null && event.clientX != null ) {

var doc = document.documentElement, body = document.body;

event.pageX = event.clientX + (doc && doc.scrollLeft || body && body.scrollLeft || 0) - (doc.clientLeft || 0);

event.pageY = event.clientY + (doc && doc.scrollTop || body && body.scrollTop || 0) - (doc.clientTop || 0);

}

// Add which for key events

if ( !event.which && ((event.charCode || event.charCode === 0) ? event.charCode : event.keyCode) )

event.which = event.charCode || event.keyCode;

// Add metaKey to non-Mac browsers (use ctrl for PC's and Meta for Macs)

if ( !event.metaKey && event.ctrlKey )

event.metaKey = event.ctrlKey;

// Add which for click: 1 == left; 2 == middle; 3 == right

// Note: button is not normalized, so don't use it

if ( !event.which && event.button )

event.which = (event.button & 1 ? 1 : ( event.button & 2 ? 3 : ( event.button & 4 ? 2 : 0 ) ));

return event;

},

proxy: function( fn, proxy ){

proxy = proxy || function(){ return fn.apply(this, arguments); };

// Set the guid of unique handler to the same of original handler, so it can be removed

proxy.guid = fn.guid = fn.guid || proxy.guid || this.guid++;

// So proxy can be declared as an argument

return proxy;

},

special: {

ready: {

// Make sure the ready event is setup

setup: bindReady,

teardown: function() {}

}

},

specialAll: {

live: {

setup: function( selector, namespaces ){

jQuery.event.add( this, namespaces[0], liveHandler );

},

teardown: function( namespaces ){

if ( namespaces.length ) {

var remove = 0, name = RegExp("(^|\\.)" + namespaces[0] + "(\\.|$)");

jQuery.each( (jQuery.data(this, "events").live || {}), function(){

if ( name.test(this.type) )

remove++;

});

if ( remove < 1 )

jQuery.event.remove( this, namespaces[0], liveHandler );

}

}

}

}

};

jQuery.Event = function( src ){

// Allow instantiation without the 'new' keyword

if( !this.preventDefault )

return new jQuery.Event(src);

// Event object

if( src && src.type ){

this.originalEvent = src;

this.type = src.type;

// Event type

}else

this.type = src;

// timeStamp is buggy for some events on Firefox(#3843)

// So we won't rely on the native value

this.timeStamp = now();

// Mark it as fixed

this[expando] = true;

};

function returnFalse(){

return false;

}

function returnTrue(){

return true;

}

// jQuery.Event is based on DOM3 Events as specified by the ECMAScript Language Binding

// http://www.w3.org/TR/2003/WD-DOM-Level-3-Events-20030331/ecma-script-binding.html

jQuery.Event.prototype = {

preventDefault: function() {

this.isDefaultPrevented = returnTrue;

var e = this.originalEvent;

if( !e )

return;

// if preventDefault exists run it on the original event

if (e.preventDefault)

e.preventDefault();

// otherwise set the returnValue property of the original event to false (IE)

e.returnValue = false;

},

stopPropagation: function() {

this.isPropagationStopped = returnTrue;

var e = this.originalEvent;

if( !e )

return;

// if stopPropagation exists run it on the original event

if (e.stopPropagation)

e.stopPropagation();

// otherwise set the cancelBubble property of the original event to true (IE)

e.cancelBubble = true;

},

stopImmediatePropagation:function(){

this.isImmediatePropagationStopped = returnTrue;

this.stopPropagation();

},

isDefaultPrevented: returnFalse,

isPropagationStopped: returnFalse,

isImmediatePropagationStopped: returnFalse

};

// Checks if an event happened on an element within another element

// Used in jQuery.event.special.mouseenter and mouseleave handlers

var withinElement = function(event) {

// Check if mouse(over|out) are still within the same parent element

var parent = event.relatedTarget;

// Traverse up the tree

while ( parent && parent != this )

try { parent = parent.parentNode; }

catch(e) { parent = this; }

if( parent != this ){

// set the correct event type

event.type = event.data;

// handle event if we actually just moused on to a non sub-element

jQuery.event.handle.apply( this, arguments );

}

};

jQuery.each({

mouseover: 'mouseenter',

mouseout: 'mouseleave'

}, function( orig, fix ){

jQuery.event.special[ fix ] = {

setup: function(){

jQuery.event.add( this, orig, withinElement, fix );

},

teardown: function(){

jQuery.event.remove( this, orig, withinElement );

}

};

});

jQuery.fn.extend({

bind: function( type, data, fn ) {

return type == "unload" ? this.one(type, data, fn) : this.each(function(){

jQuery.event.add( this, type, fn || data, fn && data );

});

},

one: function( type, data, fn ) {

var one = jQuery.event.proxy( fn || data, function(event) {

jQuery(this).unbind(event, one);

return (fn || data).apply( this, arguments );

});

return this.each(function(){

jQuery.event.add( this, type, one, fn && data);

});

},

unbind: function( type, fn ) {

return this.each(function(){

jQuery.event.remove( this, type, fn );

});

},

trigger: function( type, data ) {

return this.each(function(){

jQuery.event.trigger( type, data, this );

});

},

triggerHandler: function( type, data ) {

if( this[0] ){

var event = jQuery.Event(type);

event.preventDefault();

event.stopPropagation();

jQuery.event.trigger( event, data, this[0] );

return event.result;

}

},

toggle: function( fn ) {

// Save reference to arguments for access in closure

var args = arguments, i = 1;

// link all the functions, so any of them can unbind this click handler

while( i < args.length )

jQuery.event.proxy( fn, args[i++] );

return this.click( jQuery.event.proxy( fn, function(event) {

// Figure out which function to execute

this.lastToggle = ( this.lastToggle || 0 ) % i;

// Make sure that clicks stop

event.preventDefault();

// and execute the function

return args[ this.lastToggle++ ].apply( this, arguments ) || false;

}));

},

hover: function(fnOver, fnOut) {

return this.mouseenter(fnOver).mouseleave(fnOut);

},

ready: function(fn) {

// Attach the listeners

bindReady();

// If the DOM is already ready

if ( jQuery.isReady )

// Execute the function immediately

fn.call( document, jQuery );

// Otherwise, remember the function for later

else

// Add the function to the wait list

jQuery.readyList.push( fn );

return this;

},

live: function( type, fn ){

var proxy = jQuery.event.proxy( fn );

proxy.guid += this.selector + type;

jQuery(document).bind( liveConvert(type, this.selector), this.selector, proxy );

return this;

},

die: function( type, fn ){

jQuery(document).unbind( liveConvert(type, this.selector), fn ? { guid: fn.guid + this.selector + type } : null );

return this;

}

});

function liveHandler( event ){

var check = RegExp("(^|\\.)" + event.type + "(\\.|$)"),

stop = true,

elems = [];

jQuery.each(jQuery.data(this, "events").live || [], function(i, fn){

if ( check.test(fn.type) ) {

var elem = jQuery(event.target).closest(fn.data)[0];

if ( elem )

elems.push({ elem: elem, fn: fn });

}

});

elems.sort(function(a,b) {

return jQuery.data(a.elem, "closest") - jQuery.data(b.elem, "closest");

});

jQuery.each(elems, function(){

if ( this.fn.call(this.elem, event, this.fn.data) === false )

return (stop = false);

});

return stop;

}

function liveConvert(type, selector){

return ["live", type, selector.replace(/\./g, "`").replace(/ /g, "|")].join(".");

}

jQuery.extend({

isReady: false,

readyList: [],

// Handle when the DOM is ready

ready: function() {

// Make sure that the DOM is not already loaded

if ( !jQuery.isReady ) {

// Remember that the DOM is ready

jQuery.isReady = true;

// If there are functions bound, to execute

if ( jQuery.readyList ) {

// Execute all of them

jQuery.each( jQuery.readyList, function(){

this.call( document, jQuery );

});

// Reset the list of functions

jQuery.readyList = null;

}

// Trigger any bound ready events

jQuery(document).triggerHandler("ready");

}

}

});

var readyBound = false;

function bindReady(){

if ( readyBound ) return;

readyBound = true;

// Mozilla, Opera and webkit nightlies currently support this event

if ( document.addEventListener ) {

// Use the handy event callback

document.addEventListener( "DOMContentLoaded", function(){

document.removeEventListener( "DOMContentLoaded", arguments.callee, false );

jQuery.ready();

}, false );

// If IE event model is used

} else if ( document.attachEvent ) {

// ensure firing before onload,

// maybe late but safe also for iframes

document.attachEvent("onreadystatechange", function(){

if ( document.readyState === "complete" ) {

document.detachEvent( "onreadystatechange", arguments.callee );

jQuery.ready();

}

});

// If IE and not an iframe

// continually check to see if the document is ready

if ( document.documentElement.doScroll && window == window.top ) (function(){

if ( jQuery.isReady ) return;

try {

// If IE is used, use the trick by Diego Perini

// http://javascript.nwbox.com/IEContentLoaded/

document.documentElement.doScroll("left");

} catch( error ) {

setTimeout( arguments.callee, 0 );

return;

}

// and execute any waiting functions

jQuery.ready();

})();

}

// A fallback to window.onload, that will always work

jQuery.event.add( window, "load", jQuery.ready );

}

jQuery.each( ("blur,focus,load,resize,scroll,unload,click,dblclick," +

"mousedown,mouseup,mousemove,mouseover,mouseout,mouseenter,mouseleave," +

"change,select,submit,keydown,keypress,keyup,error").split(","), function(i, name){

// Handle event binding

jQuery.fn[name] = function(fn){

return fn ? this.bind(name, fn) : this.trigger(name);

};

});

// Prevent memory leaks in IE

// And prevent errors on refresh with events like mouseover in other browsers

// Window isn't included so as not to unbind existing unload events

jQuery( window ).bind( 'unload', function(){

for ( var id in jQuery.cache )

// Skip the window

if ( id != 1 && jQuery.cache[ id ].handle )

jQuery.event.remove( jQuery.cache[ id ].handle.elem );

});

(function(){

jQuery.support = {};

var root = document.documentElement,

script = document.createElement("script"),

div = document.createElement("div"),

id = "script" + (new Date).getTime();

div.style.display = "none";

div.innerHTML = ' <link/><table></table><a href="/a" style="color:red;float:left;opacity:.5;">a</a><select><option>text</option></select><object><param/></object>';

var all = div.getElementsByTagName("\*"),

a = div.getElementsByTagName("a")[0];

// Can't get basic test support

if ( !all || !all.length || !a ) {

return;

}

jQuery.support = {

// IE strips leading whitespace when .innerHTML is used

leadingWhitespace: div.firstChild.nodeType == 3,

// Make sure that tbody elements aren't automatically inserted

// IE will insert them into empty tables

tbody: !div.getElementsByTagName("tbody").length,

// Make sure that you can get all elements in an <object> element

// IE 7 always returns no results

objectAll: !!div.getElementsByTagName("object")[0]

.getElementsByTagName("\*").length,

// Make sure that link elements get serialized correctly by innerHTML

// This requires a wrapper element in IE

htmlSerialize: !!div.getElementsByTagName("link").length,

// Get the style information from getAttribute

// (IE uses .cssText insted)

style: /red/.test( a.getAttribute("style") ),

// Make sure that URLs aren't manipulated

// (IE normalizes it by default)

hrefNormalized: a.getAttribute("href") === "/a",

// Make sure that element opacity exists

// (IE uses filter instead)

opacity: a.style.opacity === "0.5",

// Verify style float existence

// (IE uses styleFloat instead of cssFloat)

cssFloat: !!a.style.cssFloat,

// Will be defined later

scriptEval: false,

noCloneEvent: true,

boxModel: null

};

script.type = "text/javascript";

try {

script.appendChild( document.createTextNode( "window." + id + "=1;" ) );

} catch(e){}

root.insertBefore( script, root.firstChild );

// Make sure that the execution of code works by injecting a script

// tag with appendChild/createTextNode

// (IE doesn't support this, fails, and uses .text instead)

if ( window[ id ] ) {

jQuery.support.scriptEval = true;

delete window[ id ];

}

root.removeChild( script );

if ( div.attachEvent && div.fireEvent ) {

div.attachEvent("onclick", function(){

// Cloning a node shouldn't copy over any

// bound event handlers (IE does this)

jQuery.support.noCloneEvent = false;

div.detachEvent("onclick", arguments.callee);

});

div.cloneNode(true).fireEvent("onclick");

}

// Figure out if the W3C box model works as expected

// document.body must exist before we can do this

jQuery(function(){

var div = document.createElement("div");

div.style.width = div.style.paddingLeft = "1px";

document.body.appendChild( div );

jQuery.boxModel = jQuery.support.boxModel = div.offsetWidth === 2;

document.body.removeChild( div ).style.display = 'none';

});

})();

var styleFloat = jQuery.support.cssFloat ? "cssFloat" : "styleFloat";

jQuery.props = {

"for": "htmlFor",

"class": "className",

"float": styleFloat,

cssFloat: styleFloat,

styleFloat: styleFloat,

readonly: "readOnly",

maxlength: "maxLength",

cellspacing: "cellSpacing",

rowspan: "rowSpan",

tabindex: "tabIndex"

};

jQuery.fn.extend({

// Keep a copy of the old load

\_load: jQuery.fn.load,

load: function( url, params, callback ) {

if ( typeof url !== "string" )

return this.\_load( url );

var off = url.indexOf(" ");

if ( off >= 0 ) {

var selector = url.slice(off, url.length);

url = url.slice(0, off);

}

// Default to a GET request

var type = "GET";

// If the second parameter was provided

if ( params )

// If it's a function

if ( jQuery.isFunction( params ) ) {

// We assume that it's the callback

callback = params;

params = null;

// Otherwise, build a param string

} else if( typeof params === "object" ) {

params = jQuery.param( params );

type = "POST";

}

var self = this;

// Request the remote document

jQuery.ajax({

url: url,

type: type,

dataType: "html",

data: params,

complete: function(res, status){

// If successful, inject the HTML into all the matched elements

if ( status == "success" || status == "notmodified" )

// See if a selector was specified

self.html( selector ?

// Create a dummy div to hold the results

jQuery("<div/>")

// inject the contents of the document in, removing the scripts

// to avoid any 'Permission Denied' errors in IE

.append(res.responseText.replace(/<script(.|\s)\*?\/script>/g, ""))

// Locate the specified elements

.find(selector) :

// If not, just inject the full result

res.responseText );

if( callback )

self.each( callback, [res.responseText, status, res] );

}

});

return this;

},

serialize: function() {

return jQuery.param(this.serializeArray());

},

serializeArray: function() {

return this.map(function(){

return this.elements ? jQuery.makeArray(this.elements) : this;

})

.filter(function(){

return this.name && !this.disabled &&

(this.checked || /select|textarea/i.test(this.nodeName) ||

/text|hidden|password|search/i.test(this.type));

})

.map(function(i, elem){

var val = jQuery(this).val();

return val == null ? null :

jQuery.isArray(val) ?

jQuery.map( val, function(val, i){

return {name: elem.name, value: val};

}) :

{name: elem.name, value: val};

}).get();

}

});

// Attach a bunch of functions for handling common AJAX events

jQuery.each( "ajaxStart,ajaxStop,ajaxComplete,ajaxError,ajaxSuccess,ajaxSend".split(","), function(i,o){

jQuery.fn[o] = function(f){

return this.bind(o, f);

};

});

var jsc = now();

jQuery.extend({

get: function( url, data, callback, type ) {

// shift arguments if data argument was ommited

if ( jQuery.isFunction( data ) ) {

callback = data;

data = null;

}

return jQuery.ajax({

type: "GET",

url: url,

data: data,

success: callback,

dataType: type

});

},

getScript: function( url, callback ) {

return jQuery.get(url, null, callback, "script");

},

getJSON: function( url, data, callback ) {

return jQuery.get(url, data, callback, "json");

},

post: function( url, data, callback, type ) {

if ( jQuery.isFunction( data ) ) {

callback = data;

data = {};

}

return jQuery.ajax({

type: "POST",

url: url,

data: data,

success: callback,

dataType: type

});

},

ajaxSetup: function( settings ) {

jQuery.extend( jQuery.ajaxSettings, settings );

},

ajaxSettings: {

url: location.href,

global: true,

type: "GET",

contentType: "application/x-www-form-urlencoded",

processData: true,

async: true,

/\*

timeout: 0,

data: null,

username: null,

password: null,

\*/

// Create the request object; Microsoft failed to properly

// implement the XMLHttpRequest in IE7, so we use the ActiveXObject when it is available

// This function can be overriden by calling jQuery.ajaxSetup

xhr:function(){

return window.ActiveXObject ? new ActiveXObject("Microsoft.XMLHTTP") : new XMLHttpRequest();

},

accepts: {

xml: "application/xml, text/xml",

html: "text/html",

script: "text/javascript, application/javascript",

json: "application/json, text/javascript",

text: "text/plain",

\_default: "\*/\*"

}

},

// Last-Modified header cache for next request

lastModified: {},

ajax: function( s ) {

// Extend the settings, but re-extend 's' so that it can be

// checked again later (in the test suite, specifically)

s = jQuery.extend(true, s, jQuery.extend(true, {}, jQuery.ajaxSettings, s));

var jsonp, jsre = /=\?(&|$)/g, status, data,

type = s.type.toUpperCase();

// convert data if not already a string

if ( s.data && s.processData && typeof s.data !== "string" )

s.data = jQuery.param(s.data);

// Handle JSONP Parameter Callbacks

if ( s.dataType == "jsonp" ) {

if ( type == "GET" ) {

if ( !s.url.match(jsre) )

s.url += (s.url.match(/\?/) ? "&" : "?") + (s.jsonp || "callback") + "=?";

} else if ( !s.data || !s.data.match(jsre) )

s.data = (s.data ? s.data + "&" : "") + (s.jsonp || "callback") + "=?";

s.dataType = "json";

}

// Build temporary JSONP function

if ( s.dataType == "json" && (s.data && s.data.match(jsre) || s.url.match(jsre)) ) {

jsonp = "jsonp" + jsc++;

// Replace the =? sequence both in the query string and the data

if ( s.data )

s.data = (s.data + "").replace(jsre, "=" + jsonp + "$1");

s.url = s.url.replace(jsre, "=" + jsonp + "$1");

// We need to make sure

// that a JSONP style response is executed properly

s.dataType = "script";

// Handle JSONP-style loading

window[ jsonp ] = function(tmp){

data = tmp;

success();

complete();

// Garbage collect

window[ jsonp ] = undefined;

try{ delete window[ jsonp ]; } catch(e){}

if ( head )

head.removeChild( script );

};

}

if ( s.dataType == "script" && s.cache == null )

s.cache = false;

if ( s.cache === false && type == "GET" ) {

var ts = now();

// try replacing \_= if it is there

var ret = s.url.replace(/(\?|&)\_=.\*?(&|$)/, "$1\_=" + ts + "$2");

// if nothing was replaced, add timestamp to the end

s.url = ret + ((ret == s.url) ? (s.url.match(/\?/) ? "&" : "?") + "\_=" + ts : "");

}

// If data is available, append data to url for get requests

if ( s.data && type == "GET" ) {

s.url += (s.url.match(/\?/) ? "&" : "?") + s.data;

// IE likes to send both get and post data, prevent this

s.data = null;

}

// Watch for a new set of requests

if ( s.global && ! jQuery.active++ )

jQuery.event.trigger( "ajaxStart" );

// Matches an absolute URL, and saves the domain

var parts = /^(\w+:)?\/\/([^\/?#]+)/.exec( s.url );

// If we're requesting a remote document

// and trying to load JSON or Script with a GET

if ( s.dataType == "script" && type == "GET" && parts

&& ( parts[1] && parts[1] != location.protocol || parts[2] != location.host )){

var head = document.getElementsByTagName("head")[0];

var script = document.createElement("script");

script.src = s.url;

if (s.scriptCharset)

script.charset = s.scriptCharset;

// Handle Script loading

if ( !jsonp ) {

var done = false;

// Attach handlers for all browsers

script.onload = script.onreadystatechange = function(){

if ( !done && (!this.readyState ||

this.readyState == "loaded" || this.readyState == "complete") ) {

done = true;

success();

complete();

// Handle memory leak in IE

script.onload = script.onreadystatechange = null;

head.removeChild( script );

}

};

}

head.appendChild(script);

// We handle everything using the script element injection

return undefined;

}

var requestDone = false;

// Create the request object

var xhr = s.xhr();

// Open the socket

// Passing null username, generates a login popup on Opera (#2865)

if( s.username )

xhr.open(type, s.url, s.async, s.username, s.password);

else

xhr.open(type, s.url, s.async);

// Need an extra try/catch for cross domain requests in Firefox 3

try {

// Set the correct header, if data is being sent

if ( s.data )

xhr.setRequestHeader("Content-Type", s.contentType);

// Set the If-Modified-Since header, if ifModified mode.

if ( s.ifModified )

xhr.setRequestHeader("If-Modified-Since",

jQuery.lastModified[s.url] || "Thu, 01 Jan 1970 00:00:00 GMT" );

// Set header so the called script knows that it's an XMLHttpRequest

xhr.setRequestHeader("X-Requested-With", "XMLHttpRequest");

// Set the Accepts header for the server, depending on the dataType

xhr.setRequestHeader("Accept", s.dataType && s.accepts[ s.dataType ] ?

s.accepts[ s.dataType ] + ", \*/\*" :

s.accepts.\_default );

} catch(e){}

// Allow custom headers/mimetypes and early abort

if ( s.beforeSend && s.beforeSend(xhr, s) === false ) {

// Handle the global AJAX counter

if ( s.global && ! --jQuery.active )

jQuery.event.trigger( "ajaxStop" );

// close opended socket

xhr.abort();

return false;

}

if ( s.global )

jQuery.event.trigger("ajaxSend", [xhr, s]);

// Wait for a response to come back

var onreadystatechange = function(isTimeout){

// The request was aborted, clear the interval and decrement jQuery.active

if (xhr.readyState == 0) {

if (ival) {

// clear poll interval

clearInterval(ival);

ival = null;

// Handle the global AJAX counter

if ( s.global && ! --jQuery.active )

jQuery.event.trigger( "ajaxStop" );

}

// The transfer is complete and the data is available, or the request timed out

} else if ( !requestDone && xhr && (xhr.readyState == 4 || isTimeout == "timeout") ) {

requestDone = true;

// clear poll interval

if (ival) {

clearInterval(ival);

ival = null;

}

status = isTimeout == "timeout" ? "timeout" :

!jQuery.httpSuccess( xhr ) ? "error" :

s.ifModified && jQuery.httpNotModified( xhr, s.url ) ? "notmodified" :

"success";

if ( status == "success" ) {

// Watch for, and catch, XML document parse errors

try {

// process the data (runs the xml through httpData regardless of callback)

data = jQuery.httpData( xhr, s.dataType, s );

} catch(e) {

status = "parsererror";

}

}

// Make sure that the request was successful or notmodified

if ( status == "success" ) {

// Cache Last-Modified header, if ifModified mode.

var modRes;

try {

modRes = xhr.getResponseHeader("Last-Modified");

} catch(e) {} // swallow exception thrown by FF if header is not available

if ( s.ifModified && modRes )

jQuery.lastModified[s.url] = modRes;

// JSONP handles its own success callback

if ( !jsonp )

success();

} else

jQuery.handleError(s, xhr, status);

// Fire the complete handlers

complete();

if ( isTimeout )

xhr.abort();

// Stop memory leaks

if ( s.async )

xhr = null;

}

};

if ( s.async ) {

// don't attach the handler to the request, just poll it instead

var ival = setInterval(onreadystatechange, 13);

// Timeout checker

if ( s.timeout > 0 )

setTimeout(function(){

// Check to see if the request is still happening

if ( xhr && !requestDone )

onreadystatechange( "timeout" );

}, s.timeout);

}

// Send the data

try {

xhr.send(s.data);

} catch(e) {

jQuery.handleError(s, xhr, null, e);

}

// firefox 1.5 doesn't fire statechange for sync requests

if ( !s.async )

onreadystatechange();

function success(){

// If a local callback was specified, fire it and pass it the data

if ( s.success )

s.success( data, status );

// Fire the global callback

if ( s.global )

jQuery.event.trigger( "ajaxSuccess", [xhr, s] );

}

function complete(){

// Process result

if ( s.complete )

s.complete(xhr, status);

// The request was completed

if ( s.global )

jQuery.event.trigger( "ajaxComplete", [xhr, s] );

// Handle the global AJAX counter

if ( s.global && ! --jQuery.active )

jQuery.event.trigger( "ajaxStop" );

}

// return XMLHttpRequest to allow aborting the request etc.

return xhr;

},

handleError: function( s, xhr, status, e ) {

// If a local callback was specified, fire it

if ( s.error ) s.error( xhr, status, e );

// Fire the global callback

if ( s.global )

jQuery.event.trigger( "ajaxError", [xhr, s, e] );

},

// Counter for holding the number of active queries

active: 0,

// Determines if an XMLHttpRequest was successful or not

httpSuccess: function( xhr ) {

try {

// IE error sometimes returns 1223 when it should be 204 so treat it as success, see #1450

return !xhr.status && location.protocol == "file:" ||

( xhr.status >= 200 && xhr.status < 300 ) || xhr.status == 304 || xhr.status == 1223;

} catch(e){}

return false;

},

// Determines if an XMLHttpRequest returns NotModified

httpNotModified: function( xhr, url ) {

try {

var xhrRes = xhr.getResponseHeader("Last-Modified");

// Firefox always returns 200. check Last-Modified date

return xhr.status == 304 || xhrRes == jQuery.lastModified[url];

} catch(e){}

return false;

},

httpData: function( xhr, type, s ) {

var ct = xhr.getResponseHeader("content-type"),

xml = type == "xml" || !type && ct && ct.indexOf("xml") >= 0,

data = xml ? xhr.responseXML : xhr.responseText;

if ( xml && data.documentElement.tagName == "parsererror" )

throw "parsererror";

// Allow a pre-filtering function to sanitize the response

// s != null is checked to keep backwards compatibility

if( s && s.dataFilter )

data = s.dataFilter( data, type );

// The filter can actually parse the response

if( typeof data === "string" ){

// If the type is "script", eval it in global context

if ( type == "script" )

jQuery.globalEval( data );

// Get the JavaScript object, if JSON is used.

if ( type == "json" )

data = window["eval"]("(" + data + ")");

}

return data;

},

// Serialize an array of form elements or a set of

// key/values into a query string

param: function( a ) {

var s = [ ];

function add( key, value ){

s[ s.length ] = encodeURIComponent(key) + '=' + encodeURIComponent(value);

};

// If an array was passed in, assume that it is an array

// of form elements

if ( jQuery.isArray(a) || a.jquery )

// Serialize the form elements

jQuery.each( a, function(){

add( this.name, this.value );

});

// Otherwise, assume that it's an object of key/value pairs

else

// Serialize the key/values

for ( var j in a )

// If the value is an array then the key names need to be repeated

if ( jQuery.isArray(a[j]) )

jQuery.each( a[j], function(){

add( j, this );

});

else

add( j, jQuery.isFunction(a[j]) ? a[j]() : a[j] );

// Return the resulting serialization

return s.join("&").replace(/%20/g, "+");

}

});

var elemdisplay = {},

timerId,

fxAttrs = [

// height animations

[ "height", "marginTop", "marginBottom", "paddingTop", "paddingBottom" ],

// width animations

[ "width", "marginLeft", "marginRight", "paddingLeft", "paddingRight" ],

// opacity animations

[ "opacity" ]

];

function genFx( type, num ){

var obj = {};

jQuery.each( fxAttrs.concat.apply([], fxAttrs.slice(0,num)), function(){

obj[ this ] = type;

});

return obj;

}

jQuery.fn.extend({

show: function(speed,callback){

if ( speed ) {

return this.animate( genFx("show", 3), speed, callback);

} else {

for ( var i = 0, l = this.length; i < l; i++ ){

var old = jQuery.data(this[i], "olddisplay");

this[i].style.display = old || "";

if ( jQuery.css(this[i], "display") === "none" ) {

var tagName = this[i].tagName, display;

if ( elemdisplay[ tagName ] ) {

display = elemdisplay[ tagName ];

} else {

var elem = jQuery("<" + tagName + " />").appendTo("body");

display = elem.css("display");

if ( display === "none" )

display = "block";

elem.remove();

elemdisplay[ tagName ] = display;

}

jQuery.data(this[i], "olddisplay", display);

}

}

// Set the display of the elements in a second loop

// to avoid the constant reflow

for ( var i = 0, l = this.length; i < l; i++ ){

this[i].style.display = jQuery.data(this[i], "olddisplay") || "";

}

return this;

}

},

hide: function(speed,callback){

if ( speed ) {

return this.animate( genFx("hide", 3), speed, callback);

} else {

for ( var i = 0, l = this.length; i < l; i++ ){

var old = jQuery.data(this[i], "olddisplay");

if ( !old && old !== "none" )

jQuery.data(this[i], "olddisplay", jQuery.css(this[i], "display"));

}

// Set the display of the elements in a second loop

// to avoid the constant reflow

for ( var i = 0, l = this.length; i < l; i++ ){

this[i].style.display = "none";

}

return this;

}

},

// Save the old toggle function

\_toggle: jQuery.fn.toggle,

toggle: function( fn, fn2 ){

var bool = typeof fn === "boolean";

return jQuery.isFunction(fn) && jQuery.isFunction(fn2) ?

this.\_toggle.apply( this, arguments ) :

fn == null || bool ?

this.each(function(){

var state = bool ? fn : jQuery(this).is(":hidden");

jQuery(this)[ state ? "show" : "hide" ]();

}) :

this.animate(genFx("toggle", 3), fn, fn2);

},

fadeTo: function(speed,to,callback){

return this.animate({opacity: to}, speed, callback);

},

animate: function( prop, speed, easing, callback ) {

var optall = jQuery.speed(speed, easing, callback);

return this[ optall.queue === false ? "each" : "queue" ](function(){

var opt = jQuery.extend({}, optall), p,

hidden = this.nodeType == 1 && jQuery(this).is(":hidden"),

self = this;

for ( p in prop ) {

if ( prop[p] == "hide" && hidden || prop[p] == "show" && !hidden )

return opt.complete.call(this);

if ( ( p == "height" || p == "width" ) && this.style ) {

// Store display property

opt.display = jQuery.css(this, "display");

// Make sure that nothing sneaks out

opt.overflow = this.style.overflow;

}

}

if ( opt.overflow != null )

this.style.overflow = "hidden";

opt.curAnim = jQuery.extend({}, prop);

jQuery.each( prop, function(name, val){

var e = new jQuery.fx( self, opt, name );

if ( /toggle|show|hide/.test(val) )

e[ val == "toggle" ? hidden ? "show" : "hide" : val ]( prop );

else {

var parts = val.toString().match(/^([+-]=)?([\d+-.]+)(.\*)$/),

start = e.cur(true) || 0;

if ( parts ) {

var end = parseFloat(parts[2]),

unit = parts[3] || "px";

// We need to compute starting value

if ( unit != "px" ) {

self.style[ name ] = (end || 1) + unit;

start = ((end || 1) / e.cur(true)) \* start;

self.style[ name ] = start + unit;

}

// If a +=/-= token was provided, we're doing a relative animation

if ( parts[1] )

end = ((parts[1] == "-=" ? -1 : 1) \* end) + start;

e.custom( start, end, unit );

} else

e.custom( start, val, "" );

}

});

// For JS strict compliance

return true;

});

},

stop: function(clearQueue, gotoEnd){

var timers = jQuery.timers;

if (clearQueue)

this.queue([]);

this.each(function(){

// go in reverse order so anything added to the queue during the loop is ignored

for ( var i = timers.length - 1; i >= 0; i-- )

if ( timers[i].elem == this ) {

if (gotoEnd)

// force the next step to be the last

timers[i](true);

timers.splice(i, 1);

}

});

// start the next in the queue if the last step wasn't forced

if (!gotoEnd)

this.dequeue();

return this;

}

});

// Generate shortcuts for custom animations

jQuery.each({

slideDown: genFx("show", 1),

slideUp: genFx("hide", 1),

slideToggle: genFx("toggle", 1),

fadeIn: { opacity: "show" },

fadeOut: { opacity: "hide" }

}, function( name, props ){

jQuery.fn[ name ] = function( speed, callback ){

return this.animate( props, speed, callback );

};

});

jQuery.extend({

speed: function(speed, easing, fn) {

var opt = typeof speed === "object" ? speed : {

complete: fn || !fn && easing ||

jQuery.isFunction( speed ) && speed,

duration: speed,

easing: fn && easing || easing && !jQuery.isFunction(easing) && easing

};

opt.duration = jQuery.fx.off ? 0 : typeof opt.duration === "number" ? opt.duration :

jQuery.fx.speeds[opt.duration] || jQuery.fx.speeds.\_default;

// Queueing

opt.old = opt.complete;

opt.complete = function(){

if ( opt.queue !== false )

jQuery(this).dequeue();

if ( jQuery.isFunction( opt.old ) )

opt.old.call( this );

};

return opt;

},

easing: {

linear: function( p, n, firstNum, diff ) {

return firstNum + diff \* p;

},

swing: function( p, n, firstNum, diff ) {

return ((-Math.cos(p\*Math.PI)/2) + 0.5) \* diff + firstNum;

}

},

timers: [],

fx: function( elem, options, prop ){

this.options = options;

this.elem = elem;

this.prop = prop;

if ( !options.orig )

options.orig = {};

}

});

jQuery.fx.prototype = {

// Simple function for setting a style value

update: function(){

if ( this.options.step )

this.options.step.call( this.elem, this.now, this );

(jQuery.fx.step[this.prop] || jQuery.fx.step.\_default)( this );

// Set display property to block for height/width animations

if ( ( this.prop == "height" || this.prop == "width" ) && this.elem.style )

this.elem.style.display = "block";

},

// Get the current size

cur: function(force){

if ( this.elem[this.prop] != null && (!this.elem.style || this.elem.style[this.prop] == null) )

return this.elem[ this.prop ];

var r = parseFloat(jQuery.css(this.elem, this.prop, force));

return r && r > -10000 ? r : parseFloat(jQuery.curCSS(this.elem, this.prop)) || 0;

},

// Start an animation from one number to another

custom: function(from, to, unit){

this.startTime = now();

this.start = from;

this.end = to;

this.unit = unit || this.unit || "px";

this.now = this.start;

this.pos = this.state = 0;

var self = this;

function t(gotoEnd){

return self.step(gotoEnd);

}

t.elem = this.elem;

if ( t() && jQuery.timers.push(t) && !timerId ) {

timerId = setInterval(function(){

var timers = jQuery.timers;

for ( var i = 0; i < timers.length; i++ )

if ( !timers[i]() )

timers.splice(i--, 1);

if ( !timers.length ) {

clearInterval( timerId );

timerId = undefined;

}

}, 13);

}

},

// Simple 'show' function

show: function(){

// Remember where we started, so that we can go back to it later

this.options.orig[this.prop] = jQuery.attr( this.elem.style, this.prop );

this.options.show = true;

// Begin the animation

// Make sure that we start at a small width/height to avoid any

// flash of content

this.custom(this.prop == "width" || this.prop == "height" ? 1 : 0, this.cur());

// Start by showing the element

jQuery(this.elem).show();

},

// Simple 'hide' function

hide: function(){

// Remember where we started, so that we can go back to it later

this.options.orig[this.prop] = jQuery.attr( this.elem.style, this.prop );

this.options.hide = true;

// Begin the animation

this.custom(this.cur(), 0);

},

// Each step of an animation

step: function(gotoEnd){

var t = now();

if ( gotoEnd || t >= this.options.duration + this.startTime ) {

this.now = this.end;

this.pos = this.state = 1;

this.update();

this.options.curAnim[ this.prop ] = true;

var done = true;

for ( var i in this.options.curAnim )

if ( this.options.curAnim[i] !== true )

done = false;

if ( done ) {

if ( this.options.display != null ) {

// Reset the overflow

this.elem.style.overflow = this.options.overflow;

// Reset the display

this.elem.style.display = this.options.display;

if ( jQuery.css(this.elem, "display") == "none" )

this.elem.style.display = "block";

}

// Hide the element if the "hide" operation was done

if ( this.options.hide )

jQuery(this.elem).hide();

// Reset the properties, if the item has been hidden or shown

if ( this.options.hide || this.options.show )

for ( var p in this.options.curAnim )

jQuery.attr(this.elem.style, p, this.options.orig[p]);

// Execute the complete function

this.options.complete.call( this.elem );

}

return false;

} else {

var n = t - this.startTime;

this.state = n / this.options.duration;

// Perform the easing function, defaults to swing

this.pos = jQuery.easing[this.options.easing || (jQuery.easing.swing ? "swing" : "linear")](this.state, n, 0, 1, this.options.duration);

this.now = this.start + ((this.end - this.start) \* this.pos);

// Perform the next step of the animation

this.update();

}

return true;

}

};

jQuery.extend( jQuery.fx, {

speeds:{

slow: 600,

fast: 200,

// Default speed

\_default: 400

},

step: {

opacity: function(fx){

jQuery.attr(fx.elem.style, "opacity", fx.now);

},

\_default: function(fx){

if ( fx.elem.style && fx.elem.style[ fx.prop ] != null )

fx.elem.style[ fx.prop ] = fx.now + fx.unit;

else

fx.elem[ fx.prop ] = fx.now;

}

}

});

if ( document.documentElement["getBoundingClientRect"] )

jQuery.fn.offset = function() {

if ( !this[0] ) return { top: 0, left: 0 };

if ( this[0] === this[0].ownerDocument.body ) return jQuery.offset.bodyOffset( this[0] );

var box = this[0].getBoundingClientRect(), doc = this[0].ownerDocument, body = doc.body, docElem = doc.documentElement,

clientTop = docElem.clientTop || body.clientTop || 0, clientLeft = docElem.clientLeft || body.clientLeft || 0,

top = box.top + (self.pageYOffset || jQuery.boxModel && docElem.scrollTop || body.scrollTop ) - clientTop,

left = box.left + (self.pageXOffset || jQuery.boxModel && docElem.scrollLeft || body.scrollLeft) - clientLeft;

return { top: top, left: left };

};

else

jQuery.fn.offset = function() {

if ( !this[0] ) return { top: 0, left: 0 };

if ( this[0] === this[0].ownerDocument.body ) return jQuery.offset.bodyOffset( this[0] );

jQuery.offset.initialized || jQuery.offset.initialize();

var elem = this[0], offsetParent = elem.offsetParent, prevOffsetParent = elem,

doc = elem.ownerDocument, computedStyle, docElem = doc.documentElement,

body = doc.body, defaultView = doc.defaultView,

prevComputedStyle = defaultView.getComputedStyle(elem, null),

top = elem.offsetTop, left = elem.offsetLeft;

while ( (elem = elem.parentNode) && elem !== body && elem !== docElem ) {

computedStyle = defaultView.getComputedStyle(elem, null);

top -= elem.scrollTop, left -= elem.scrollLeft;

if ( elem === offsetParent ) {

top += elem.offsetTop, left += elem.offsetLeft;

if ( jQuery.offset.doesNotAddBorder && !(jQuery.offset.doesAddBorderForTableAndCells && /^t(able|d|h)$/i.test(elem.tagName)) )

top += parseInt( computedStyle.borderTopWidth, 10) || 0,

left += parseInt( computedStyle.borderLeftWidth, 10) || 0;

prevOffsetParent = offsetParent, offsetParent = elem.offsetParent;

}

if ( jQuery.offset.subtractsBorderForOverflowNotVisible && computedStyle.overflow !== "visible" )

top += parseInt( computedStyle.borderTopWidth, 10) || 0,

left += parseInt( computedStyle.borderLeftWidth, 10) || 0;

prevComputedStyle = computedStyle;

}

if ( prevComputedStyle.position === "relative" || prevComputedStyle.position === "static" )

top += body.offsetTop,

left += body.offsetLeft;

if ( prevComputedStyle.position === "fixed" )

top += Math.max(docElem.scrollTop, body.scrollTop),

left += Math.max(docElem.scrollLeft, body.scrollLeft);

return { top: top, left: left };

};

jQuery.offset = {

initialize: function() {

if ( this.initialized ) return;

var body = document.body, container = document.createElement('div'), innerDiv, checkDiv, table, td, rules, prop, bodyMarginTop = body.style.marginTop,

html = '<div style="position:absolute;top:0;left:0;margin:0;border:5px solid #000;padding:0;width:1px;height:1px;"><div></div></div><table style="position:absolute;top:0;left:0;margin:0;border:5px solid #000;padding:0;width:1px;height:1px;" cellpadding="0" cellspacing="0"><tr><td></td></tr></table>';

rules = { position: 'absolute', top: 0, left: 0, margin: 0, border: 0, width: '1px', height: '1px', visibility: 'hidden' };

for ( prop in rules ) container.style[prop] = rules[prop];

container.innerHTML = html;

body.insertBefore(container, body.firstChild);

innerDiv = container.firstChild, checkDiv = innerDiv.firstChild, td = innerDiv.nextSibling.firstChild.firstChild;

this.doesNotAddBorder = (checkDiv.offsetTop !== 5);

this.doesAddBorderForTableAndCells = (td.offsetTop === 5);

innerDiv.style.overflow = 'hidden', innerDiv.style.position = 'relative';

this.subtractsBorderForOverflowNotVisible = (checkDiv.offsetTop === -5);

body.style.marginTop = '1px';

this.doesNotIncludeMarginInBodyOffset = (body.offsetTop === 0);

body.style.marginTop = bodyMarginTop;

body.removeChild(container);

this.initialized = true;

},

bodyOffset: function(body) {

jQuery.offset.initialized || jQuery.offset.initialize();

var top = body.offsetTop, left = body.offsetLeft;

if ( jQuery.offset.doesNotIncludeMarginInBodyOffset )

top += parseInt( jQuery.curCSS(body, 'marginTop', true), 10 ) || 0,

left += parseInt( jQuery.curCSS(body, 'marginLeft', true), 10 ) || 0;

return { top: top, left: left };

}

};

jQuery.fn.extend({

position: function() {

var left = 0, top = 0, results;

if ( this[0] ) {

// Get \*real\* offsetParent

var offsetParent = this.offsetParent(),

// Get correct offsets

offset = this.offset(),

parentOffset = /^body|html$/i.test(offsetParent[0].tagName) ? { top: 0, left: 0 } : offsetParent.offset();

// Subtract element margins

// note: when an element has margin: auto the offsetLeft and marginLeft

// are the same in Safari causing offset.left to incorrectly be 0

offset.top -= num( this, 'marginTop' );

offset.left -= num( this, 'marginLeft' );

// Add offsetParent borders

parentOffset.top += num( offsetParent, 'borderTopWidth' );

parentOffset.left += num( offsetParent, 'borderLeftWidth' );

// Subtract the two offsets

results = {

top: offset.top - parentOffset.top,

left: offset.left - parentOffset.left

};

}

return results;

},

offsetParent: function() {

var offsetParent = this[0].offsetParent || document.body;

while ( offsetParent && (!/^body|html$/i.test(offsetParent.tagName) && jQuery.css(offsetParent, 'position') == 'static') )

offsetParent = offsetParent.offsetParent;

return jQuery(offsetParent);

}

});

// Create scrollLeft and scrollTop methods

jQuery.each( ['Left', 'Top'], function(i, name) {

var method = 'scroll' + name;

jQuery.fn[ method ] = function(val) {

if (!this[0]) return null;

return val !== undefined ?

// Set the scroll offset

this.each(function() {

this == window || this == document ?

window.scrollTo(

!i ? val : jQuery(window).scrollLeft(),

i ? val : jQuery(window).scrollTop()

) :

this[ method ] = val;

}) :

// Return the scroll offset

this[0] == window || this[0] == document ?

self[ i ? 'pageYOffset' : 'pageXOffset' ] ||

jQuery.boxModel && document.documentElement[ method ] ||

document.body[ method ] :

this[0][ method ];

};

});

// Create innerHeight, innerWidth, outerHeight and outerWidth methods

jQuery.each([ "Height", "Width" ], function(i, name){

var tl = i ? "Left" : "Top", // top or left

br = i ? "Right" : "Bottom", // bottom or right

lower = name.toLowerCase();

// innerHeight and innerWidth

jQuery.fn["inner" + name] = function(){

return this[0] ?

jQuery.css( this[0], lower, false, "padding" ) :

null;

};

// outerHeight and outerWidth

jQuery.fn["outer" + name] = function(margin) {

return this[0] ?

jQuery.css( this[0], lower, false, margin ? "margin" : "border" ) :

null;

};

var type = name.toLowerCase();

jQuery.fn[ type ] = function( size ) {

// Get window width or height

return this[0] == window ?

// Everyone else use document.documentElement or document.body depending on Quirks vs Standards mode

document.compatMode == "CSS1Compat" && document.documentElement[ "client" + name ] ||

document.body[ "client" + name ] :

// Get document width or height

this[0] == document ?

// Either scroll[Width/Height] or offset[Width/Height], whichever is greater

Math.max(

document.documentElement["client" + name],

document.body["scroll" + name], document.documentElement["scroll" + name],

document.body["offset" + name], document.documentElement["offset" + name]

) :

// Get or set width or height on the element

size === undefined ?

// Get width or height on the element

(this.length ? jQuery.css( this[0], type ) : null) :

// Set the width or height on the element (default to pixels if value is unitless)

this.css( type, typeof size === "string" ? size : size + "px" );

};

});

})();