

Halaman utama

```
<?php
session_start();

include 'koneksi.php';

$_SESSION['username'];

$username=$_SESSION['username'];

$find_nama_user=mysql_query("select*from user where username='$username'");

$get_nama_user=mysql_fetch_array($find_nama_user);

if($get_nama_user['level'] !='admin'){

header('location: index.php');

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>Beasiswa</title>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

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

<link rel="stylesheet" href="css/bootstrap-responsive.min.css" />

<link rel="stylesheet" href="css/matrix-style.css" />

<link rel="stylesheet" href="css/matrix-media.css" />

<link href="font-awesome/css/font-awesome.css" rel="stylesheet" />

<link href='http://fonts.googleapis.com/css?family=Open+Sans:400,700,800' rel='stylesheet'
type='text/css'>

<!--<link rel="stylesheet" type="text/css" href="style.css" media="all" />-->
```

```
<script type="text/javascript" src="js/jquery.min.js"></script>
<script type="text/javascript" src="js/jquery-ui.min.js"></script>
<link rel="stylesheet" type="text/css" href="js/jquery-ui.min.css">
    <link rel="stylesheet" type="text/css" href="js/style.css">
```

```
<script type="text/javascript">
    $(document).ready(function(){
        //tambahkan auto complete
        $("#nis").autocomplete({
            source: "<?php echo 'cari_data_siswa.php';?>",
            maxlength: 4,
            select: function( event, ui ) {
                $('#nis').val(ui.item.nis);
            }
        });
    });
</script>
```

```
</head>
```

```
<body>
```

```
<!--Header-part-->
```

```
<div id="header">
```

```
    <h1><a href="dashboard.html">Admin</a></h1>
```

```
</div>
```

```
<!--close-Header-part-->
```

```

<!--top-Header-menu-->
<div id="user-nav" class="navbar navbar-inverse">
  <ul class="nav">
    <li class=""><a title="" href="#"><i class="icon icon-user"></i> <span
class="text">Halaman Admin</span></a></li>
    <li class=""><a title="" href="logout.php"><i class="icon icon-share-alt"></i> <span
class="text">Logout</span></a></li>
  </ul>
</div>

<!--start-top-serch--><!--
<div id="search">
  <input type="text" placeholder="Search here..."/>
  <button type="submit" class="tip-bottom" title="Search"><i class="icon-search icon-
white"></i></button>
</div>-->
<!--close-top-serch-->

<!--sidebar-menu-->

<div id="sidebar"> <a href="#" class="visible-phone"><i class="icon icon-
fullscreen"></i>Full width</a>

  <ul>
    <!--<li><a href="index.html"><i class="icon icon-home"></i>
<span>Dashboard</span></a> </li>
    <li><a href="charts.html"><i class="icon icon-signal"></i> <span>Charts &
graphs</span></a> </li>
    <li><a href="widgets.html"><i class="icon icon-inbox"></i> <span>Widgets</span></a>
</li>
    <li><a href="tables.html"><i class="icon icon-th"></i> <span>Tables</span></a></li>-->

```

```
<li ><a href="home_admin.php"><i class="icon icon-home"></i> <span>Profil
Sekolah</span></a></li>

<li ><a href="home_biodata_siswa.php"><i class="icon icon-user"></i> <span>Biodata
Siswa</span></a></li>

<li class="active"><a href="home_basiswa.php"><i class="icon icon-money"></i>
<span>Beasiswa</span></a></li>

<li ><a href="home_laporan_admin.php"><i class="icon icon-book"></i>
<span>Laporan Beasiswa</span></a></li>

<!--

<li class="submenu"> <a href="#"><i class="icon icon-list"></i> <span>Forms</span>
<span class="label label-important">3</span></a>

<ul>

<li><a href="form-common.html">Basic Form</a></li>

<li><a href="form-validation.html">Form with Validation</a></li>

<li><a href="form-wizard.html">Form with Wizard</a></li>

</ul>

</li>

-->

</ul>

</div>

<div id="content">

<div id="content-header">

<div id="breadcrumb"> <a href="#" title="Go to Home" class="tip-bottom"><i
class="icon-home"></i> Home</a>

<a href="#" class="current">Beasiswa</a> </div>

<h1>SMK NEGERI 4 BANDAR LAMPUNG</h1>

</div>

<div class="container-fluid">
```

```

<div class="row-fluid">
  <div class="span12">
    <div class="widget-box">
      <div class="widget-title"> <span class="icon"> <i class="icon-list"></i> </span>
      <h5>Beasiswa</h5>
    </div>

    <div class="widget-content">
      <?php
include 'koneksi.php';
if(isset($_POST['hitung'])){
    //echo "NIS PRESTASI<br>";

    //Nis Prestasi
    /*echo*/ $prestasi_a1=$_POST['prestasi']; /*echo "<br>";*/
    /*echo*/ $prestasi_a2=$_POST['prestasi2']; /*echo "<br>";*/
    /*echo*/ $prestasi_a3=$_POST['prestasi3']; /*echo "<br>";*/

    //echo "KIP<br>";

    //KIP

    //A1-----

    $nis_a1=$_POST['nis'];

    /*echo*/ $kartu_ip_a1=$_POST['kip']; /*echo "<br>";*/

    //A2-----

    $nis_a2=$_POST['nis2'];

    /*echo*/ $kartu_ip_a2=$_POST['kip2']; /*echo "<br>";*/

```

```

//A3-----
$nis_a3=$_POST['nis3'];
/*echo*/ $kartu_ip_a3=$_POST['kip3']; /*echo "<br>";*/

//echo "<br>Disiplin<br>";

//Kasus-Kasus
/*echo*/ $kasus_a1=$_POST['kasus']; /*echo "<br>";*/
/*echo*/ $absensi_a1=(($_POST['absensi']/100)); /*echo "<br>";*/
/*echo*/ $disiplin_a1=($kasus_a1+$absensi_a1)/2; /*echo "<br>";*/

/*echo*/ $kasus_a2=$_POST['kasus2']; /*echo "<br>";*/
/*echo*/ $absensi_a2=(($_POST['absensi2']/100)); /*echo "<br>";*/
/*echo*/ $disiplin_a2=($kasus_a2+$absensi_a2)/2; /*echo "<br>";*/

/*echo*/ $kasus_a3=$_POST['kasus3']; /*echo "<br>";*/
/*echo*/ $absensi_a3=(($_POST['absensi3']/100)); /*echo "<br>";*/
/*echo*/ $disiplin_a3=($kasus_a3+$absensi_a3)/2; /*echo "<br>";*/

//echo "Raport";

//Raport-Raport
/*echo "<br>";*/

//Raport A1
/*echo*/ $raport=$_POST['raport'];
/*echo*/ $raport2=$_POST['raport2']; /*echo "<br>";*/
/*echo*/ $raport_a1=((($raport)+($raport2))/2);
/*echo "<br>";*/

//Raport A2

```

```

/*echo*/ $raport21=$_POST['raport21'];

/*echo*/ $raport22=$_POST['raport22']; /*echo "<br>";*/

/*echo*/ $raport_a2=(( $raport2)+( $raport22))/2;

/*echo "<br>";*/

//Raport A3

/*echo*/ $raport3=$_POST['raport3'];

/*echo*/ $raport23=$_POST['raport23']; /*echo "<br>";*/

/*echo*/ $raport_a3=(( $raport3)+( $raport23))/2;

/*echo "<br>";*/

//echo "Prestasi Akademik/Non-Akademik<br>";

// echo " ";

//Kriteria Prestasi

/*echo*/ $prs_a1_row_1_1=($prestasi_a1/$prestasi_a1);// echo " ";

/*echo*/ $prs_a1_row_1_2=($prestasi_a2/$prestasi_a1);// echo " ";

/*echo*/ $prs_a1_row_1_3=($prestasi_a3/$prestasi_a1);

// $prs_a1_row_1_4=($prestasi_a4/$prestasi_a1);

//echo "<br>";

// echo " ";

/*echo*/ $prs_a2_row_2_1=($prestasi_a1/$prestasi_a2);// echo " ";

/*echo*/ $prs_a2_row_2_2=($prestasi_a2/$prestasi_a2);// echo " ";

/*echo*/ $prs_a2_row_2_3=($prestasi_a3/$prestasi_a2);

// $prs_a2_row_2_4=($prestasi_a4/$prestasi_a2);

//echo "<br>";

// echo " ";

/*echo*/ $prs_a3_row_3_1=($prestasi_a1/$prestasi_a3);// echo " ";

/*echo*/ $prs_a3_row_3_2=($prestasi_a2/$prestasi_a3);// echo " ";

```

```

/*echo*/ $prs_a3_row_3_3=($prestasi_a3/$prestasi_a3);
// $prs_a3_row_3_4=($prestasi_a4/$prestasi_a3);

//Kriteria Prestasi Pembagian dengan total Kolom

//echo "<br><br>";

// echo " ";

/*echo*/
$tot_prs_col_1=($prs_a1_row_1_1)+($prs_a2_row_2_1)+($prs_a3_row_3_1);// echo " ";

/*echo*/
$tot_prs_col_2=($prs_a1_row_1_2)+($prs_a2_row_2_2)+($prs_a3_row_3_2);// echo " ";

/*echo*/
$tot_prs_col_3=($prs_a1_row_1_3)+($prs_a2_row_2_3)+($prs_a3_row_3_3);

// $tot_prs_col_4=($prs_a1_row_1_4)+($prs_a2_row_2_4)+($prs_a3_row_3_4);

//Kolom 1

//echo "<br><br>";

// echo " ";

/*echo*/ $t2_prs_a1_row_1_1=($prs_a1_row_1_1)/($tot_prs_col_1);// echo " ";

/*echo*/ $t2_prs_a2_row_2_1=($prs_a2_row_2_1)/($tot_prs_col_1);// echo " ";

/*echo*/ $t2_prs_a3_row_3_1=($prs_a3_row_3_1)/($tot_prs_col_1);

//Kolom 2

//echo "<br>";

// echo " ";

/*echo*/ $t2_prs_a1_row_1_2=($prs_a1_row_1_2)/($tot_prs_col_2);// echo " ";

/*echo*/ $t2_prs_a2_row_2_2=($prs_a2_row_2_2)/($tot_prs_col_2);// echo " ";

/*echo*/ $t2_prs_a3_row_3_2=($prs_a3_row_3_2)/($tot_prs_col_2);

//Kolom 3

//echo "<br>";

```



```

// echo " ";

/*echo*/ $t2_prs_a1_row_1_3=($prs_a1_row_1_3)/($tot_prs_col_3);// echo " ";

/*echo*/ $t2_prs_a2_row_2_3=($prs_a2_row_2_3)/($tot_prs_col_3);// echo " ";

/*echo*/ $t2_prs_a3_row_3_3=($prs_a3_row_3_3)/($tot_prs_col_3);

//Kolom 4

/*$t2_prs_a1_row_1_4=($prs_a1_row_1_4)/($tot_prs_col_4);

$t2_prs_a2_row_2_4=($prs_a2_row_2_4)/($tot_prs_col_4);

$t2_prs_a3_row_3_4=($prs_a3_row_3_4)/($tot_prs_col_4);

*/

/*echo "<br>";*/

/*echo*/
$w1=((($t2_prs_a1_row_1_1)+($t2_prs_a1_row_1_2)+($t2_prs_a1_row_1_3))/3; //echo
"<br>";

/*echo*/
$w2=((($t2_prs_a2_row_2_1)+($t2_prs_a2_row_2_2)+($t2_prs_a2_row_2_3))/3; //echo
"<br>";

/*echo*/
$w3=((($t2_prs_a3_row_3_1)+($t2_prs_a3_row_3_2)+($t2_prs_a3_row_3_3))/3; //echo
"<br>";

/*echo "<table border='1'>";

echo "<tr>";

    echo "<td></td>";

    echo "<td>A1</td>";

    echo "<td>A2</td>";

    echo "<td>A3</td>";

echo "</tr>";

echo "<tr>";

    echo "<td>A1</td>";

    echo "<td>$prs_a1_row_1_1</td>";

```

```

    echo "<td>$prs_a1_row_1_2</td>";
    echo "<td>$prs_a1_row_1_3</td>";
echo "</tr>";
echo "<tr>";
    echo "<td>A2</td>";
    echo "<td>$prs_a2_row_2_1</td>";
    echo "<td>$prs_a2_row_2_2</td>";
    echo "<td>$prs_a2_row_2_3</td>";
echo "</tr>";
echo "<tr>";
    echo "<td>A3</td>";
    echo "<td>$prs_a3_row_3_1</td>";
    echo "<td>$prs_a3_row_3_2</td>";
    echo "<td>$prs_a3_row_3_3</td>";
echo "</tr>";
echo "</table>";
echo "<table border='1'>";
echo "<tr>";
    echo "<td></td>";
    echo "<td>A1</td>";
    echo "<td>A2</td>";
    echo "<td>A3</td>";
echo "</tr>";
echo "<tr>";
    echo "<td>A1</td>";
    echo "<td>$t2_prs_a1_row_1_1</td>";
    echo "<td>$t2_prs_a1_row_1_2</td>";

```

```

        echo "<td>$t2_prs_a1_row_1_3</td>";
echo "</tr>";
echo "<tr>";

    echo "<td>A2</td>";

    echo "<td>$t2_prs_a2_row_2_1</td>";
    echo "<td>$t2_prs_a2_row_2_2</td>";
    echo "<td>$t2_prs_a2_row_2_3</td>";

echo "</tr>";
echo "<tr>";

    echo "<td>A3</td>";

    echo "<td>$t2_prs_a3_row_3_1</td>";
    echo "<td>$t2_prs_a3_row_3_2</td>";
    echo "<td>$t2_prs_a3_row_3_3</td>";

echo "</tr>";

echo "</table>";

*/

/*echo*/ $w1."<br>";

/*echo*/ $w2."<br>";

/*echo*/ $w3."<br>";

//-----
//echo "//-----<br>";

//echo "Nilai raport<br>";

//Kriteria Prestasi

/*echo*/ $rpt_a1_row_1_1=($raport_a1/$raport_a1);// echo " ";

/*echo*/ $rpt_a1_row_1_2=($raport_a2/$raport_a1);// echo " ";

```

```

/*echo*/ $rpt_a1_row_1_3=($raport_a3/$raport_a1);
//$rpt_a1_row_1_4=($raport_a4/$raport_a1);
//echo "<br>";
// echo " ";
/*echo*/ $rpt_a2_row_2_1=($raport_a1/$raport_a2);// echo " ";
/*echo*/ $rpt_a2_row_2_2=($raport_a2/$raport_a2);// echo " ";
/*echo*/ $rpt_a2_row_2_3=($raport_a3/$raport_a2);
//$rpt_a2_row_2_4=($raport_a4/$raport_a2);
//echo "<br>";
// echo " ";
/*echo*/ $rpt_a3_row_3_1=($raport_a1/$raport_a3);// echo " ";
/*echo*/ $rpt_a3_row_3_2=($raport_a2/$raport_a3);// echo " ";
/*echo*/ $rpt_a3_row_3_3=($raport_a3/$raport_a3);
//$rpt_a3_row_3_4=($raport_a4/$raport_a3);

//Kriteria raport Pembagian dengan total Kolom
//echo "<br><br>";
// echo " ";
/*echo*/
$tot_rpt_col_1=($rpt_a1_row_1_1)+($rpt_a2_row_2_1)+($rpt_a3_row_3_1);// echo " ";
/*echo*/
$tot_rpt_col_2=($rpt_a1_row_1_2)+($rpt_a2_row_2_2)+($rpt_a3_row_3_2);// echo " ";
/*echo*/
$tot_rpt_col_3=($rpt_a1_row_1_3)+($rpt_a2_row_2_3)+($rpt_a3_row_3_3);
//$tot_rpt_col_4=($rpt_a1_row_1_4)+($rpt_a2_row_2_4)+($rpt_a3_row_3_4);

//Kolom 1
//echo "<br><br>";

```

```

// echo " ";

/*echo*/ $t2_rpt_a1_row_1_1=($rpt_a1_row_1_1)/($tot_rpt_col_1);// echo " ";

/*echo*/ $t2_rpt_a2_row_2_1=($rpt_a2_row_2_1)/($tot_rpt_col_1);// echo " ";

/*echo*/ $t2_rpt_a3_row_3_1=($rpt_a3_row_3_1)/($tot_rpt_col_1);

//Kolom 2

//echo "<br>";

// echo " ";

/*echo*/ $t2_rpt_a1_row_1_2=($rpt_a1_row_1_2)/($tot_rpt_col_2);// echo " ";

/*echo*/ $t2_rpt_a2_row_2_2=($rpt_a2_row_2_2)/($tot_rpt_col_2);// echo " ";

/*echo*/ $t2_rpt_a3_row_3_2=($rpt_a3_row_3_2)/($tot_rpt_col_2);

//Kolom 3

//echo "<br>";

// echo " ";

/*echo*/ $t2_rpt_a1_row_1_3=($rpt_a1_row_1_3)/($tot_rpt_col_3);// echo " ";

/*echo*/ $t2_rpt_a2_row_2_3=($rpt_a2_row_2_3)/($tot_rpt_col_3);// echo " ";

/*echo*/ $t2_rpt_a3_row_3_3=($rpt_a3_row_3_3)/($tot_rpt_col_3);

//Kolom 4

/*$t2_rpt_a1_row_1_4=($rpt_a1_row_1_4)/($tot_rpt_col_4);

$t2_rpt_a2_row_2_4=($rpt_a2_row_2_4)/($tot_rpt_col_4);

$t2_rpt_a3_row_3_4=($rpt_a3_row_3_4)/($tot_rpt_col_4);

*/

/*echo*/
$w21=(( $t2_rpt_a1_row_1_1)+($t2_rpt_a1_row_1_2)+($t2_rpt_a1_row_1_3))/3; //echo
"<br>";

/*echo*/
$w22=(( $t2_rpt_a2_row_2_1)+($t2_rpt_a2_row_2_2)+($t2_rpt_a2_row_2_3))/3; //echo
"<br>";

```

```

/*echo*/
$w23=((($t2_rpt_a3_row_3_1)+($t2_rpt_a3_row_3_2)+($t2_rpt_a3_row_3_3))/3; //echo
"<br>";

/*echo "<table border='1'>";

echo "<tr>";

    echo "<td></td>";

    echo "<td>A1</td>";

    echo "<td>A2</td>";

    echo "<td>A3</td>";

echo "</tr>";

echo "<tr>";

    echo "<td>A1</td>";

    echo "<td>$rpt_a1_row_1_1</td>";

    echo "<td>$rpt_a1_row_1_2</td>";

    echo "<td>$rpt_a1_row_1_3</td>";

echo "</tr>";

echo "<tr>";

    echo "<td>A2</td>";

    echo "<td>$rpt_a2_row_2_1</td>";

    echo "<td>$rpt_a2_row_2_2</td>";

    echo "<td>$rpt_a2_row_2_3</td>";

echo "</tr>";

echo "<tr>";

    echo "<td>A3</td>";

    echo "<td>$rpt_a3_row_3_1</td>";

    echo "<td>$rpt_a3_row_3_2</td>";

    echo "<td>$rpt_a3_row_3_3</td>";

echo "</tr>";

```

```
echo "</table>";
echo "<table border='1'>";
echo "<tr>";
    echo "<td></td>";
    echo "<td>A1</td>";
    echo "<td>A2</td>";
    echo "<td>A3</td>";
echo "</tr>";
echo "<tr>";
    echo "<td>A1</td>";
    echo "<td>${t2_rpt_a1_row_1_1}</td>";
    echo "<td>${t2_rpt_a1_row_1_2}</td>";
    echo "<td>${t2_rpt_a1_row_1_3}</td>";
echo "</tr>";
echo "<tr>";
    echo "<td>A2</td>";
    echo "<td>${t2_rpt_a2_row_2_1}</td>";
    echo "<td>${t2_rpt_a2_row_2_2}</td>";
    echo "<td>${t2_rpt_a2_row_2_3}</td>";
echo "</tr>";
echo "<tr>";
    echo "<td>A3</td>";
    echo "<td>${t2_rpt_a3_row_3_1}</td>";
    echo "<td>${t2_rpt_a3_row_3_2}</td>";
    echo "<td>${t2_rpt_a3_row_3_3}</td>";
echo "</tr>";
echo "</table>";
```

```

*/

/*echo*/ $w21."<br>";

/*echo*/ $w22."<br>";

/*echo*/ $w23."<br>";

//-----

//echo "//-----<br>";

//echo "Kedisiplinan<br>";

//Kriteria Prestasi

/*echo*/ $dis_a1_row_1_1=($disiplin_a1/$disiplin_a1);// echo " ";

/*echo*/ $dis_a1_row_1_2=($disiplin_a2/$disiplin_a1);// echo " ";

/*echo*/ $dis_a1_row_1_3=($disiplin_a3/$disiplin_a1);

//$dis_a1_row_1_4=($disiplin_a4/$disiplin_a1);

//echo "<br>";

// echo " ";

/*echo*/ $dis_a2_row_2_1=($disiplin_a1/$disiplin_a2);// echo " ";

/*echo*/ $dis_a2_row_2_2=($disiplin_a2/$disiplin_a2);// echo " ";

/*echo*/ $dis_a2_row_2_3=($disiplin_a3/$disiplin_a2);

//$dis_a2_row_2_4=($disiplin_a4/$disiplin_a2);

//echo "<br>";

// echo " ";

/*echo*/ $dis_a3_row_3_1=($disiplin_a1/$disiplin_a3);// echo " ";

/*echo*/ $dis_a3_row_3_2=($disiplin_a2/$disiplin_a3);// echo " ";

/*echo*/ $dis_a3_row_3_3=($disiplin_a3/$disiplin_a3);

//$rpt_a3_row_3_4=($raport_a4/$raport_a3);

```



```

//Kriteria raport Pembagian dengan total Kolom

//echo "<br><br>";

// echo " ";

/*echo*/
$tot_dis_col_1=($dis_a1_row_1_1)+($dis_a2_row_2_1)+($dis_a3_row_3_1);// echo " ";

/*echo*/
$tot_dis_col_2=($dis_a1_row_1_2)+($dis_a2_row_2_2)+($dis_a3_row_3_2);// echo " ";

/*echo*/
$tot_dis_col_3=($dis_a1_row_1_3)+($dis_a2_row_2_3)+($dis_a3_row_3_3);

//$tot_dis_col_4=($dis_a1_row_1_4)+($dis_a2_row_2_4)+($dis_a3_row_3_4);

//Kolom 1

//echo "<br><br>";

// echo " ";

/*echo*/ $t2_dis_a1_row_1_1=($dis_a1_row_1_1)/($tot_dis_col_1);// echo " ";

/*echo*/ $t2_dis_a2_row_2_1=($dis_a2_row_2_1)/($tot_dis_col_1);// echo " ";

/*echo*/ $t2_dis_a3_row_3_1=($dis_a3_row_3_1)/($tot_dis_col_1);

//Kolom 2

//echo "<br>";

// echo " ";

/*echo*/ $t2_dis_a1_row_1_2=($dis_a1_row_1_2)/($tot_dis_col_2);// echo " ";

/*echo*/ $t2_dis_a2_row_2_2=($dis_a2_row_2_2)/($tot_dis_col_2);// echo " ";

/*echo*/ $t2_dis_a3_row_3_2=($dis_a3_row_3_2)/($tot_dis_col_2);

//Kolom 3

//echo "<br>";

// echo " ";

/*echo*/ $t2_dis_a1_row_1_3=($dis_a1_row_1_3)/($tot_dis_col_3);// echo " ";

/*echo*/ $t2_dis_a2_row_2_3=($dis_a2_row_2_3)/($tot_dis_col_3);// echo " ";

```

```

/*echo*/ $t2_dis_a3_row_3_3=($dis_a3_row_3_3)/($tot_dis_col_3);

//Kolom 4

/*$t2_dis_a1_row_1_4=($dis_a1_row_1_4)/($tot_dis_col_4);

$t2_dis_a2_row_2_4=($dis_a2_row_2_4)/($tot_dis_col_4);

$t2_dis_a3_row_3_4=($dis_a3_row_3_4)/($tot_dis_col_4);

*/

/*echo*/
$w31=(( $t2_dis_a1_row_1_1)+($t2_dis_a1_row_1_2)+($t2_dis_a1_row_1_3))/3; //echo
"<br>";

/*echo*/
$w32=(( $t2_dis_a2_row_2_1)+($t2_dis_a2_row_2_2)+($t2_dis_a2_row_2_3))/3; //echo
"<br>";

/*echo*/
$w33=(( $t2_dis_a3_row_3_1)+($t2_dis_a3_row_3_2)+($t2_dis_a3_row_3_3))/3; //echo
"<br>";

/*echo "<table border='1'>";

echo "<tr>";

    echo "<td></td>";

    echo "<td>A1</td>";

    echo "<td>A2</td>";

    echo "<td>A3</td>";

echo "</tr>";

echo "<tr>";

    echo "<td>A1</td>";

    echo "<td>$dis_a1_row_1_1</td>";

    echo "<td>$dis_a1_row_1_2</td>";

    echo "<td>$dis_a1_row_1_3</td>";

echo "</tr>";

echo "<tr>";

```

```
echo "<td>A2</td>";
echo "<td>$dis_a2_row_2_1</td>";
echo "<td>$dis_a2_row_2_2</td>";
echo "<td>$dis_a2_row_2_3</td>";
echo "</tr>";
echo "<tr>";
echo "<td>A3</td>";
echo "<td>$dis_a3_row_3_1</td>";
echo "<td>$dis_a3_row_3_2</td>";
echo "<td>$dis_a3_row_3_3</td>";
echo "</tr>";
echo "</table>";
echo "<table border='1'>";
echo "<tr>";
echo "<td></td>";
echo "<td>A1</td>";
echo "<td>A2</td>";
echo "<td>A3</td>";
echo "</tr>";
echo "<tr>";
echo "<td>A1</td>";
echo "<td>$t2_dis_a1_row_1_1</td>";
echo "<td>$t2_dis_a1_row_1_2</td>";
echo "<td>$t2_dis_a1_row_1_3</td>";
echo "</tr>";
echo "<tr>";
echo "<td>A2</td>";
```

```

        echo "<td>$t2_dis_a2_row_2_1</td>";
        echo "<td>$t2_dis_a2_row_2_2</td>";
        echo "<td>$t2_dis_a2_row_2_3</td>";
    echo "</tr>";
    echo "<tr>";
        echo "<td>A3</td>";
        echo "<td>$t2_dis_a3_row_3_1</td>";
        echo "<td>$t2_dis_a3_row_3_2</td>";
        echo "<td>$t2_dis_a3_row_3_3</td>";
    echo "</tr>";
echo "</table>";

*/

/*echo*/ $w31."<br>";
/*echo*/ $w32."<br>";
/*echo*/ $w33."<br>";

//-----
//echo "//-----<br>";
//echo "Kartu Indonesia Pintar<br>";

//Kriteria Prestasi
/*echo*/ $kip_a1_row_1_1=($kartu_ip_a1/$kartu_ip_a1);// echo " ";
/*echo*/ $kip_a1_row_1_2=($kartu_ip_a2/$kartu_ip_a1);// echo " ";
/*echo*/ $kip_a1_row_1_3=($kartu_ip_a3/$kartu_ip_a1);
//$kip_a1_row_1_4=($kartu_ip_a4/$kartu_ip_a1);
//echo "<br>";
// echo " ";

```

```

/*echo*/ $kip_a2_row_2_1=($kartu_ip_a1/$kartu_ip_a2);// echo " ";
/*echo*/ $kip_a2_row_2_2=($kartu_ip_a2/$kartu_ip_a2);// echo " ";
/*echo*/ $kip_a2_row_2_3=($kartu_ip_a3/$kartu_ip_a2);
//$kip_a2_row_2_4=($kartu_ip_a4/$kartu_ip_a2);
//echo "<br>";
// echo " ";
/*echo*/ $kip_a3_row_3_1=($kartu_ip_a1/$kartu_ip_a3);// echo " ";
/*echo*/ $kip_a3_row_3_2=($kartu_ip_a2/$kartu_ip_a3);// echo " ";
/*echo*/ $kip_a3_row_3_3=($kartu_ip_a3/$kartu_ip_a3);
//$rpt_a3_row_3_4=($raport_a4/$raport_a3);

//Kriteria raport Pembagian dengan total Kolom
//echo "<br><br>";
// echo " ";
/*echo*/
$tot_kip_col_1=($kip_a1_row_1_1)+($kip_a2_row_2_1)+($kip_a3_row_3_1);// echo " ";
/*echo*/
$tot_kip_col_2=($kip_a1_row_1_2)+($kip_a2_row_2_2)+($kip_a3_row_3_2);// echo " ";
/*echo*/
$tot_kip_col_3=($kip_a1_row_1_3)+($kip_a2_row_2_3)+($kip_a3_row_3_3);
//$tot_kip_col_4=($kip_a1_row_1_4)+($kip_a2_row_2_4)+($kip_a3_row_3_4);

//Kolom 1
//echo "<br><br>";
// echo " ";
/*echo*/ $t2_kip_a1_row_1_1=($kip_a1_row_1_1)/($tot_kip_col_1);// echo " ";
/*echo*/ $t2_kip_a2_row_2_1=($kip_a2_row_2_1)/($tot_kip_col_1);// echo " ";
/*echo*/ $t2_kip_a3_row_3_1=($kip_a3_row_3_1)/($tot_kip_col_1);

```

```

//Kolom 2

//echo "<br>";

// echo " ";

/*echo*/ $t2_kip_a1_row_1_2=($kip_a1_row_1_2)/($tot_kip_col_2);// echo " ";

/*echo*/ $t2_kip_a2_row_2_2=($kip_a2_row_2_2)/($tot_kip_col_2);// echo " ";

/*echo*/ $t2_kip_a3_row_3_2=($kip_a3_row_3_2)/($tot_kip_col_2);

//Kolom 3

//echo "<br>";

// echo " ";

/*echo*/ $t2_kip_a1_row_1_3=($kip_a1_row_1_3)/($tot_kip_col_3);// echo " ";

/*echo*/ $t2_kip_a2_row_2_3=($kip_a2_row_2_3)/($tot_kip_col_3);// echo " ";

/*echo*/ $t2_kip_a3_row_3_3=($kip_a3_row_3_3)/($tot_kip_col_3);

//Kolom 4

/*$t2_kip_a1_row_1_4=($kip_a1_row_1_4)/($tot_kip_col_4);

$t2_kip_a2_row_2_4=($kip_a2_row_2_4)/($tot_kip_col_4);

$t2_kip_a3_row_3_4=($kip_a3_row_3_4)/($tot_kip_col_4);

*/

/*echo*/
$w41=(( $t2_kip_a1_row_1_1)+($t2_kip_a1_row_1_2)+($t2_kip_a1_row_1_3))/3; //echo
"<br>";

/*echo*/
$w42=(( $t2_kip_a2_row_2_1)+($t2_kip_a2_row_2_2)+($t2_kip_a2_row_2_3))/3; //echo
"<br>";

/*echo*/
$w43=(( $t2_kip_a3_row_3_1)+($t2_kip_a3_row_3_2)+($t2_kip_a3_row_3_3))/3; //echo
"<br>";

/*echo "<table border='1'>";

echo "<tr>";

echo "<td></td>";

```

```
    echo "<td>A1</td>";
    echo "<td>A2</td>";
    echo "<td>A3</td>";
echo "</tr>";
echo "<tr>";
    echo "<td>A1</td>";
    echo "<td>$skip_a1_row_1_1</td>";
    echo "<td>$skip_a1_row_1_2</td>";
    echo "<td>$skip_a1_row_1_3</td>";
echo "</tr>";
echo "<tr>";
    echo "<td>A2</td>";
    echo "<td>$skip_a2_row_2_1</td>";
    echo "<td>$skip_a2_row_2_2</td>";
    echo "<td>$skip_a2_row_2_3</td>";
echo "</tr>";
echo "<tr>";
    echo "<td>A3</td>";
    echo "<td>$skip_a3_row_3_1</td>";
    echo "<td>$skip_a3_row_3_2</td>";
    echo "<td>$skip_a3_row_3_3</td>";
echo "</tr>";
echo "</table>";
echo "<table border='1'>";
echo "<tr>";
    echo "<td></td>";
    echo "<td>A1</td>";
```

```

        echo "<td>A2</td>";
        echo "<td>A3</td>";
    echo "</tr>";
    echo "<tr>";
        echo "<td>A1</td>";
        echo "<td>$t2_kip_a1_row_1_1</td>";
        echo "<td>$t2_kip_a1_row_1_2</td>";
        echo "<td>$t2_kip_a1_row_1_3</td>";
    echo "</tr>";
    echo "<tr>";
        echo "<td>A2</td>";
        echo "<td>$t2_kip_a2_row_2_1</td>";
        echo "<td>$t2_kip_a2_row_2_2</td>";
        echo "<td>$t2_kip_a2_row_2_3</td>";
    echo "</tr>";
    echo "<tr>";
        echo "<td>A3</td>";
        echo "<td>$t2_kip_a3_row_3_1</td>";
        echo "<td>$t2_kip_a3_row_3_2</td>";
        echo "<td>$t2_kip_a3_row_3_3</td>";
    echo "</tr>";
echo "</table>";

*/

/*echo*/ $w41."<br>";
/*echo*/ $w42."<br>";
/*echo*/ $w43."<br>";

```



```
$wall_1=0.566;
```

```
$wall_2=0.155;
```

```
$wall_3=0.118;
```

```
$wall_4=0.161;
```

```
/*echo*/ $nis_a1."
```

```
".$fin_a1=($w1*$wall_1)+($w21*$wall_1)+($w31*$wall_1)+($w41*$wall_1); //echo  
"<br>";
```

```
/*echo*/ $nis_a2."
```

```
".$fin_a2=($w2*$wall_2)+($w22*$wall_2)+($w32*$wall_2)+($w42*$wall_2); //echo  
"<br>";
```

```
/*echo*/ $nis_a3."
```

```
".$fin_a3=($w3*$wall_3)+($w23*$wall_3)+($w33*$wall_3)+($w43*$wall_3); //echo  
"<br>";
```

```
/*
```

```
echo $nis_a1."
```

```
".$fin_a1=($w1*$wall_1)+($w21*$wall_2)+($w31*$wall_3)+($w41*$wall_4); echo  
"<br>";
```

```
echo $nis_a2."
```

```
".$fin_a2=($w2*$wall_1)+($w22*$wall_2)+($w32*$wall_3)+($w42*$wall_4); echo  
"<br>";
```

```
echo $nis_a3."
```

```
".$fin_a3=($w3*$wall_1)+($w23*$wall_2)+($w33*$wall_3)+($w43*$wall_4); echo  
"<br>";
```

```
*/
```

```
$tahun=date('Y');
```

```
$sket="Mendapatkan Beasiswa";
```

```
$sket2="Tidak Mendapatkan Beasiswa";
```

```
//Insert ke tabel lap_basiswa_detail
```

```
mysql_query("insert into lap_basiswa_detail(nis,raport,prestasi,disiplin,kip,total)
```

```

values('$nis_a1','$raport_a1','$prestasi_a1','$disiplin_a1','$kartu_ip_a1','$fin_a1');

mysql_query("insert into lap_basiswa_detail(nis,raport,prestasi,disiplin,kip,total)

values('$nis_a2','$raport_a2','$prestasi_a2','$disiplin_a2','$kartu_ip_a2','$fin_a2');

mysql_query("insert into lap_basiswa_detail(nis,raport,prestasi,disiplin,kip,total)

values('$nis_a3','$raport_a3','$prestasi_a3','$disiplin_a3','$kartu_ip_a3','$fin_a3');

//Perubahan status pengajuan beasiswa dari belum jadi sudah

mysql_query("update pengajuan set status='sudah' where pengajuan.nis='$nis_a1'");
mysql_query("update pengajuan set status='sudah' where pengajuan.nis='$nis_a2'");
mysql_query("update pengajuan set status='sudah' where pengajuan.nis='$nis_a3'");

if ($fin_a1>$fin_a2 && $fin_a1>$fin_a3){

echo "<b>Yang berhak mendapatkan beasiswa : </b><br>";

echo "NIS : ".$nis_a1; echo "<br>";

$go_data=mysql_query("select
tb_bio_siswa.nama,tb_bio_siswa.jurusan,tb_bio_siswa.kelas

from tb_bio_siswa where tb_bio_siswa.nis='$nis_a1'");

$data=mysql_fetch_array($go_data);

echo "Nama : ".$data[0]; echo "<br>";

echo "Jurusan : ".$data[1]; echo "<br>";

echo "Kelas : ".$data[2]; echo "<br>";

echo "<a href='cetak_pdf.php?cetak_bea=$nis_a1' title='Cetak data $data[0]'><img
src='img/pdf.png'> Cetak PDF</a>";

mysql_query("insert into
lap_basiswa(tahun,nis,ket)values('$tahun','$nis_a1','$ket')");

mysql_query("insert into
lap_basiswa(tahun,nis,ket)values('$tahun','$nis_a2','$ket2')");

```

```

        mysql_query("insert into
lap_bebasiswa(tahun,nis,ket)values('$tahun','$nis_a3','$ket2')");

    }

    else if($fin_a2>$fin_a1 && $fin_a2>$fin_a3){

        echo "<b>Yang berhak mendapatkan beasiswa : </b><br>";

        echo "NIS : ".$nis_a1; echo "<br>";

        $go_data=mysql_query("select
tb_bio_siswa.nama,tb_bio_siswa.jurusan,tb_bio_siswa.kelas

                from tb_bio_siswa where tb_bio_siswa.nis='$nis_a2'");

        $data=mysql_fetch_array($go_data);

        echo "Nama : ".$data[0]; echo "<br>";

        echo "Jurusan : ".$data[1]; echo "<br>";

        echo "Kelas : ".$data[2]; echo "<br>";

        echo "<a href='cetak_pdf.php?cetak_bea=$nis_a2'><img src='img/pdf.png'> Cetak
PDF</a>";

        mysql_query("insert into
lap_bebasiswa(tahun,nis,ket)values('$tahun','$nis_a2','$ket')");

        mysql_query("insert into
lap_bebasiswa(tahun,nis,ket)values('$tahun','$nis_a1','$ket2')");

        mysql_query("insert into
lap_bebasiswa(tahun,nis,ket)values('$tahun','$nis_a3','$ket2')");

    }

    else if($fin_a3>$fin_a1 && $fin_a3>$fin_a2){

        echo "<b>Yang berhak mendapatkan beasiswa : </b><br>";

        echo "NIS : ".$nis_a1; echo "<br>";

        $go_data=mysql_query("select
tb_bio_siswa.nama,tb_bio_siswa.jurusan,tb_bio_siswa.kelas

                from tb_bio_siswa where tb_bio_siswa.nis='$nis_a3'");

```

```

$data=mysql_fetch_array($go_data);
echo "Nama : ".$data[0]; echo "<br>";
echo "Jurusan : ".$data[1]; echo "<br>";
echo "Kelas : ".$data[2]; echo "<br>";

mysql_query("insert into
lap_basiswa(tahun,nis,ket)values('$tahun','$nis_a3','$ket)");

mysql_query("insert into
lap_basiswa(tahun,nis,ket)values('$tahun','$nis_a1','$ket2)");

mysql_query("insert into
lap_basiswa(tahun,nis,ket)values('$tahun','$nis_a2','$ket2)");

}

echo "<br><b><font color='black'>Detail perhitungan nilai : </font></b><br>";
echo "<table border=1 >";

echo "<tr height='45' bgcolor=#C2CCD4' align='center'>";

echo "<td></td>";

echo "<td width='45'><b>NIS</b></td>";

echo "<td width='75'><b>Raport</b></td>";

echo "<td width='105'><b>Prestasi Akademik/Non</b></td>";

echo "<td width='85'><b>Kedisiplinan</b></td>";

echo "<td width='35'><b>KIP</b></td>";

echo "<td><b>Total Perhitungan</b></td>";

echo "</tr>";

echo "<tr align='center'>";

echo "<td><a href='cetak_pdf.php?cetak_bea=$nis_a1' >
<img src='img/pdf.png'> Cetak PDF</a>
</td>";

echo "<td>$nis_a1</td>";

```

```

echo "<td>$raport_a1</td>";
echo "<td>$prestasi_a1</td>";
echo "<td>$disiplin_a1</td>";
echo "<td>$kartu_ip_a1</td>";
echo "<td>$fin_a1</td>";
echo "</tr>";
echo "<tr align='center'>";
echo "<td><a href='cetak_pdf.php?cetak_bea=$nis_a3' >
    <img src='img/pdf.png'> Cetak PDF</a>
    </td>";
echo "<td>$nis_a3</td>";
echo "<td>$raport_a3</td>";
echo "<td>$prestasi_a3</td>";
echo "<td>$disiplin_a3</td>";
echo "<td>$kartu_ip_a3</td>";
echo "<td>$fin_a2</td>";
echo "</tr>";
echo "<tr align='center'>";
echo "<td><a href='cetak_pdf.php?cetak_bea=$nis_a2' >
    <img src='img/pdf.png'> Cetak PDF</a>
    </td>";
echo "<td>$nis_a2</td>";
echo "<td>$raport_a2</td>";
echo "<td>$prestasi_a2</td>";
echo "<td>$disiplin_a2</td>";
echo "<td>$kartu_ip_a2</td>";
echo "<td>$fin_a3</td>";

```

```
        echo "</tr>";
        echo "</table>";
    }
    ?>
    </div>

    </div>

    </div>

    </div>

    </div>

    </div>

    </div>

    <!--Footer-part-->
    <div class="row-fluid">
        <div id="footer" class="span12"> 2013 &copy; Matrix Admin. Brought to you by <a
href="http://themedesigner.in">Themedesigner.in</a> </div>
    </div>

    <!--end-Footer-part-->

    <script src="js/jquery.min.js"></script>
    <script src="js/jquery.ui.custom.js"></script>
    <script src="js/bootstrap.min.js"></script>
    <script src="js/matrix.js"></script>
</body>
</html>

/**
```

\* jQuery Validation Plugin 1.9.0

\*

\* <http://bassistance.de/jquery-plugins/jquery-plugin-validation/>

\* <http://docs.jquery.com/Plugins/Validation>

\*

\* Copyright (c) 2006 - 2011 Jörn Zaefferer

\*

\* Dual licensed under the MIT and GPL licenses:

\* <http://www.opensource.org/licenses/mit-license.php>

\* <http://www.gnu.org/licenses/gpl.html>

\*/

```
(function($) {
```

```
$.extend($.fn, {
```

```
    // http://docs.jquery.com/Plugins/Validation/validate
```

```
    validate: function( options ) {
```

```
        // if nothing is selected, return nothing; can't chain anyway
```

```
        if (!this.length) {
```

```
            options && options.debug && window.console && console.warn(
                "nothing selected, can't validate, returning nothing" );
```

```
            return;
```

```
        }
```

```
        // check if a validator for this form was already created
```

```
        var validator = $.data(this[0], 'validator');
```

```

if ( validator ) {
    return validator;
}

// Add novalidate tag if HTML5.
this.attr('novalidate', 'novalidate');

validator = new $.validator( options, this[0] );
$.data(this[0], 'validator', validator);

if ( validator.settings.onSubmit ) {

    var inputsAndButtons = this.find("input, button");

    // allow suppressing validation by adding a cancel class to the submit
button
    inputsAndButtons.filter(".cancel").click(function () {
        validator.cancelSubmit = true;
    });

    // when a submitHandler is used, capture the submitting button
    if (validator.settings.submitHandler) {
        inputsAndButtons.filter(":submit").click(function () {
            validator.submitButton = this;
        });
    }
}

```



```

// validate the form on submit
this.submit( function( event ) {
    if ( validator.settings.debug )
        // prevent form submit to be able to see console output
        event.preventDefault();

    function handle() {
        if ( validator.settings.submitHandler ) {
            if (validator.submitButton) {
                // insert a hidden input as a replacement
                for the missing submit button
                var hidden = $("<input
                type='hidden'/>").attr("name",
                validator.submitButton.name).val(validator.submitButton.value).appendTo(validator.currentF
                orm);
            }
            validator.settings.submitHandler.call( validator,
            validator.currentForm );
            if (validator.submitButton) {
                // and clean up afterwards; thanks to no-
                block-scope, hidden can be referenced
                hidden.remove();
            }
            return false;
        }
        return true;
    }

    // prevent submit for invalid forms or custom submit handlers

```

```

        if ( validator.cancelSubmit ) {
            validator.cancelSubmit = false;
            return handle();
        }
        if ( validator.form() ) {
            if ( validator.pendingRequest ) {
                validator.formSubmitted = true;
                return false;
            }
            return handle();
        } else {
            validator.focusInvalid();
            return false;
        }
    });
}

return validator;
},
// http://docs.jquery.com/Plugins/Validation/valid
valid: function() {
if ( $(this[0]).is('form') ) {
    return this.validate().form();
} else {
    var valid = true;
    var validator = $(this[0].form).validate();
    this.each(function() {

```

```

        valid &= validator.element(this);

    });

    return valid;
}

},

// attributes: space separated list of attributes to retrieve and remove
removeAttrs: function(attributes) {

    var result = {},

        $element = this;

    $.each(attributes.split(/\s/), function(index, value) {

        result[value] = $element.attr(value);

        $element.removeAttr(value);

    });

    return result;

},

// http://docs.jquery.com/Plugins/Validation/rules
rules: function(command, argument) {

    var element = this[0];

    if (command) {

        var settings = $.data(element.form, 'validator').settings;

        var staticRules = settings.rules;

        var existingRules = $.validator.staticRules(element);

        switch(command) {

            case "add":

                $.extend(existingRules, $.validator.normalizeRule(argument));

                staticRules[element.name] = existingRules;

```

```

        if (argument.messages)
            settings.messages[element.name] = $.extend(
settings.messages[element.name], argument.messages );
        break;
    case "remove":
        if (!argument) {
            delete staticRules[element.name];
            return existingRules;
        }
        var filtered = {};
        $.each(argument.split(/\s/), function(index, method) {
            filtered[method] = existingRules[method];
            delete existingRules[method];
        });
        return filtered;
    }
}

```

```

var data = $.validator.normalizeRules(
$.extend(
    {},
    $.validator.metadataRules(element),
    $.validator.classRules(element),
    $.validator.attributeRules(element),
    $.validator.staticRules(element)
), element);

```

```

        // make sure required is at front
        if (data.required) {
            var param = data.required;
            delete data.required;
            data = $.extend({required: param}, data);
        }

        return data;
    }
});

// Custom selectors
$.extend($.expr[":"], {
    // http://docs.jquery.com/Plugins/Validation/blank
    blank: function(a) {return !$.trim("" + a.value);},
    // http://docs.jquery.com/Plugins/Validation/filled
    filled: function(a) {return !!$.trim("" + a.value);},
    // http://docs.jquery.com/Plugins/Validation/unchecked
    unchecked: function(a) {return !a.checked;}
});

// constructor for validator
$.validator = function( options, form ) {
    this.settings = $.extend( true, {}, $.validator.defaults, options );
    this.currentForm = form;
    this.init();
};

```

```

$.validator.format = function(source, params) {
    if ( arguments.length == 1 )
        return function() {
            var args = $.makeArray(arguments);
            args.unshift(source);
            return $.validator.format.apply( this, args );
        };
    if ( arguments.length > 2 && params.constructor != Array ) {
        params = $.makeArray(arguments).slice(1);
    }
    if ( params.constructor != Array ) {
        params = [ params ];
    }
    $.each(params, function(i, n) {
        source = source.replace(new RegExp("\\{" + i + "\\}", "g"), n);
    });
    return source;
};

```

```

$.extend($.validator, {

    defaults: {

        messages: {},

        groups: {},

        rules: {},

        errorClass: "error",

```

```

validClass: "valid",
errorElement: "label",
focusInvalid: true,
errorContainer: $( [] ),
errorLabelContainer: $( [] ),
onsubmit: true,
ignore: ":hidden",
ignoreTitle: false,
onfocusin: function(element, event) {
    this.lastActive = element;

    // hide error label and remove error class on focus if enabled
    if ( this.settings.focusCleanup && !this.blockFocusCleanup ) {
        this.settings.unhighlight && this.settings.unhighlight.call( this,
element, this.settings.errorClass, this.settings.validClass );
        this.addWrapper(this.errorsFor(element)).hide();
    }
},
onfocusout: function(element, event) {
    if ( !this.checkable(element) && (element.name in this.submitted ||
!this.optional(element)) ) {
        this.element(element);
    }
},
onkeyup: function(element, event) {
    if ( element.name in this.submitted || element == this.lastElement ) {
        this.element(element);
    }
}

```

```

    },
    onclick: function(element, event) {
        // click on selects, radiobuttons and checkboxes
        if ( element.name in this.submitted )
            this.element(element);
        // or option elements, check parent select in that case
        else if (element.parentNode.name in this.submitted)
            this.element(element.parentNode);
    },
    highlight: function(element, errorClass, validClass) {
        if (element.type === 'radio') {

this.findByName(element.name).addClass(errorClass).removeClass(validClass);

            } else {
                $(element).addClass(errorClass).removeClass(validClass);
            }
        },
        unhighlight: function(element, errorClass, validClass) {
            if (element.type === 'radio') {

this.findByName(element.name).removeClass(errorClass).addClass(validClass);

            } else {
                $(element).removeClass(errorClass).addClass(validClass);
            }
        }
    },

```

// <http://docs.jquery.com/Plugins/Validation/Validator/setDefaults>



```
setDefaults: function(settings) {  
    $.extend( $.validator.defaults, settings );  
},  
  
messages: {  
    required: "This field is required.",  
    remote: "Please fix this field.",  
    email: "Please enter a valid email address.",  
    url: "Please enter a valid URL.",  
    date: "Please enter a valid date.",  
    dateISO: "Please enter a valid date (ISO).",  
    number: "Please enter a valid number.",  
    digits: "Please enter only digits.",  
    creditcard: "Please enter a valid credit card number.",  
    equalTo: "Please enter the same value again.",  
    accept: "Please enter a value with a valid extension.",  
    maxlength: $.validator.format("Please enter no more than {0} characters."),  
    minlength: $.validator.format("Please enter at least {0} characters."),  
    rangelength: $.validator.format("Please enter a value between {0} and {1}  
characters long."),  
    range: $.validator.format("Please enter a value between {0} and {1}."),  
    max: $.validator.format("Please enter a value less than or equal to {0}."),  
    min: $.validator.format("Please enter a value greater than or equal to {0}."),  
},  
  
autoCreateRanges: false,
```

```

prototype: {

    init: function() {

        this.labelContainer = $(this.settings.errorLabelContainer);

        this.errorContext = this.labelContainer.length && this.labelContainer ||
$(this.currentForm);

        this.containers = $(this.settings.errorContainer).add(
this.settings.errorLabelContainer );

        this.submitted = {};

        this.valueCache = {};

        this.pendingRequest = 0;

        this.pending = {};

        this.invalid = {};

        this.reset();

        var groups = (this.groups = {});

        $.each(this.settings.groups, function(key, value) {

            $.each(value.split(/\s/), function(index, name) {

                groups[name] = key;

            });

        });

        var rules = this.settings.rules;

        $.each(rules, function(key, value) {

            rules[key] = $.validator.normalizeRule(value);

        });

        function delegate(event) {

            var validator = $.data(this[0].form, "validator"),

```

```

        eventType = "on" + event.type.replace(/^validate/, "");

        validator.settings[eventType] &&
validator.settings[eventType].call(validator, this[0], event);

    }

    $(this.currentForm)

        .validateDelegate("[type='text'], [type='password'], [type='file'],
select, textarea, " +

                                "[type='number'], [type='search'], [type='tel'],
[type='url'], " +

                                "[type='email'], [type='datetime'], [type='date'],
[type='month'], " +

                                "[type='week'], [type='time'], [type='datetime-
local'], " +

                                "[type='range'], [type='color'] ",

                                "focusin focusout keyup", delegate)

        .validateDelegate("[type='radio'], [type='checkbox'], select,
option", "click", delegate);

    if (this.settings.invalidHandler)

        $(this.currentForm).bind("invalid-form.validate",
this.settings.invalidHandler);

    },

// http://docs.jquery.com/Plugins/Validation/Validator/form
form: function() {

    this.checkForm();

    $.extend(this.submitted, this.errorMap);

    this.invalid = $.extend({}, this.errorMap);

    if (!this.valid())

        $(this.currentForm).triggerHandler("invalid-form", [this]);

```

```

        this.showErrors();
        return this.valid();
    },

    checkForm: function() {
        this.prepareForm();
        for ( var i = 0, elements = (this.currentElements = this.elements());
elements[i]; i++ ) {
            this.check( elements[i] );
        }
        return this.valid();
    },

// http://docs.jquery.com/Plugins/Validation/Validator/element
element: function( element ) {
    element = this.validationTargetFor( this.clean( element ) );
    this.lastElement = element;
    this.prepareElement( element );
    this.currentElements = $(element);
    var result = this.check( element );
    if ( result ) {
        delete this.invalid[element.name];
    } else {
        this.invalid[element.name] = true;
    }
    if ( !this.numberOfInvalids() ) {
        // Hide error containers on last error

```

```

        this.toHide = this.toHide.add( this.containers );
    }
    this.showErrors();
    return result;
},

// http://docs.jquery.com/Plugins/Validation/Validator/showErrors
showErrors: function(errors) {
    if(errors) {
        // add items to error list and map
        $.extend( this.errorMap, errors );
        this.errorList = [];
        for ( var name in errors ) {
            this.errorList.push({
                message: errors[name],
                element: this.findByName(name)[0]
            });
        }
        // remove items from success list
        this.successList = $.grep( this.successList, function(element) {
            return !(element.name in errors);
        });
    }
    this.settings.showErrors
        ? this.settings.showErrors.call( this, this.errorMap,
this.errorList )
        : this.defaultShowErrors();
}

```

```
},
```

```
// http://docs.jquery.com/Plugins/Validation/Validator/resetForm
```

```
resetForm: function() {
```

```
    if ( $.fn.resetForm )
```

```
        $( this.currentForm ).resetForm();
```

```
    this.submitted = {};
```

```
    this.lastElement = null;
```

```
    this.prepareForm();
```

```
    this.hideErrors();
```

```
    this.elements().removeClass( this.settings.errorClass );
```

```
},
```

```
numberOfInvalids: function() {
```

```
    return this.objectLength(this.invalid);
```

```
},
```

```
objectLength: function( obj ) {
```

```
    var count = 0;
```

```
    for ( var i in obj )
```

```
        count++;
```

```
    return count;
```

```
},
```

```
hideErrors: function() {
```

```
    this.addWrapper( this.toHide ).hide();
```

```
},
```

```

valid: function() {
    return this.size() == 0;
},

size: function() {
    return this.errorList.length;
},

focusInvalid: function() {
    if( this.settings.focusInvalid ) {
        try {
            $(this.findLastActive() || this.errorList.length &&
this.errorList[0].element || [])
                .filter(":visible")
                .focus()
                // manually trigger focusin event; without it, focusin
handler isn't called, findLastActive won't have anything to find
                .trigger("focusin");
        } catch(e) {
            // ignore IE throwing errors when focusing hidden
elements
        }
    }
},

findLastActive: function() {
    var lastActive = this.lastActive;

```

```

return lastActive && $.grep(this.errorList, function(n) {
    return n.element.name == lastActive.name;
}).length == 1 && lastActive;
},

elements: function() {
    var validator = this,
        rulesCache = {};

    // select all valid inputs inside the form (no submit or reset buttons)
    return $(this.currentForm)
        .find("input, select, textarea")
        .not(":submit, :reset, :image, [disabled]")
        .not( this.settings.ignore )
        .filter(function() {
            !this.name && validator.settings.debug && window.console
            && console.error( "%o has no name assigned", this);

            // select only the first element for each name, and only those
            with rules specified
            if ( this.name in rulesCache ||
                !validator.objectLength($(this).rules()) )
                return false;

            rulesCache[this.name] = true;

            return true;
        });
},

```



```
clean: function( selector ) {
    return $( selector )[0];
},

errors: function() {
    return $( this.settings.errorElement + "." + this.settings.errorClass,
this.errorContext );
},

reset: function() {
    this.successList = [];
    this.errorList = [];
    this.errorMap = {};
    this.toShow = $([]);
    this.toHide = $([]);
    this.currentElements = $([]);
},

prepareForm: function() {
    this.reset();
    this.toHide = this.errors().add( this.containers );
},

prepareElement: function( element ) {
    this.reset();
    this.toHide = this.errorsFor(element);
```

```
},
```

```
check: function( element ) {  
    element = this.validationTargetFor( this.clean( element ) );  
  
    var rules = $(element).rules();  
    var dependencyMismatch = false;  
    for (var method in rules ) {  
        var rule = { method: method, parameters: rules[method] };  
        try {  
            var result = $.validator.methods[method].call( this,  
element.value.replace(/r/g, ""), element, rule.parameters );  
  
            // if a method indicates that the field is optional and  
therefore valid,  
  
            // don't mark it as valid when there are no other rules  
            if ( result == "dependency-mismatch" ) {  
                dependencyMismatch = true;  
                continue;  
            }  
            dependencyMismatch = false;  
  
            if ( result == "pending" ) {  
                this.toHide = this.toHide.not(  
this.errorsFor(element) );  
  
                return;  
            }  
        }  
    }  
}
```

```

        if( !result ) {
            this.formatAndAdd( element, rule );
            return false;
        }
    } catch(e) {
        this.settings.debug && window.console &&
console.log("exception occured when checking element " + element.id
            + ", check the '" + rule.method + "' method", e);
        throw e;
    }
}
if (dependencyMismatch)
    return;
if ( this.objectLength(rules) )
    this.successList.push(element);
return true;
},

// return the custom message for the given element and validation method
// specified in the element's "messages" metadata
customMetaMessage: function(element, method) {
    if (!$.metadata)
        return;

    var meta = this.settings.meta
        ? $(element).metadata()[this.settings.meta]
        : $(element).metadata();

```

```

        return meta && meta.messages && meta.messages[method];
    },

    // return the custom message for the given element name and validation
method
    customMessage: function( name, method ) {
        var m = this.settings.messages[name];
        return m && (m.constructor === String
            ? m
            : m[method]);
    },

    // return the first defined argument, allowing empty strings
    findDefined: function() {
        for(var i = 0; i < arguments.length; i++) {
            if (arguments[i] !== undefined)
                return arguments[i];
        }
        return undefined;
    },

    defaultMessage: function( element, method ) {
        return this.findDefined(
            this.customMessage( element.name, method ),
            this.customMetaMessage( element, method ),
            // title is never undefined, so handle empty string as undefined

```

```

        !this.settings.ignoreTitle && element.title || undefined,
        $.validator.messages[method],
        "<strong>Warning: No message defined for " + element.name
+ "</strong>"
    );
},

```

```

formatAndAdd: function( element, rule ) {
    var message = this.defaultMessage( element, rule.method ),
        theregex = /^$\{(\d+)\}/g;
    if ( typeof message == "function" ) {
        message = message.call(this, rule.parameters, element);
    } else if (theregex.test(message)) {
        message = jQuery.format(message.replace(theregex, '{$1}'),
rule.parameters);
    }
    this.errorList.push({
        message: message,
        element: element
    });

    this.errorMap[element.name] = message;
    this.submitted[element.name] = message;
},

```

```

addWrapper: function(toToggle) {
    if ( this.settings.wrapper )

```

```

        toToggle = toToggle.add( toToggle.parent(
this.settings.wrapper ) );

        return toToggle;

    },

    defaultShowErrors: function() {

        for ( var i = 0; this.errorList[i]; i++ ) {

            var error = this.errorList[i];

            this.settings.highlight && this.settings.highlight.call( this,
error.element, this.settings.errorClass, this.settings.validClass );

            this.showLabel( error.element, error.message );

        }

        if( this.errorList.length ) {

            thisToShow = thisToShow.add( this.containers );

        }

        if (this.settings.success) {

            for ( var i = 0; this.successList[i]; i++ ) {

                this.showLabel( this.successList[i] );

            }

        }

        if (this.settings.unhighlight) {

            for ( var i = 0, elements = this.validElements(); elements[i]; i++
) {

                this.settings.unhighlight.call( this, elements[i],
this.settings.errorClass, this.settings.validClass );

            }

        }

        this.toHide = this.toHide.not( thisToShow );

```

```

        this.hideErrors();
        this.addWrapper( thisToShow ).show();
    },

    validElements: function() {
        return this.currentElements.not(this.invalidElements());
    },

    invalidElements: function() {
        return $(this.errorList).map(function() {
            return this.element;
        });
    },

    showLabel: function(element, message) {
        var label = this.errorsFor( element );
        if ( label.length ) {
            // refresh error/success class
            label.removeClass( this.settings.validClass ).addClass(
this.settings.errorClass );

            // check if we have a generated label, replace the message then
            label.attr("generated") && label.html(message);
        } else {
            // create label
            label = $("<" + this.settings.errorElement + ">")
                .attr({"for": this.idOrName(element), generated: true})

```

```

        .addClass(this.settings.errorClass)
        .html(message || "");
    if ( this.settings.wrapper ) {
        // make sure the element is visible, even in IE
        // actually showing the wrapped element is handled
elsewhere
        label = label.hide().show().wrap("<" +
this.settings.wrapper + ">").parent();
    }
    if ( !this.labelContainer.append(label).length )
        this.settings.errorPlacement
            ? this.settings.errorPlacement(label, $(element) )
            : label.insertAfter(element);
}
if ( !message && this.settings.success ) {
    label.text("");
    typeof this.settings.success == "string"
        ? label.addClass( this.settings.success )
        : this.settings.success( label );
}
this.toShow = this.toShow.add(label);
},

errorsFor: function(element) {
    var name = this.idOrName(element);
    return this.errors().filter(function() {
        return $(this).attr('for') == name;
    });
});

```



```

    },

    idOrName: function(element) {
        return this.groups[element.name] || (this.checkable(element) ?
element.name : element.id || element.name);
    },

    validationTargetFor: function(element) {
        // if radio/checkbox, validate first element in group instead
        if (this.checkable(element)) {
            element = this.findByName( element.name
).not(this.settings.ignore)[0];
        }
        return element;
    },

    checkable: function( element ) {
        return /radio|checkbox/i.test(element.type);
    },

    findByName: function( name ) {
        // select by name and filter by form for performance over
form.find("[name=...]")
        var form = this.currentForm;
        return $(document.getElementsByName(name)).map(function(index,
element) {
            return element.form == form && element.name == name &&
element || null;
        });
    }

```

```
},
```

```
getLength: function(value, element) {  
    switch( element.nodeName.toLowerCase() ) {  
        case 'select':  
            return $("option:selected", element).length;  
        case 'input':  
            if( this.checkable( element) )  
                return  
this.findByName(element.name).filter(':checked').length;  
    }  
    return value.length;  
},
```

```
depend: function(param, element) {  
    return this.dependTypes[typeof param]  
        ? this.dependTypes[typeof param](param, element)  
        : true;  
},
```

```
dependTypes: {  
    "boolean": function(param, element) {  
        return param;  
    },  
    "string": function(param, element) {  
        return !!$(param, element.form).length;  
    },  
},
```

```

        "function": function(param, element) {
            return param(element);
        }
    },

    optional: function(element) {
        return !$.validator.methods.required.call(this, $.trim(element.value),
element) && "dependency-mismatch";
    },

    startRequest: function(element) {
        if (!this.pending[element.name]) {
            this.pendingRequest++;
            this.pending[element.name] = true;
        }
    },

    stopRequest: function(element, valid) {
        this.pendingRequest--;

        // sometimes synchronization fails, make sure pendingRequest is never
        < 0

        if (this.pendingRequest < 0)
            this.pendingRequest = 0;
        delete this.pending[element.name];
        if ( valid && this.pendingRequest == 0 && this.formSubmitted &&
this.form() ) {
            $(this.currentForm).submit();
            this.formSubmitted = false;

```

```

        } else if (!valid && this.pendingRequest == 0 && this.formSubmitted)
    {
        $(this.currentForm).triggerHandler("invalid-form", [this]);
        this.formSubmitted = false;
    }
},

previousValue: function(element) {
    return $.data(element, "previousValue") || $.data(element,
"previousValue", {
        old: null,
        valid: true,
        message: this.defaultMessage( element, "remote" )
    });
}

},

classRuleSettings: {
    required: {required: true},
    email: {email: true},
    url: {url: true},
    date: {date: true},
    dateISO: {dateISO: true},
    dateDE: {dateDE: true},
    number: {number: true},
    numberDE: {numberDE: true},
    digits: {digits: true},

```

```

        creditcard: { creditcard: true }
    },

    addClassRules: function(className, rules) {
        className.constructor == String ?
            this.classRuleSettings[className] = rules :
            $.extend(this.classRuleSettings, className);
    },

    classRules: function(element) {
        var rules = {};
        var classes = $(element).attr('class');
        classes && $.each(classes.split(' '), function() {
            if (this in $.validator.classRuleSettings) {
                $.extend(rules, $.validator.classRuleSettings[this]);
            }
        });
        return rules;
    },

    attributeRules: function(element) {
        var rules = {};
        var $element = $(element);

        for (var method in $.validator.methods) {
            var value;
            // If .prop exists (jQuery >= 1.6), use it to get true/false for required

```

```

    if (method === 'required' && typeof $.fn.prop === 'function') {
        value = $element.prop(method);
    } else {
        value = $element.attr(method);
    }
    if (value) {
        rules[method] = value;
    } else if ($element[0].getAttribute("type") === method) {
        rules[method] = true;
    }
}

```

text inputs // maxlength may be returned as -1, 2147483647 (IE) and 524288 (safari) for

```

if (rules.maxlength && /-1|2147483647|524288/.test(rules.maxlength)) {
    delete rules.maxlength;
}

```

```

return rules;

```

```

},

```

```

metadataRules: function(element) {

```

```

    if (!$.metadata) return {};

```

```

    var meta = $.data(element.form, 'validator').settings.meta;

```

```

    return meta ?

```

```

        $(element).metadata()[meta] :

```

```

        $(element).metadata();
    },

    staticRules: function(element) {
        var rules = {};
        var validator = $.data(element.form, 'validator');
        if (validator.settings.rules) {
            rules =
$.validator.normalizeRule(validator.settings.rules[element.name]) || {};
        }
        return rules;
    },

    normalizeRules: function(rules, element) {
        // handle dependency check
        $.each(rules, function(prop, val) {
            // ignore rule when param is explicitly false, eg. required:false
            if (val === false) {
                delete rules[prop];
                return;
            }
            if (val.param || val.depends) {
                var keepRule = true;
                switch (typeof val.depends) {
                    case "string":
                        keepRule = !!(val.depends,
element.form).length;
                    break;

```

```

        case "function":
            keepRule = val.depends.call(element, element);
            break;
        }
        if (keepRule) {
            rules[prop] = val.param !== undefined ? val.param :
true;
        } else {
            delete rules[prop];
        }
    }
});

// evaluate parameters
$.each(rules, function(rule, parameter) {
    rules[rule] = $.isFunction(parameter) ? parameter(element) :
parameter;
});

// clean number parameters
$.each(['minlength', 'maxlength', 'min', 'max'], function() {
    if (rules[this]) {
        rules[this] = Number(rules[this]);
    }
});

$.each(['rangelength', 'range'], function() {
    if (rules[this]) {
        rules[this] = [Number(rules[this][0]), Number(rules[this][1])];
    }
});

```



```

    }
  });

  if ($.validator.autoCreateRanges) {
    // auto-create ranges
    if (rules.min && rules.max) {
      rules.range = [rules.min, rules.max];
      delete rules.min;
      delete rules.max;
    }
    if (rules.minlength && rules.maxlength) {
      rules.rangelength = [rules.minlength, rules.maxlength];
      delete rules.minlength;
      delete rules.maxlength;
    }
  }

  // To support custom messages in metadata ignore rule methods titled
  "messages"
  if (rules.messages) {
    delete rules.messages;
  }

  return rules;
},

// Converts a simple string to a {string: true} rule, e.g., "required" to {required:true}

```

```

normalizeRule: function(data) {
    if( typeof data == "string" ) {
        var transformed = {};
        $.each(data.split(/\s/), function() {
            transformed[this] = true;
        });
        data = transformed;
    }
    return data;
},

```

// <http://docs.jquery.com/Plugins/Validation/Validator/addMethod>

```

addMethod: function(name, method, message) {
    $.validator.methods[name] = method;
    $.validator.messages[name] = message != undefined ? message :
$.validator.messages[name];
    if (method.length < 3) {
        $.validator.addClassRules(name, $.validator.normalizeRule(name));
    }
},

```

```

methods: {

```

// <http://docs.jquery.com/Plugins/Validation/Methods/required>

```

required: function(value, element, param) {
    // check if dependency is met
    if ( !this.depend(param, element) )

```

this way

```
        return "dependency-mismatch";
switch( element.nodeName.toLowerCase() ) {
case 'select':
    // could be an array for select-multiple or a string, both are fine

    var val = $(element).val();
    return val && val.length > 0;
case 'input':
    if ( this.checkable(element) )
        return this.getLength(value, element) > 0;
default:
    return $.trim(value).length > 0;
}
},
```

// <http://docs.jquery.com/Plugins/Validation/Methods/remote>

```
remote: function(value, element, param) {
    if ( this.optional(element) )
        return "dependency-mismatch";

    var previous = this.previousValue(element);
    if (!this.settings.messages[element.name] )
        this.settings.messages[element.name] = { };

    previous.originalMessage =
this.settings.messages[element.name].remote;

    this.settings.messages[element.name].remote = previous.message;

    param = typeof param == "string" && { url:param } || param;
```

```

if ( this.pending[element.name] ) {
    return "pending";
}
if ( previous.old === value ) {
    return previous.valid;
}

previous.old = value;
var validator = this;
this.startRequest(element);
var data = {};
data[element.name] = value;
$.ajax($.extend(true, {
    url: param,
    mode: "abort",
    port: "validate" + element.name,
    dataType: "json",
    data: data,
    success: function(response) {
        validator.settings.messages[element.name].remote =
previous.originalMessage;
        var valid = response === true;
        if ( valid ) {
            var submitted = validator.formSubmitted;
            validator.prepareElement(element);
            validator.formSubmitted = submitted;

```

```

        validator.successList.push(element);
        validator.showErrors();
    } else {
        var errors = {};
        var message = response ||
validator.defaultMessage( element, "remote" );
        errors[element.name] = previous.message =
$.isFunction(message) ? message(value) : message;
        validator.showErrors(errors);
    }
    previous.valid = valid;
    validator.stopRequest(element, valid);
}
}, param));
return "pending";
},

// http://docs.jquery.com/Plugins/Validation/Methods/minlength
minlength: function(value, element, param) {
    return this.optional(element) || this.getLength($.trim(value), element)
    >= param;
},

// http://docs.jquery.com/Plugins/Validation/Methods/maxlength
maxlength: function(value, element, param) {
    return this.optional(element) || this.getLength($.trim(value), element)
    <= param;
},

```

```
// http://docs.jquery.com/Plugins/Validation/Methods/rangelength  
rangelength: function(value, element, param) {  
    var length = this.getLength($.trim(value), element);  
    return this.optional(element) || ( length >= param[0] && length <=  
param[1] );  
},
```

```
// http://docs.jquery.com/Plugins/Validation/Methods/min  
min: function( value, element, param ) {  
    return this.optional(element) || value >= param;  
},
```

```
// http://docs.jquery.com/Plugins/Validation/Methods/max  
max: function( value, element, param ) {  
    return this.optional(element) || value <= param;  
},
```

```
// http://docs.jquery.com/Plugins/Validation/Methods/range  
range: function( value, element, param ) {  
    return this.optional(element) || ( value >= param[0] && value <=  
param[1] );  
},
```

```
// http://docs.jquery.com/Plugins/Validation/Methods/email  
email: function(value, element) {  
    // contributed by Scott Gonzalez:  
http://projects.scottsplayground.com/email\_address\_validation/
```

```
return this.optional(element) || /^(([a-z]|\d|!#$%&'\*+|-
√=?\^_`{|}~)[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])+(\.([a-z]|\d|!#$%&'\*+|-
√=?\^_`{|}~)[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-
\uFFEF]+)*)((\x22)((\x20|\x09)*(\x0d\x0a))?(\\x20|\x09)+)?(([\x01-\x08\x0b\x0c\x0e-
\x1f\x7f]|x21|[\x23-\x5b]|[\x5d-\x7e]|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-
\uFFEF])|(\([\x01-\x09\x0b\x0c\x0d-\x7f]|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-
\uFFEF])))*)*((\x20|\x09)*(\x0d\x0a))?(\\x20|\x09)+)?(\x22))@((([a-z]|\d|[\u00A0-
\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-
\uFFEF])([a-z]|\d|-|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])*([a-
z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])))\.)+((([a-z]|\d|[\u00A0-\uD7FF\uF900-
\uFDCF\uFDF0-\uFFEF])|([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])([a-z]|\d|-
|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])*([a-z]|\d|[\u00A0-\uD7FF\uF900-
\uFDCF\uFDF0-\uFFEF])))$/i.test(value);
```

},

```
// http://docs.jquery.com/Plugins/Validation/Methods/url
```

```
url: function(value, element) {
```

```
// contributed by Scott Gonzalez:
```

```
http://projects.scottsplyground.com/iri/
```

```
return this.optional(element) || /^(https?|ftp):\\V(((([a-z]|\d|-
|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|(%[\da-
f]{2})|[!$&'\(\)\*+,\;=:]|@)?((\d|[1-9]\d|1\d\d|2[0-4]\d|25[0-5])\.)((\d|[1-9]\d|1\d\d|2[0-4]\d|25[0-5])\.)((\d|[1-9]\d|1\d\d|2[0-4]\d|25[0-5])\.)((\d|[1-9]\d|1\d\d|2[0-4]\d|25[0-5])\.)|([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])([a-z]|\d|-|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])*([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF]))\.)+((([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])([a-z]|\d|-|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])*([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF]))\.)?)(:(\d*))?(\V(((([a-z]|\d|-|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|(%[\da-f]{2})|[!$&'\(\)\*+,\;=:]|@)+(\V(((([a-z]|\d|-|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|(%[\da-f]{2})|[!$&'\(\)\*+,\;=:]|@)|[\uE000-\uF8FF])\V|(?)*)?(\#(((([a-z]|\d|-|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|(%[\da-f]{2})|[!$&'\(\)\*+,\;=:]|@)|\V|(?)*)?$/i.test(value);
```

},

```
// http://docs.jquery.com/Plugins/Validation/Methods/date
```

```

date: function(value, element) {
    return this.optional(element) || !/Invalid|NaN/.test(new Date(value));
},

// http://docs.jquery.com/Plugins/Validation/Methods/dateISO
dateISO: function(value, element) {
    return this.optional(element) || /^\d{4}[V-]\d{1,2}[V-
]\d{1,2}$/.test(value);
},

// http://docs.jquery.com/Plugins/Validation/Methods/number
number: function(value, element) {
    return this.optional(element) || /^-
?(?:\d+|\d{1,3}(?:,\d{3})+)(?:\.\d+)?$/ .test(value);
},

// http://docs.jquery.com/Plugins/Validation/Methods/digits
digits: function(value, element) {
    return this.optional(element) || /^\d+$/ .test(value);
},

// http://docs.jquery.com/Plugins/Validation/Methods/creditcard
// based on http://en.wikipedia.org/wiki/Luhn
creditcard: function(value, element) {
    if ( this.optional(element) )
        return "dependency-mismatch";
    // accept only spaces, digits and dashes
    if (/[^0-9 -]+/.test(value))

```



```

        return false;

    var nCheck = 0,
        nDigit = 0,
        bEven = false;

    value = value.replace(/\D/g, "");

    for (var n = value.length - 1; n >= 0; n--) {
        var cDigit = value.charAt(n);
        var nDigit = parseInt(cDigit, 10);
        if (bEven) {
            if ((nDigit * 2) > 9)
                nDigit -= 9;
        }
        nCheck += nDigit;
        bEven = !bEven;
    }

    return (nCheck % 10) == 0;
},

// http://docs.jquery.com/Plugins/Validation/Methods/accept
accept: function(value, element, param) {
    param = typeof param == "string" ? param.replace(/,/g, "|") :
    "png|jpe?g|gif";

    return this.optional(element) || value.match(new RegExp("(" + param
    + ")$", "i"));
},

```

```

// http://docs.jquery.com/Plugins/Validation/Methods/equalTo
equalTo: function(value, element, param) {
    // bind to the blur event of the target in order to revalidate whenever
the target field is updated

    // TODO find a way to bind the event just once, avoiding the unbind-
rebind overhead

    var target = $(param).unbind(".validate-equalTo").bind("blur.validate-
equalTo", function() {
        $(element).valid();
    });
    return value == target.val();
}

}

});

// deprecated, use $.validator.format instead
$.format = $.validator.format;

})(jQuery);

// ajax mode: abort

// usage: $.ajax({ mode: "abort"[, port: "uniqueport"]});

// if mode:"abort" is used, the previous request on that port (port can be undefined) is aborted
via XMLHttpRequest.abort()

;(function($) {

```

```

var pendingRequests = {};

// Use a prefilter if available (1.5+)
if ( $.ajaxPrefilter ) {
    $.ajaxPrefilter(function(settings, _, xhr) {
        var port = settings.port;
        if (settings.mode == "abort") {
            if ( pendingRequests[port] ) {
                pendingRequests[port].abort();
            }
            pendingRequests[port] = xhr;
        }
    });
} else {
    // Proxy ajax
    var ajax = $.ajax;
    $.ajax = function(settings) {
        var mode = ( "mode" in settings ? settings : $.ajaxSettings ).mode,
            port = ( "port" in settings ? settings : $.ajaxSettings ).port;
        if (mode == "abort") {
            if ( pendingRequests[port] ) {
                pendingRequests[port].abort();
            }
            return (pendingRequests[port] = ajax.apply(this, arguments));
        }
        return ajax.apply(this, arguments);
    };
}

```

```

})(jQuery);

// provides cross-browser focusin and focusout events

// IE has native support, in other browsers, use event capturing (neither bubbles)

// provides delegate(type: String, delegate: Selector, handler: Callback) plugin for easier
event delegation

// handler is only called when $(event.target).is(delegate), in the scope of the jquery-object
for event.target

;(function($) {

    // only implement if not provided by jQuery core (since 1.4)

    // TODO verify if jQuery 1.4's implementation is compatible with older jQuery
special-event APIs

    if (!jQuery.event.special.focusin && !jQuery.event.special.focusout &&
document.addEventListener) {

        $.each({

            focus: 'focusin',

            blur: 'focusout'

        }, function( original, fix ){

            $.event.special[fix] = {

                setup:function() {

                    this.addEventListener( original, handler, true );

                },

                teardown:function() {

                    this.removeEventListener( original, handler, true );

                },

                handler: function(e) {

                    arguments[0] = $.event.fix(e);

```

```

        arguments[0].type = fix;
        return $.event.handle.apply(this, arguments);
    }
};
function handler(e) {
    e = $.event.fix(e);
    e.type = fix;
    return $.event.handle.call(this, e);
}
});
};
$.extend($.fn, {
    validateDelegate: function(delegate, type, handler) {
        return this.bind(type, function(event) {
            var target = $(event.target);
            if (target.is(delegate)) {
                return handler.apply(target, arguments);
            }
        });
    }
});
})(jQuery);

```