**Form Validation**

1. **Client Registration Form**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Field type** | **Name** | **ID** | **Validation** | **Error Field** |
|  |  |  |  |  |  |
| E-mail | email | useremail | email | validate(email,emailerror,this.value) | emailerror |
| Create Password | password | pass1 | pass1 | validate(password,passerror,this.value) | pass1error |
| Confirm Password | password | pass2 | pass2 | script | pass2error |
| Company Name | text | compname | compnam | validate(company,comperror,this.value) | comperror |
| Company Type | radio | comtype[] | type1,type2 | validateRadio(comtype[],typeerror) | typeerror |
| Industry Tpe | select | indtype | indtype | required | inderror |
| Address | textarea | addr | addr | validate(address,addrerror,this.value) | addrerror |
| Pincode | text | pincode | pincode | validate(digit,pinerror,this.value) | pinerror |
| Person | text | person | person | validate(username,personerror,this.value) | personerror |
| Location (3 Elements): | select | country | country\_id | required |  |
|  | select | state | state\_id | required |  |
|  | select | city | city\_id | required |  |
|  |  |  |  |  |  |

1. **Job Posting Form**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Field type** | **Name** | **ID** | **Validation** | **Error Field** |
|  |  |  |  |  |  |
| Job title/ Designation | text | desig | desig | validate(text,deser,this.value) | deser |
| Vacancy | text | vacno | vac\_no | validate(digit,deser,this.value) | vacer |
| Job description | textarea | jobdesc | job\_desc | valdate(longtext,jober,this.value) | jober |
| Work experience | select | exp | exp | required |  |
| Basic Pay (2 Element): | select | money | money | required |  |
|  | text | pay | pay | validate(digit,deser,this.value) | payer |
| Functional area | text | fnarea | fnarea | required | fner |
| Location (3 Elements): | select | country | country\_id | required |  |
|  | select | state | state\_id | required |  |
|  | select | city | city\_id | required |  |
| Industry type | select | indtype | indtype | required |  |
| Ug course | select | **ugcourse** | **ugcourse** | required |  |
| Pg course | select | pgcourse | pgcourse | required |  |
| Required Candidate Profile | textarea | profile | profile | validate(longtext,proer,this.value) | proer |

1. **Job Seeker Registration**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Field type** | **Name** | **ID** | **Validation** | **Error Field** |
|  |  |  |  |  |  |
| E-mail | email | useremail | email | validate(email,emailerror,this.value) | emailerror |
| Create Password | password | pass1 | passnew | validate(password,passerror,this.value) | passerror |
| Confirm Password | password | pass2 | passconf | script | passerror2 |
| Full Name | text | uname | name | validate(username,nameerror,this.value) | nameerror |
| Location (3 Elements): | select | country | country\_id | required |  |
|  | select | state | state\_id | required |  |
|  | select | city | city\_id | required |  |
| Mobile Number | text | mobno | mobno | validate(mobilenum,mobnoerror,this.value) | mobnoerror |
| Experience | select | experience | experience | required |  |
| Skills | text | skills | skills | validate(text,skillerror,this.value) | skillerror |
| Ug course | select | **ugcourse** | **ugcourse** | required |  |
| Pg course | select | pgcourse | pgcourse | required |  |

function validate(type,field, query) {

var xmlhttp;

if (window.XMLHttpRequest) { // for IE7+, Firefox, Chrome, Opera, Safari

xmlhttp = new XMLHttpRequest();

} else { // for IE6, IE5

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

}

xmlhttp.onreadystatechange = function() {

if (xmlhttp.readyState != 4 && xmlhttp.status == 200) {

document.getElementById(field).innerHTML = "Validating..";

} else if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {

document.getElementById(field).innerHTML = xmlhttp.responseText;

} else {

document.getElementById(field).innerHTML = "Error Occurred. <a href='index.php'>Reload Or Try Again</a> the page.";

}

}

xmlhttp.open("GET", "/job\_portal/js/valid.php?type=" + type + "&query=" + query, true);

xmlhttp.send();

}

function validateRadio(name,errorfeild){

if ( ( document.getElementsByName(name)[0].checked == false ) && (document.getElementsByName(name)[1].checked == false ) )

{ document.getElementById(errorfeild).innerHTML="Select an option";}

else{

document.getElementById(errorfeild).innerHTML="";

}

}

<?php

$value = $\_GET['query'];

$formfield = $\_GET['type']; // type of validation to do

// Check Valid or Invalid user name when user enters user name in username input field.

if ($formfield == "username") {

if($value=="")

echo "Name Cant be empty";

else

if (strlen($value) < 3) {

echo "Must be 3+ letters";

}

}

// Check Valid or Invalid password when user enters password in password input field.

if ($formfield == "password") {

if($value=="")

echo "password cant be empty";

else

if (strlen($value) < 8) {

echo "Password too short";

}

}

// Check Valid or Invalid email when user enters email in email input field.

if ($formfield == "email") {

if($value=="")

echo "Enter your email";

else

if (!preg\_match("/([\w\-]+\@[\w\-]+\.[\w\-]+)/", $value)) {

echo "Invalid email";

}

}

// Check Valid or Invalid website address when user enters website address in website input field.

if ($formfield == "website") {

if (!preg\_match("/\b(?:(?:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?=~\_|!:,.;]\*[-a-z0-9+&@#\/%=~\_|]/i", $value)) {

echo "Invalid website";

}

}

// long text validation

if ($formfield == "longtext") {

if (strlen($value) <40 ) {

echo "Must be 40+ letters";

}

}

// address validation

if ($formfield == "address") {

if($value=="")

echo "Enter your address";

else

if (strlen($value) <20 ) {

echo "Must be 20+ letters";

}

}

//digit validation

if ($formfield == "digit") {

if(!is\_numeric($value))

{

echo "Input should be numeric";

}

}

//small text validation

if ($formfield == "text") {

if (strlen($value) <10 ) {

echo "Must be 10+ letters";

}

}

//mobile no validation

if ($formfield == "mobilenum") {

$mob="/^[1-9][0-9]\*$/"; // this is a regular expresssion

if($value=='')

echo "Enter your phone number";

else if ( ( !preg\_match($mob, $value) ) && ( strlen($value) < 10 )) {

echo "Enter a valid phone number";

}

}

//file uplad validation

if($formfield == "file"){

$allowed = array('doc','docx' ,'pdf');

$filename = $\_FILES["resumefile"]["name"];

$ext = pathinfo($filename, PATHINFO\_EXTENSION);

if(!in\_array($ext,$allowed) ) {

echo 'error';

}

}

// company name validation

if ($formfield == "company") {

if($value=="")

echo "Enter the company name!";

else

if (strlen($value) < 3) {

echo "Must be 3+ letters!";

}

}

if ($formfield=="pincode") {

if($value=="")

echo "Pincode cant be empty";

else if(!is\_numeric($value))

{

echo "Pincode should be numeric";

}

}

?>