

Ex : 1

1. activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/hello_world_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello, World!"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

2. MainActivity.java

```
package com.example.helloworld;

import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    @Override
    public boolean onCreateOptionsMenu (Menu menu) {
        getMenuInflater().inflate(R.menu.main, menu);
    }
}
```

```
return true;
}
}
```

Ex : 2

1. activity_main.xml

```
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/click_me" />
    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World" />
</FrameLayout>
```

2. MainActivity.java

```
package com.example. EventListener;
import android.os.Bundle;
import android.app.Activity;
```

```
import android.view.Menu;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

public class MainActivity extends Activity {

    private Button button1;

    private TextView textView1;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        button1 = (Button) findViewById(R.id.button1);

        textView1 = (TextView) findViewById(R.id.textView1);

        button1.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View arg0) {

                textView1.setText("Button Clicked");

            }

        });

        @Override

        public boolean onCreateOptionsMenu (Menu menu) {

            // Inflate the menu; this adds items to the action bar if it is present.

            getMenuInflater().inflate(R.menu.main, menu);

            return true;

        }

    }

}
```

Ex : 3

APP.js

```
import React, {useState} from 'react'
import './index.css'

function App () {
  // state
  const [weight, setWeight] = useState (0)
  const [height, setHeight] = useState (0)
  const [bmi, setBmi] = useState("")
  =
  const [message, setMessage] = useState("")
  let calcBmi = (event) =>
  {
    //prevent submitting
    event.preventDefault()
    if (weight
    {
      === 0 || height === 0)
      alert('Please enter a valid weight and height')
    }
    else
    {
      let bmi = (weight / (height *height) * 703)
```

```
setBmi (bmi. to Fixed (1))

// Logic for message
if (bmi < 25)
{
  setMessage("You are underweight")
} else if (bmi >= 25 && bmi < 30)
{
  setMessage("You are a healthy weight")
} else
  setMessage("You are overweight")
}
}
}

let reload
() =>
window.location.reload()

{
}

return
)

<div className="app">
<div className='container'>

<h2 className='center'>BMI Calculator</h2>

<form onSubmit={calcBmi} >

<div>

<label>Weight (lbs) </label>
```

```

<input value={weight} onChange={(e) => setWeight (e.target.value)} />
</div>
<div>
<label>Height (in)</label>
<input value={height} onChange={(event) => set Height (event.target.value)} />
</div>
<div>
<button className='btn' type='submit'>Submit</button>
<button className='btn btn-outline' onClick={reload}
type='submit'>Reload</button>
</div>
</form>
<div className='center'>
<h3>Your BMI is: {bmi}</h3>
<p> {message}</p>
</div>
<div className='img-container'>
<img src={img Src} alt=''></img>
</div>
</div>
</div>
);
}
export default App;

```

Index.css

```
box-sizing: border-box;

margin: 0;

padding: 0;

.app {

}

display: flex;

flex-direction: column;

justify-content: center;

align-items: center;

width: 100%;

height: 100vh;

.container {

box-shadow: 0px 0px 12px #ccc;

border-radius: 8px;

padding: 1rem;

input (

width: 100%;

font-size: 1.2rem;

padding: 8px 4px;

margin: 8px 0;

border-radius: 8px;

.bt (

display: block;,

width: 100%;

font-size: 1.2rem;

margin: 8px 0;
```

```
padding: 10px 0;
background-color: #333;
color: #fff;
border: 1px solid #333;
border-radius: 8px;
cursor: pointer;
.btn-outline {
background-color: #fff;
color: #333;
.center {
text-align: center;
margin: 24px 0;
p
margin: 10px 0;
.img-container {
text-align: center;
.img-container img {
}
height: 200px;
body {
margin: 0;
font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto',
'Oxygen',
'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
sans-serif;
-webkit-font-smoothing: antialiased;
```



```
-moz-osx-font-smoothing: grayscale;

code {
font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
monospace;
}
```

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
const root = ReactDOM.createRoot (document.getElementById('root'));
root.render(
  <React.StrictMode>
  <App />
</React.StrictMode>
);
```

Ex : 4

APP.is

```
import React, { useState } from 'react';
import AddTransaction from './components/AddTransaction';
import Transaction List from './components/TransactionList';
import Weekly Report from './components/WeeklyReport';
import
```

```

./App.css';

function App () {
  const [transactions, setTransactions] = useState([]);
  const addTransaction = (transaction) => {
    setTransactions ([...transactions, transaction]);
  };
  return (
    <div className="App">
      <h1>Expense Manager</h1>
      <AddTransaction addTransaction={addTransaction} />
      <TransactionList transactions={transactions} />
      <WeeklyReport transactions={transactions} />
    </div>
  );
  export default App:

```

1. Create a new Folder and rename as components and create a need JS file as

Add Transaction.js

Transaction List.js

Weekly Report.js

Add Transaction.js

```

import React, { useState } from 'react';

function AddTransaction({ addTransaction }) {
  const [date, setDate] = useState("");
  const [category, setCategory] = useState("");

```

```

const [type, setType] useState('income');

const [amount, setAmount] = useState("");

const handleSubmit = (e) => {
  e.preventDefault();

  const transaction
  { date, category, type, amount: parseFloat (amount);
  addTransaction (transaction);
  setDate("");
  setCategory("");
  set Amount
  ');
  };
  return (
    <form onSubmit={handleSubmit}>
    <div>
    <label>Date: </label>
    <input type="date" value={date} onChange={(e) => set Date (e.target.value)}
    required />
    </div>
    <div>
    <label>Category:</label>
    <input type="text" value={category} onChange={(e) =>
    setCategory (e.target.value)} required />
    </div>
    <div>
    <label>Type:</label>

```

```

<select value={type} onChange={e => setType(e.target.value)}>
  <option value="income">Income</option>
  <option value="expense">Expense</option>
</select>
</div>
<div>
  <label>Amount:</label>
  <input type="number" value={amount} onChange={e => setAmount (e.target.value) }
  required />
</div>
<button type="submit">Add Transaction</button>
</form>
);
export default AddTransaction;

```

Transaction List.js

```

import React from 'react';

function TransactionList({ transactions })
return
<div>
  <h2>Transaction List</h2>
  <ul>
    {transactions.map((transaction, index) => (
      <li key={index} >
        {transaction.date} - {transaction.category}-{transaction.type}-
        ${transaction.amount}
      </li>
    ))}
  </ul>

```

```

</li>
  )})
</ul>
</div>

};

}

export default TransactionList;

```

Weekly Report.js

```

import React from 'react';

function Weekly Report ({ transactions }) {
  const getWeekly Report = () => {
    const report = {};

    transactions.forEach((transaction) => {
      const category = transaction.category;
      const amount = transaction. amount;

      if (!report [category]) {
        report [category] = { income: 0, expense: 0 };
      }

      if (transaction.type === 'income') {
        report [category].income += amount;
      }

      else if (transaction.type
        'expense') {
        report [category].expense += amount;
      }
    });
  };
}

```

```
});  
return report;  
};  
const report = getWeekly Report ();  
return  
<div>  
  <h2>Weekly Report</h2>  
  <table>  
    <thead>  
      <tr>  
        <th>Category</th>  
        <th>Total Income</th>  
        <th>Total Expense</th>  
      </tr>  
    </thead>  
    <tbody>  
      {Object.keys(report).map((category, index) => (  
        <tr key={index}>  
          <td> [category]</td>  
          <td>${report [category].income}</td>  
          <td>${report [category].expense}</td>  
        </tr>  
      )) }  
    </tbody>  
  </table>  
</div>
```

```
);
```

```
export default Weekly Report;
```

```
Index.css
```

```
Body
```

```
{
```

```
margin: 0;
```

```
font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
```

```
background-color: #f4f4f4;
```

```
color: #333;
```

```
}
```

```
h1, h2
```

```
{
```

```
color: #444;
```

```
text-align: center;
```

```
margin-bottom: 20px;
```

```
. App
```

```
max-width: 800px;
```

```
margin: 0 auto;
```

```
padding: 20px;
```

```
background-color: #fff;
```

```
border-radius: 8px;
```

```
box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
```

```
}
```

```
/* Form Styling */
```

```
form {
```

```
display: flex;
flex-direction: column;
gap: 10px;
margin-bottom: 30px;
padding: 20px;
background-color: #f9f9f9;
border-radius: 8px;
box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
form div
}
display: flex;
flex-direction: column;
label
font-size: 14px;
margin-bottom: 5px;
}
input, select
padding: 10px;
font-size: 14px;
border: 1px solid #ddd;
border-radius: 4px;
transition: border-color 0.3s ease;
}
input: focus, select: focus
border-color: #4CAF50;
outline: none;
```



```
button (  
padding: 12px;  
font-size: 16px;  
background-color: #4CAF50;  
color: white;  
border: none;  
border-radius: 4px;  
cursor: pointer;  
transition: background-color 0.3s ease;  
}  
button:hover  
{  
background-color: #45a049;  
}  
/* Transaction List Styling */  
Ul  
{  
list-style-type: none;  
padding: 0;  
Li  
{  
padding: 10px;  
background-color: #fafafa;  
border: 1px solid #ddd;  
border-radius: 4px;  
margin-bottom: 10px;
```

```
box-shadow: 0 1px 3px rgba(0, 0, 0, 0.1);
```

```
}
```

```
/* Table Styling */
```

```
table {
```

```
width: 100%;
```

```
border-collapse: collapse;
```

```
margin: 20px 0;
```

```
box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
```

```
}
```

```
th, td
```

```
{
```

```
padding: 12px;
```

```
text-align:
```

```
left;
```

```
border-bottom: 1px solid #ddd;
```

```
Th
```

```
background-color: #4CAF50;
```

```
color: white;
```

```
}
```

```
tr:nth-child(even)
```

```
{
```

```
background-color: #f9f9f9;
```

```
}
```

```
tr:hover
```

```
{
```

```
background-color: #f1f1f1;
```

```
}  
  
/* Responsive Design */  
  
@media (max-width: 768px) {  
  
  .App  
  
    padding: 15px;  
  
    form {  
  
      padding: 15px;  
  
      Button  
  
        font-size: 14px;  
  
      table, th, td  
  
        font-size: 14px;  
  

```

```
Index.js  
  
import React from 'react';  
import ReactDOM from 'react-dom/client';  
import './index.css';  
import App from './App';  
  
//import reportWebVitals from './reportWebVitals';  
  
=  
  
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(  
  <React.StrictMode>  
    <App />  
  </React.StrictMode>  
);  
  
//reportWebVitals();
```

Ex : 5

Modify index.html to Display Location

```
<!DOCTYPE html>

<html>

<head>

<title>My Location App</title>

<meta charset="utf-8" />

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

</head>

<body>

<h1>Current Location</h1>

<div id="location">

<p>Latitude: <span id="latitude">Loading...</span></p>

<p>Longitude: <span id="longitude">Loading...</span></p>

</div>

<script src="cordova.js"></script>

<script src="js/index.js"></script>

</body>

</html>
```

Step 5: Update JavaScript to Get and Display Location

```
var app = {

  initialize: function () {

    document.addEventListener('device ready', this.on DeviceReady.bind(this), false);

  },
```

```

on DeviceReady: function () {
  this.getLocation();
  getLocation: function () {
    if (navigator.geolocation) {
      navigator.geolocation.getCurrentPosition(this.onSuccess, this.onError);
    } else {
      alert("Geolocation is not supported by this browser.");
    }
  },
  onSuccess: function (position) {
    document.getElementById("latitude").textContent
    ==
    position.coords.latitude;
    document.getElementById("longitude").textContent = position.coords.longitude;
  },
  onError: function (error) {
    alert("Error getting location: " + error.message);
  },
};
app.initialize();

```

Ex : 6

Modify index.html for the Login Screen

Open www/index.html and replace its contents with the following code:

```
<!DOCTYPE html>
```

```
<html>

<head>

<meta charset="utf-8">

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

<title>Login Screen</title>

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

</head>

<body>

<div class="login-container">

<!-- Header Image -->



<!-- Login Label -->

<h2>Login</h2>

<!-- Login Form -->

<form id="loginForm">

<label for="username">Username</label>

<input type="text" id="username" name="username" required>

<label for="password">Password</label>

<input type="password" id="password" name="password" required>

<!-- Buttons -->

<button type="reset" class="button reset-button">Reset</button>

<button type="submit" class="button submit-button">Submit</button>

</form>

</div>

<!-- Add your Cordova script -->

<script src="cordova.js"></script>
```

```
<script src="js/app.js"></script>
```

```
</body>
```

```
</html>
```

Step 4: Style the Login Screen with CSS

Create a new CSS file named `www/css/styles.css` and add the following styles:

```
}
```

```
Margin: 0;
```

```
Padding: 0;
```

```
Box-sizing: border-box;
```

```
/* Basic Container Style */
```

```
Body {
```

```
Font-family: Arial, sans-serif;
```

```
Display: flex;
```

```
Align-items: center;
```

```
Justify-content: center;
```

```
Height: 100vh;
```

```
Background-color: #f0f0f0;
```

```
}
```

```
/* Login Container */
```

```
.login-container {
```

```
Width: 90%;
```

```
Max-width: 400px;
```

```
Padding: 20px;
```

```
Background: #fff;
```

```
Box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
```

```
Text-align: center;
```

```
Border-radius: 8px;

/* Header Image */
.header-image
width: 100%;
max-height: 100px;
margin-bottom: 20px;

/* Form and Labels */
form {
display: flex;
flex-direction: column;
label {
margin-top: 10px;
text-align: left;
input[type="text"],
input[type="password"] {
padding: 10px;
margin-top: 5px;
border-radius: 4px;
border: 1px solid #ccc;

/* Buttons */
.button {
padding: 10px;
margin-top: 15px;
border: none;
border-radius: 4px;
color: #fff;
```



```
cursor: pointer;
```

```
}
```

```
.reset-button {
```

```
Background-color: #3d9534f%;B
```

```
.submit-button {
```

```
Background-color: #5cb85c;
```

```
}
```