Aim:- Design a registration form using form elements in ReactJS along with appropriate validations.

```
Program code:-
import React from 'react';
import './App.css';
class RegisterForm extends React.Component {
  constructor() {
   super();
   this.state = {
    fields: {},
    errors: {}
   }
   this.handleChange = this.handleChange.bind(this);
   this.submituserRegistrationForm = this.submituserRegistrationForm.bind(this);
  };
  handleChange(e) {
   let fields = this.state.fields;
   fields[e.target.name] = e.target.value;
   this.setState({
    fields
   });
  }
  submituserRegistrationForm(e) {
   e.preventDefault();
   if (this.validateForm()) {
     let fields = {};
     fields["username"] = "";
     fields["emailid"] = "";
     fields["mobileno"] = "";
     fields["password"] = "";
     this.setState({fields:fields});
     alert("Form submitted");
   }
  }
  validateForm() {
   let fields = this.state.fields;
   let errors = {};
   let formIsValid = true;
   if (!fields["username"]) {
    formIsValid = false;
```

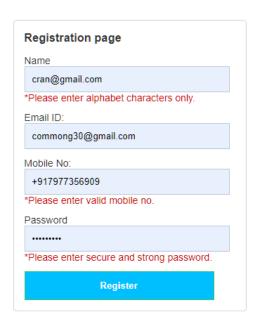
errors["username"] = "*Please enter your username.";

}

```
if (typeof fields["username"] !== "undefined") {
    if (!fields["username"].match(/^[a-zA-Z]*$/)) {
     formIsValid = false;
     errors["username"] = "*Please enter alphabet characters only.";
   }
   if (!fields["emailid"]) {
    formIsValid = false;
    errors["emailid"] = "*Please enter your email-ID.";
   }
   if (typeof fields["emailid"] !== "undefined") {
    //regular expression for email validation
    var pattern = new
RegExp(/^(("[\w-\s]+")|([\w-]+(?:\.[\w-]+)*))(@((?:[\w-]+\.)))
*\w[\w-]{0,66})\.([a-z]{2,6}(?:\.[a-z]{2})?)$)|(@\[?((25[0-5]\.|2[0-4][0-9]\.|1[0-9]{2}\.|[0-9]{
1,2}\.))((25[0-5]|2[0-4][0-9]|1[0-9]{2}|[0-9]{1,2})\.){2}(25[0-5]|2[0-4][0-9]|1[0-9]{2}|[0-9]{1,
2})\]?$)/i);
    if (!pattern.test(fields["emailid"])) {
     formIsValid = false;
     errors["emailid"] = "*Please enter valid email-ID.";
    }
   if (!fields["mobileno"]) {
    formIsValid = false;
    errors["mobileno"] = "*Please enter your mobile no.";
   if (typeof fields["mobileno"] !== "undefined") {
    if (!fields["mobileno"].match(/^[0-9]{10}$/)) {
     formIsValid = false;
     errors["mobileno"] = "*Please enter valid mobile no.";
    }
   if (!fields["password"]) {
    formIsValid = false;
    errors["password"] = "*Please enter your password.";
   if (typeof fields["password"] !== "undefined") {
(!fields["password"].match(/^.*(?=.{8,})(?=.*\d)(?=.*[a-z])(?=.*[A-Z])(?=.*[@#$%&]).*$/)) {
     formIsValid = false;
     errors["password"] = "*Please enter secure and strong password.";
    }
   }
   this.setState({
    errors: errors
```

```
});
   return formIsValid;
 }
 render() {
  return (
  <div id="main-registration-container">
  <div id="register">
    <h3>Registration page</h3>
    <form method="post" name="userRegistrationForm" onSubmit=
{this.submituserRegistrationForm} >
    <label>Name</label>
    <input type="text" name="username" value={this.state.fields.username}
onChange={this.handleChange} />
    <div className="errorMsg">{this.state.errors.username}</div>
    <label>Email ID:</label>
    <input type="text" name="emailid" value={this.state.fields.emailid}
onChange={this.handleChange} />
    <div className="errorMsg">{this.state.errors.emailid}</div>
    <label>Mobile No:</label>
    <input type="text" name="mobileno" value={this.state.fields.mobileno}
onChange={this.handleChange} />
    <div className="errorMsg">{this.state.errors.mobileno}</div>
    <label>Password</label>
    <input type="password" name="password" value={this.state.fields.password}
onChange={this.handleChange} />
    <div className="errorMsg">{this.state.errors.password}</div>
    <input type="submit" className="button" value="Register"/>
    </form>
  </div>
</div>
   );
}
export default RegisterForm;
```

Output:-



Conclusion:-Successfully designed a register form with proper validations