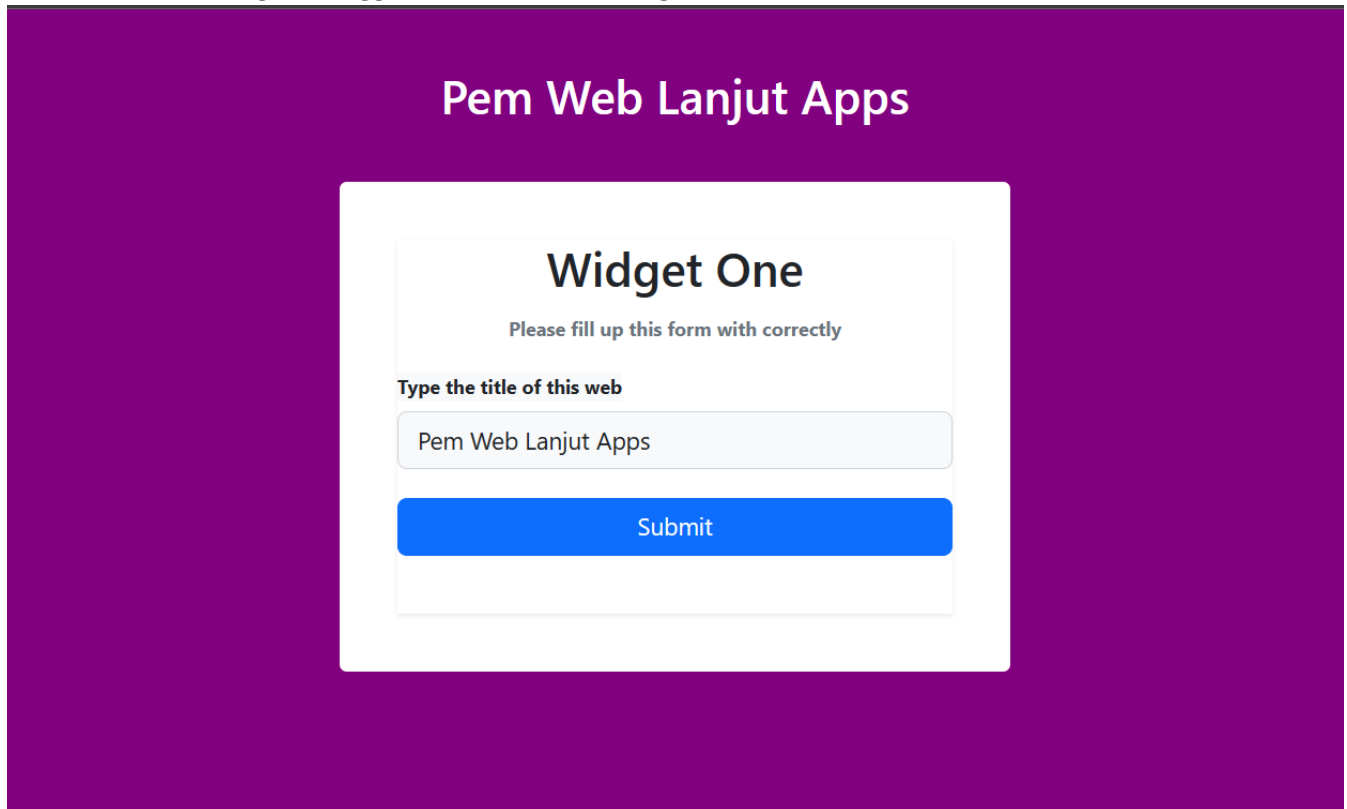


REACT JS STATEMENTS AND HANDLER EVENT

1. Pada project sebelumnya buatlah sebuah package baru dengan nama latihan-5_1. Ubahlah script pada contoh kasus Gambar 4, dengan menggunakan RCC untuk Widget1.



Layout.js

```
2. Import React from "react";
3.
4. Const Layout = (props) => {
5.   Return (
6.     <div
7.       Classname="d-flex flex-column flex-root vh-100"
8.       Style={{ backgroundColor: "purple" }}
9.     >
10.      <div classname="d-flex flex-column flex-column-fluid">
11.        <div classname="d-flex flex-center flex-column flex-column-fluid p-10 pb-lg-20">
12.          <h1 classname="mb-12 text-white text-center my-5 fs-8">{props.title}</h1>
13.          <div classname="w-50 bg-body rounded p-5 mx-auto">
14.            {props.children}
15.          </div>
16.        </div>
17.      </div>
18.    </div>
19.  );
20.};
21.
22.Export default Layout;
```

Widget1.js

```
import React, { Component } from "react";
import Layout from "../Layout";

class Widget1 extends Component {
  constructor(props) {
    super(props);
    this.state = {
      Title: "Pem Web Lanjut Apps",
    };
  }

  Newtitle = (newtitle) => {
    this.setState({ title: newtitle });
  };

  Render() {
    return (
      <Layout title={this.state.title}>
        <div classname="shadow-sm">
          <form method="post" autocomplete="off" classname="form">
            <div classname="text-center mb-10">
              <h1 classname="text-dark mb-3">Widget One</h1>
              <div classname="text-secondary fw-bold fs-10 mb-3">
                Please fill up this form with correctly
              </div>
            </div>

            <div classname="fv-row mb-10 fv-plugins-icon-container">
              <label classname="form-label fs-6 fw-bolder text-dark my-2 bg-light">
                Type the title of this web
              </label>
              <input
                Type="text"
                Name="title"
                Classname="form-control form-control-lg form-control-solid bg-light"
                Placeholder="Pem Web Lanjut Apps"
                Defaultvalue={this.state.title}
                Onchange={(e) => this.newtitle(e.target.newtitle)}
              />
            </div>

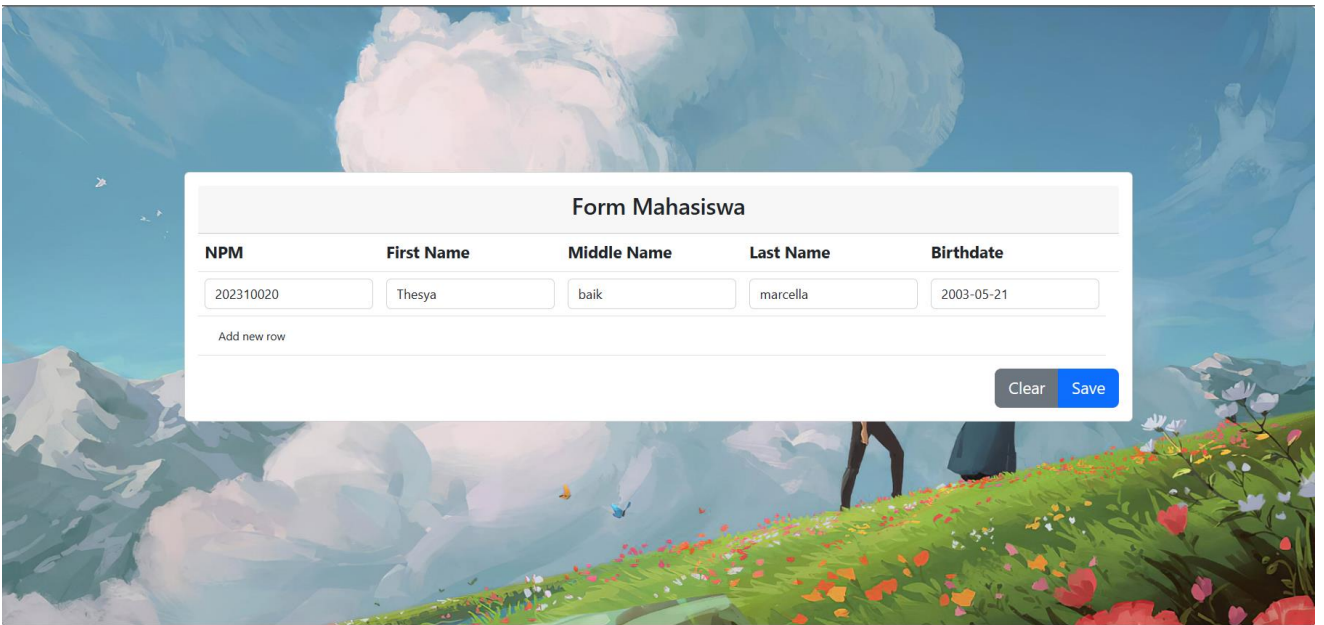
            <div classname="text-center">
              <button
                Type="button"
                Classname="btn btn-lg btn-primary w-100 mb-5 my-4"
              >
```

```
        Submit
    </button>
</div>
</form>
</div>
</Layout>
);
}
}

Export default Widget1;
```

23. Pada tugas praktikum sebelumnya tentang form biodata anda, ubahlah tugas tersebut sehingga mampu menampung data dalam bentuk multiple.

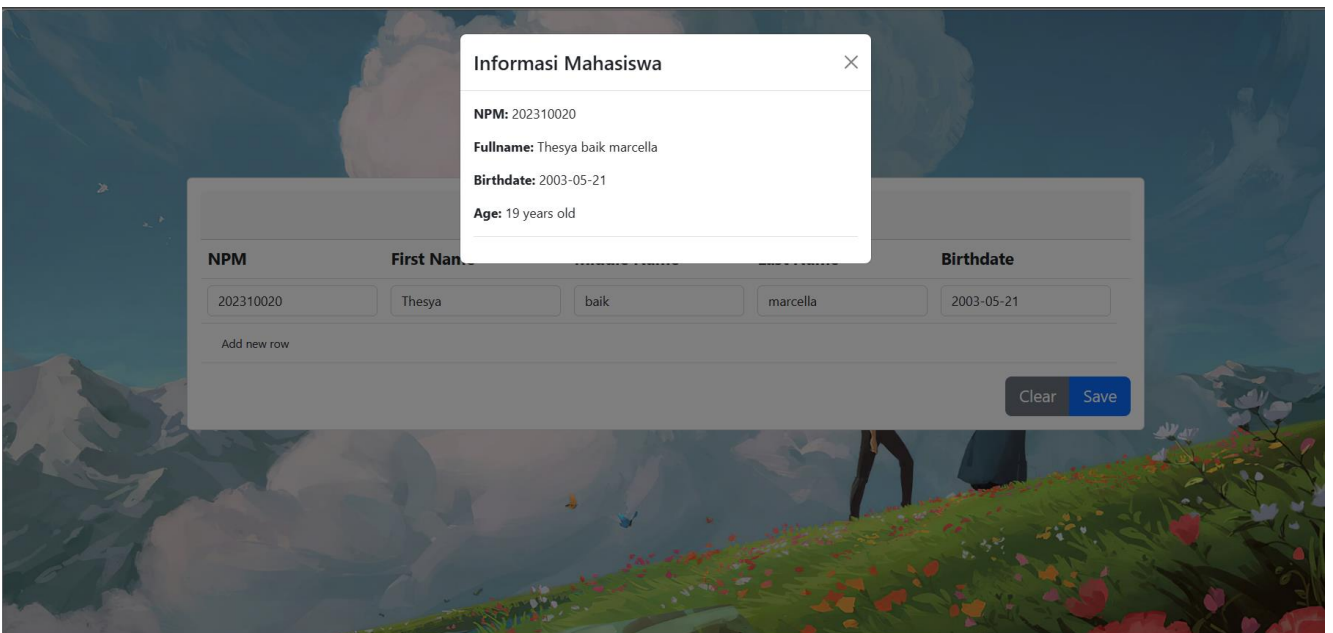
A. Dokumentasi screenshot



Form Mahasiswa				
NPM	First Name	Middle Name	Last Name	Birthdate
202310020	Thesya	baik	marcella	2003-05-21
Add new row				

Clear Save

23.1 Form Mahasiswa dengan add new row



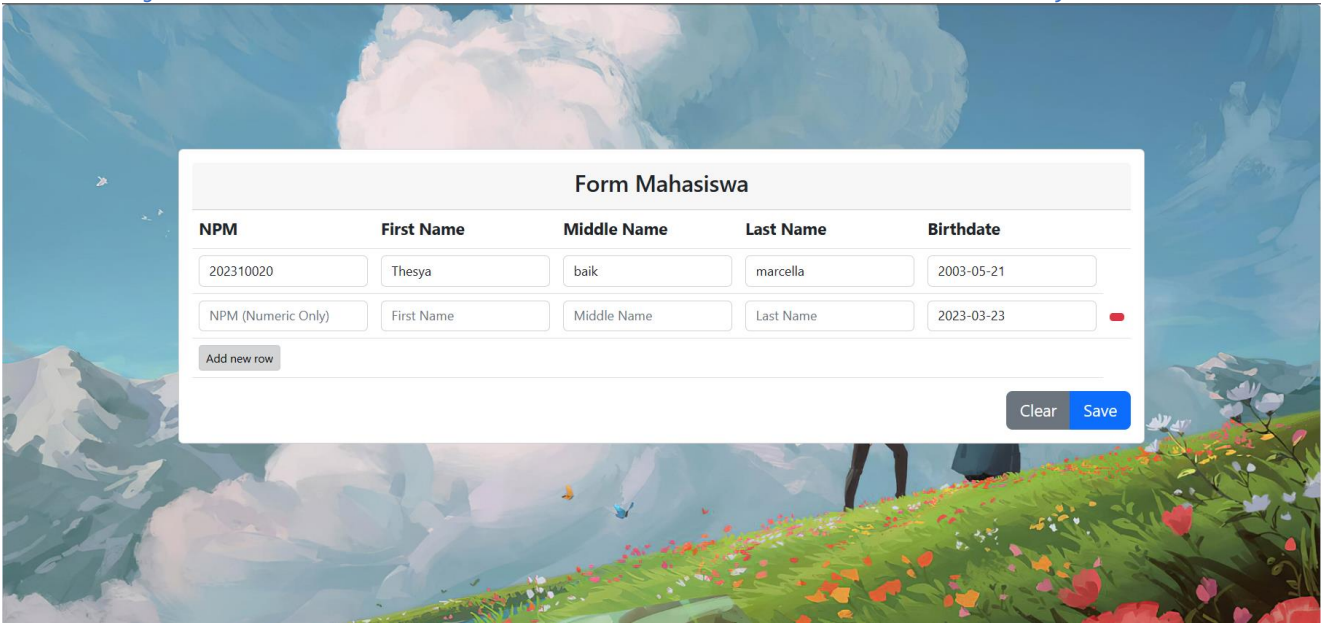
Informasi Mahasiswa ✕

NPM: 202310020
Fullname: Thesya baik marcella
Birthdate: 2003-05-21
Age: 19 years old

NPM	First Name	Middle Name	Last Name	Birthdate
202310020	Thesya	baik	marcella	2003-05-21
Add new row				

Clear Save

23.2 Pop up modal berupa data



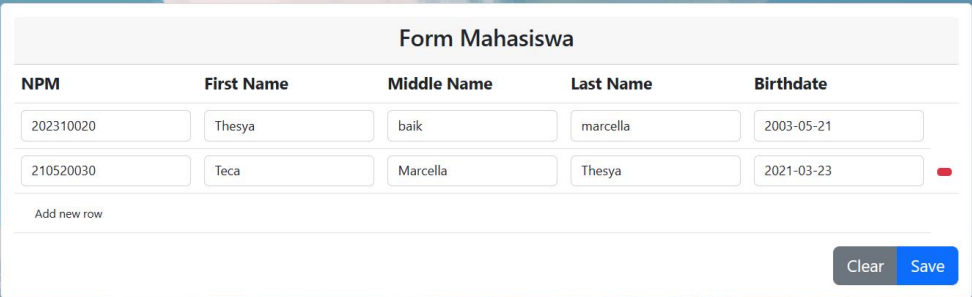
Form Mahasiswa

NPM	First Name	Middle Name	Last Name	Birthdate
202310020	Thesya	baik	marcella	2003-05-21
NPM (Numeric Only)	First Name	Middle Name	Last Name	2023-03-23

Add new row

Clear Save

23.3 Row baru ketika tombol add new row di klik



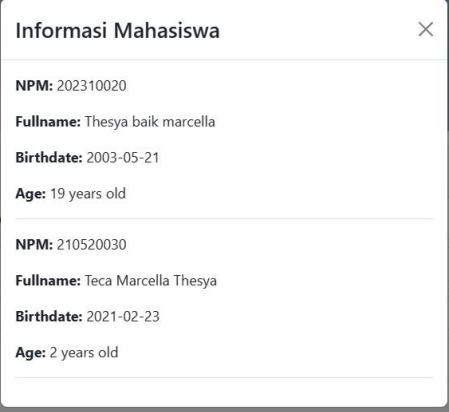
The image shows a web form titled "Form Mahasiswa" (Student Form) overlaid on a scenic background of a grassy hill with flowers and mountains under a blue sky with clouds. The form has a table with five columns: NPM, First Name, Middle Name, Last Name, and Birthdate. It contains two rows of data. The first row has NPM 202310020, First Name Thesya, Middle Name baik, Last Name marcella, and Birthdate 2023-05-21. The second row has NPM 210520030, First Name Teca, Middle Name Marcella, Last Name Thesya, and Birthdate 2021-03-23. Below the table is a link "Add new row". At the bottom right of the form are "Clear" and "Save" buttons.

NPM	First Name	Middle Name	Last Name	Birthdate
202310020	Thesya	baik	marcella	2023-05-21
210520030	Teca	Marcella	Thesya	2021-03-23

[Add new row](#)

[Clear](#) [Save](#)

23.4 Penambahan Data Baru



The image shows a modal window titled "Informasi Mahasiswa" (Student Information) overlaid on the same background as the previous image. The modal displays the details of the student with NPM 202310020. It shows the full name as "Thesya baik marcella", the birthdate as "2023-05-21", and the age as "19 years old". Below this, it shows the details of the student with NPM 210520030, with the full name "Teca Marcella Thesya", the birthdate "2021-02-23", and the age "2 years old". The modal has a close button (X) in the top right corner. In the background, the "Form Mahasiswa" is visible, showing the same table as before.

Informasi Mahasiswa ✕

NPM: 202310020
Fullname: Thesya baik marcella
Birthdate: 2023-05-21
Age: 19 years old

NPM: 210520030
Fullname: Teca Marcella Thesya
Birthdate: 2021-02-23
Age: 2 years old

2.5 Modal ketika data baru ditambahkan

B. Source Code Form

```
Import React, { useState } from "react";
Import { Modal, Card, Form } from "react-bootstrap";
Import datepicker from "react-datepicker";
Import "react-datepicker/dist/react-datepicker.css";
Import Background from "../../assets/wallpaper.jpg";

Export default function multipledataform() {
  Var today = new Date();
  Const currdate = formatdate(today);
  Const objprofile = {
    Npm: 0,
    Fname: "",
    Mname: "",
    Lname: "",
    Birthdate: currdate,
  };
  Const [postdata, setpostdata] = useState([objprofile]);
  Const [mymodal, setmymodal] = useState({
    Show: false,
    Message: "",
  });
  Const modalhandleclose = () => {
    Setmymodal({ ...mymodal, show: false });
  };
  Const [show, setshow] = useState(false);
  Const [validated, setvalidated] = useState(false);

  Const rowitem = ({ data, index }) => {
    Return (
      <tr>
        <td>
          <input
            Type="text"
            Name="npm"
            Classname="form-control"
            Placeholder="NPM (Numeric Only)"
            Pattern="^[0-9]*$"
            Defaultvalue={data.npm ? Data.npm : ""}
            Onmouseleave={(e) => {
              Setpostdata((prev) => {
                Const newdata = [...prev];
                Newdata[index].npm = parseInt(e.target.value);
                Return newdata;
              })
            })
          />
        />
      />
    )
  }
}
```

```
        Required
      />
    </td>
    <td>
      <input
        Type="text"
        Name="fname"
        Classname="form-control"
        Placeholder="First Name"
        Defaultvalue={data.fname}
        Onmouseleave={(e) =>
          Setpostdata((prev) => {
            Const newdata = [...prev];
            Newdata[index].fname = e.target.value;
            Return newdata;
          })
        }
        Required
      />
    </td>
    <td>
      <input
        Type="text"
        Name="mname"
        Classname="form-control"
        Placeholder="Middle Name"
        Defaultvalue={data.mname}
        Onmouseleave={(e) =>
          Setpostdata((prev) => {
            Const newdata = [...prev];
            Newdata[index].mname = e.target.value;
            Return newdata;
          })
        }
      />
    </td>
    <td>
      <input
        Type="text"
        Name="lname"
        Classname="form-control"
        Placeholder="Last Name"
        Defaultvalue={data.lname}
        Onmouseleave={(e) =>
          Setpostdata((prev) => {
            Const newdata = [...prev];
            Newdata[index].lname = e.target.value;
            Return newdata;
          })
        }
      />
    </td>
```



```

    }
    Required
  />
</td>
<td>
  <datepicker
    Selected={new Date(data.birthdate)}
    Onchange={(value) =>
      Setpostdata((prev) => {
        Const newdata = [...prev];
        Newdata[index].birthdate = formatdate(value);
        Return newdata;
      })
    }
    Classname="form-control"
    Dateformat="yyyy-MM-dd"
    Placeholder="Birthdate"
    Required
  />
</td>
{index !== 0 ? (
  <td>
    <button
      Classname="btn btn-icon btn-sm btn-danger"
      Type="button"
      Onclick={(e) =>
        Setpostdata((prev) => {
          Const newdata = [...prev];
          Delete newdata[index];
          Return newdata;
        })
      }
    >
    <i classname="bi bi-trash"></i>
  </button>
</td>
) : (
  ""
)}
</tr>
);
};

Const resultpost = (obj) => {
  Return (
    <div classname="result">
      {obj.map((v, index) =>
        v ? (
          <div classname="border-bottom mb-3" key={index}>

```

```

        <p>
            <span classname="fw-bolder">NPM:</span> {v.npm}
        </p>
        <p>
            <span classname="fw-bolder">Fullname:</span> {v.fname} {v.mname}{ " "}
            {v.lname}{ " "}
        </p>
        <p>
            <span classname="fw-bolder">Birthdate:</span> {v.birthdate}{ " "}
        </p>
        <p>
            <span classname="fw-bolder">Age:</span>{ " "}
            {calculateage(v.birthdate)} years old{ " "}
        </p>
    </div>
    ) : (
        ""
    )
    })
</div>
);
};

Const handlersubmit = (e) => {
    E.preventDefault();
    Let myresult = resultpost(postdata);
    Setmymodal({ ...mymodal, show: true, message: myresult });
    Const form = e.currentTarget;

    If (form.checkvalidity() === false) {
        E.preventDefault();
        E.stopPropagation();
    } else {
        E.preventDefault();
        Setshow(true);
    }
    Setvalidated(true);
};

Return (
    <Card
        Classname="d-flex justify-content-center align-items-center vh-100"
        Style={{
            Backgroundimage: `url(${Background})`,
            Backgroundrepeat: "no-repeat",
            Backgroundsize: "cover",
            Backgroundposition: "center",
        }}
    >

```

```

<div id="form-multiple" classname="card mb-5 p-3">
  <div classname="card-header">
    <div classname="card-title text-center">
      <h3>Form Mahasiswa</h3>
    </div>
  </div>
  <div classname="card-body p-0">
    <form
      Method="post"
      Autocomplete="off"
      Onsubmit={(e) => handlersubmit(e)}
    >
      <table classname="table table-flush align-middle table-row-bordered table-
row-solid gy-4 gs-9">
        <thead classname="border-gray-200 fs-5 fw-bold bg-lighten">
          <tr>
            <th>NPM</th>
            <th>First Name</th>
            <th>Middle Name</th>
            <th>Last Name</th>
            <th>Birthdate</th>
            <th></th>
          </tr>
        </thead>
        <tbody>
          {postData.map((v, index) =>
            V ? <rowitem key={index} data={v} index={index} /> : ""
          )}
        </tbody>
        <tfoot>
          <tr>
            <td colspan={5}>
              <button
                Classname="btn btn-sm btn-light"
                Type="button"
                Onclick={() =>
                  SetpostData((prev) => {
                    Return [...prev, objprofile];
                  })
              }
            >
              <i classname="bi bi-plus-circle"></i>
              Add new row
            </button>
          </td>
        </tr>
        </tfoot>
      </table>
    </div>
  </div>

```

```

        <div classname="text-end p-10">
            <div classname="btn-group">
                <button classname="btn btn-lg btn-secondary" type="button">
                    Clear
                </button>
                <button classname="btn btn-lg btn-primary" type="submit">
                    Save
                </button>
            </div>
        </div>
    </form>
</div>
<modalpopup
    Show={mymodal.show}
    Handleclose={modalhandleclose}
    Message={mymodal.message}
/>
</div>
</Card>
);
}

Const formatdate = (datestring) => {
    Const today = new Date(datestring);
    Const yyyy = today.getfullyear();
    Let mm = today.getmonth() + 1; // Months start at 0!
    Let dd = today.getdate();

    If (dd < 10) dd = "0" + dd;
    If (mm < 10) mm = "0" + mm;

    Return yyyy + "-" + mm + "-" + dd;
};

Const calculateage = (birthdate) => {
    Var today = new Date();
    Var birthdate = new Date(birthdate);
    Var age = today.getfullyear() - birthdate.getfullyear();
    Var m = today.getmonth() - birthdate.getmonth();
    If (m < 0 || (m === 0 && today.getdate() < birthdate.getdate())) {
        Age--;
    }
    Return age;
};

Const modalpopup = ({ show, handleclose, message }) => {
    Return (
        <Modal show={show} onhide={handleclose}>
            <Modal.Header closebutton>

```

```
        <Modal.Title>Informasi Mahasiswa</Modal.Title>
      </Modal.Header>
      <Modal.Body>{message}</Modal.Body>
    </Modal>
  );
};
```