

Booking App

npx create-react-app booking-app

git clone --single-branch -b "react-mini" http://github...

git clone --single-branch -b "react-mini" http://github...

npm install

npm start

npm i react-router-dom

import { BrowserRouter, Routes, Route } from "react-router-dom"

App.js

<BrowserRouter>

<Routes>

<Route path="/" element={<Home />}/>

<Route path="/hotels" element={<List />}/>

<Route path="/hotels/:id" element={<Hotel />}/>

</Routes>

</BrowserRouter>

Navbar.jsx

div.navbar

div.navContainer

Navbar.css

```
.navbar{  
height: 50px;  
display: flex;  
justify-content: center;  
}
```

```
.navContainer{  
width: 100%;  
max-width: 1024px;
```

* ButtonList
yona [margin-right: 20px]; border: 1px solid black; border-radius: 10px; padding: 5px 10px; background-color: #f0f0f0; color: black; font-weight: bold; font-size: 14px; margin-bottom: 10px;

* Buttons

Header.jsx

div.header

div.headerList

 div.headerListItem

Font Awesome Doc

npm install fontawesome core package

import { } files

margin-right: 20px; border: 1px solid black; border-radius: 10px; padding: 5px 10px; background-color: #f0f0f0; color: black; font-weight: bold; font-size: 14px; margin-bottom: 10px;

padding: 5px 10px; border: 1px solid black; border-radius: 10px; background-color: #f0f0f0; color: black; font-weight: bold; font-size: 14px; margin-bottom: 10px;

Home.jsx

{div}

{Header/}

{Header/}

{div id="homeContainer"}

{div}

{mailList/}

{Footer/}

{div}

<FontAwesomeIcon icon={faCircleArrowRight}/>

<FontAwesomeIcon icon={faBed}/>

header.css

```
.header {  
    display: flex;  
    justify-content: center;  
}
```

```
.headerContainer {
```

```
width: 100%;
```

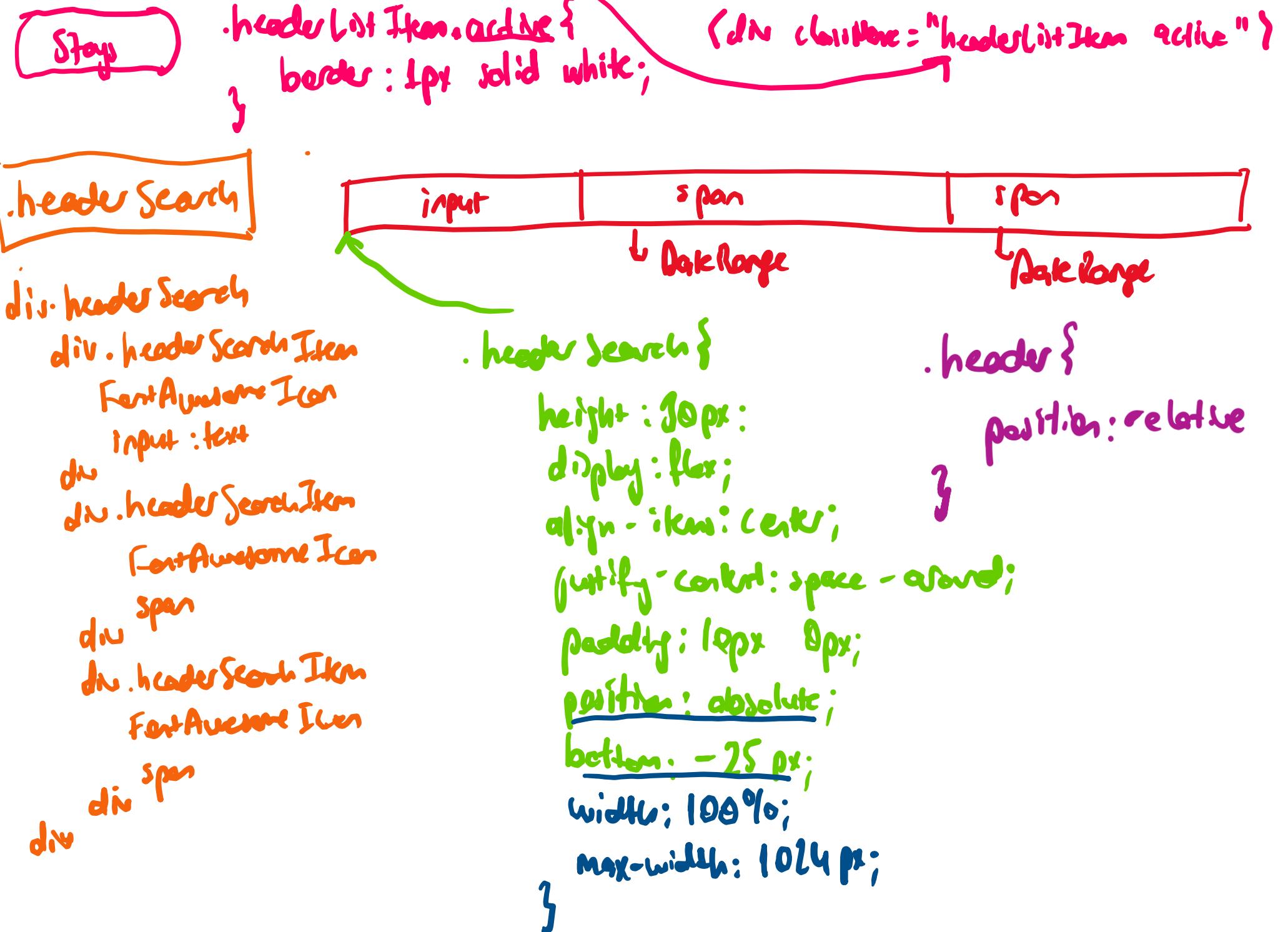
```
max-width: 1000px;
```

```
margin: 20px 0px 100px 0px; → Aşağıda 100px boşluk var.
```

```
}
```

Arama çubuğu var. Arama Çubuğu
position: absolute; ile konumlandırıluy.
bottom: -25px

* Aynı div içindeki elementler arası boşluk **gap** ile verili.
gap: 20px;



```
span.headerSearchInput → color: lightgray;  
< FontAwesome className="headerIcon" /> .headerIcon { color: lightgray; }
```

Date Range

npm install react-date-range

npm install date-fns → format functions

import CJS file for date range

```
import { DateRange } from "react-date-range";
```

```
import { format } from "date-fns";
```

```
const [date, setDate] = useState([
```

```
    { startDate: new Date(),
```

```
    endDate: new Date(),
```

```
    key: "selection"
```

```
});
```

```
{openAll < DateRange onChange={item => setDate([item.selection])}  
    ranges={date} className="date"  
    maxRange={intSelection = false}  
/}>
```

```
.date {
    position: absolute;
    top: 50px;
} onClick = { () => setOpenDate(!openDate) }
```

```
<span> {` ${format(date[0].startDate, "MM/dd/yyyy")}` to
         ${format(date[0].endDate, "MM/dd/yyyy")}``}
```

```
</span>
```

```
const [openDate, setOpenDate] = useState(false)
```

```
0 - 0 - 00 - 00 - 0 - 0 - 0
```

```
const [openOption, setOpenOption] = useState(false)
```

```
const [options, setOptions] = useState({
    adult: 1,
    children: 0,
    room: 1
})
```

```
<span onClick = { () => setOpenOption(!openOption) }> {options.adult} adult +
    {options.children} children +
    {options.room} room
```

```
</span>
```

```
optionCounterButton : disabled {  
    cursor: not-allowed;  
}
```

div.options

div.optionItem

span.optionText

disabled = {options.adult <= 1}

button.optionCounterButton - onClick = {() => handleOpen("adult", "d")}

span {options.adult}

button.optionCounterButton + onClick = {() => handleOpen("adult", "i")}

div

div.optionItem

disabled = {options.children <= 0}

button.optionCounterButton - onClick = {() => handleOpen("children", "c")}

span {options.children}

button.optionCounterButton + onClick = {() => handleOpen("children", "i")}

div

div.optionItem

disabled = {options.room <= 1}

button.optionCounterButton - onClick = {() => handleOpen("room", "d")}

span {options.room}

button.optionCounterButton + onClick = {() => handleOpen("room", "i")}

div

div

```
.options {
    position: absolute;
}
#kp:50px;
.optionsIcon {
    width: 200px;
    display: flex;
    justify-content: space-between;
}
.optionsCounterButton {
    width: 30px;
    height: 30px;
}
```

```
const handleOption = (name, operation) => {
    setOptions((prev) => {
        return {
            ...prev,
            [name]: operation === "+" ? options : options[name] + 1
        }
    })
}
```

```
const handleOption = (name, operation) => {
    setOptions((prev) => {
        return {
            ...prev,
            [name]: operation === "i" ? options[name] + 1 : options[name] - 1
        }
    })
}

const handleOption = (name, operation) => {
    setOptions((prev) => {
        return {
            ...prev,
            [name]: operation === "i" ? options[name] + 1 : options[name] - 1
        }
    })
}

const handleOption = (name, operation) => {
    setOptions((prev) => {
        return {
            ...prev,
            [name]: operation === "i" ? options[name] + 1 : options[name] + 1
        }
    })
}
```

```
const handleOption = (name, operation) => {
    setOptions((prev) => {
        return {
            ...prev,
            [name]: operation === "i" ? option[name] + 1 : option[name] - 1
        }
    })
}
```

```
const handleOption = (name, operation) => {
    setOptions((prev) => {
        return {
            ...prev,
            [name]: operation === "i" ? option[name] + 1 : option[name] - 1
        }
    })
}
```

```
const handleOption = (name, operation) => {
    setOptions((prev) => {
        return {
            ...prev,
            [name]: operation === "i" ? option[name] + 1 : option[name] - 1
        }
    })
}
```

```
const handleOption = (name, operation) => {
    setOption((prev) => {
        return {
            ...prev,
            [name]: operation === ":" ? options[name] + 1 : options[name] - 1
        }
    })
}

const handleOption = (name, operation) => {
    setOption((prev) => {
        return {
            ...prev,
            [name]: operation === ":" ? options[name] + 1 : options[name] - 1
        }
    })
}

const handleOption = (name, operation) => {
    setOption((prev) => {
        return {
            ...prev,
            [name]: operation === ":" ? options[name] + 1 : options[name] - 1
        }
    })
}
```

Header.js

```
(const Header = ({type}) => {  
  <div>...  
  <div>  
    {type!=="list" ? <h1>...  
    :  
  }  
})
```

Navbar.js

```
<Header type="list">  
  .homeContainer
```

h1: homeTitle

Home.js

```
<Navbar />  
<Header />  
div.homeContainer  
  <Featured />  
  <h1.homeTitle />  
<PropertyList />
```

home.css

```
.homeContainer{  
  margin-top: 30px;  
  display: flex;  
  flex-direction: column;  
  align-items: center;  
  gap: 30px  
}  
  
h1  
.homeTitle {  
  width: 100px  
}
```

```
üg .featured {  
    width: 100%;  
    max-width: 1000px;  
    display: flex;  
    justify-content: space-between;  
    gap: 20px;  
    z-index: 1  
}  
  
.featuredImg {  
    width: 100%;  
    object-fit: cover;  
}  
}  
  
1: image in terms of intent is in  
  
.date {  
    z-index: 2;  
}  
}  
  
.options {  
    z-index: 2;  
}  
}
```

```
.pList{  
    width: 100%;  
    max-width: 1024px;  
    display: flex;  
    justify-content: space-between;  
    gap: 20px;  
}  
.listIcon{  
    overflow: hidden;  
    flex: 1;  
}  
.listImage{  
    width: 100%;  
    height: 250px;  
    object-fit: cover;  
}
```

```
sdsf"rf11fv5  
.mail InputContainer > input{  
    width: 30px;  
    height: 30px;  
    padding: 10px;  
    border: none;  
    margin-right: 10px;  
}
```

list.jsx

```
<div>  
  <Navbar />  
  <Header type="list" />  
  <div className="listContainer">  
    <div className="listWrapper">  
      <div className="listHeader">  
        <div>  
          <div>  
            <div>
```

```
:listContainer{  
    display:flex;  
    justify-content:center;  
    margin-top:20px;  
}  
  
.listWrapper{  
    width:100%;  
    max-width:1024px;  
    display:flex;  
    gap:20px;  
}  
  
.listSearch {flex:1;}  
.listResult {flex:3;}  
  
.listSearch {  
    flex:1;  
    position:sticky;  
    top:10px
```

Header.jsx

```
<div className="headerSearch">
  <button className="headerSearchBtn" onClick={()=>handleSearch}>
    const navigate = useNavigate()
    const handleSearch = ()=>{
      navigate ("/hotels", { state: { destination, date, option } })
    }
    const [destination, setDestination] = useState ("")
    const [date, setDate] = useState ({
      startDate: new Date(),
      endDate: new Date(),
      key: selection
    })
    const [option, setOption] = useState ({
      adult: 1,
      children: 0,
      room: 1
    })
  </button>
</div>
```

```
<input type="text" onChange={e=> setDestination(e.target.value)} />
```

List.jsx

```
const location = useLocation()
const [destination, setDestination] = useState(location.state.destination)
const [date, setDate] = useState(location.state.date)
const [options, setOptions] = useState(location.state.options)

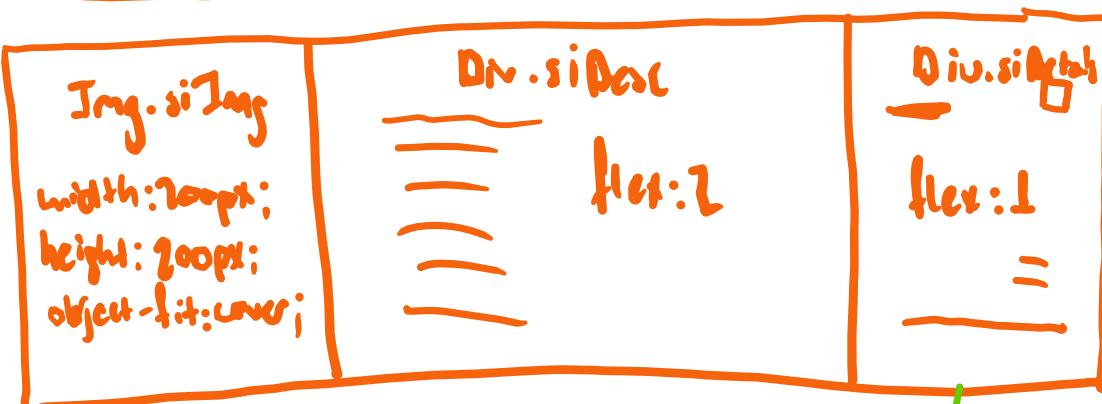
<span onClick={()=> setDate(!openDate)}>
  ${` ${format(date[0].startDate, "MM/dd/yyyy")} to ${format(date[0].endDate, "MM/dd/yyyy")}`}
</span>

<input type="text" placeholder={destination}/>
.liItem > span {
  display: flex;
  align-items: center;
}
```

```
{ openBook dk  
<DatePicker onChange={ (item) => setDate([item.selection]) } minDate={ new Date() } range={ date } />
```

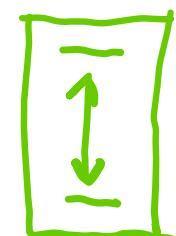
```
•  
<input type="number" min={ 1 } placeholder={ options.adult } />  
<input type="number" min={ 0 } placeholder={ options.children } />  
<input type="number" min={ 1 } placeholder={ options.room } />  
.list Search button{  
    width:100%;  
}
```

Search Item.js



display:flex;
justify-content:space-between
flex-direction:column;

free airport taxi:
width: max-content;
↓
free airport taxi width: max-content;



Hotel.jsx

```
(div)
  (div)
    (Header type="list")
    (div className="hotelContainer")
      (div className="hotelWrapper")
        :
        (div className="hotelImages")
          {photos.map((photo,i)=>{
            (div className="hotelImageWrapper" key={i})
              (img src={photo.src} className="hotelImg")
            })
          }
        .
        (div)
        (div className="hotelDetails")
          (div className="hotelDetailsTexts") (div)
          (div className="hotelDetailsPrice") (div)
        .
        (div)
        (div)
        (div)
      
```

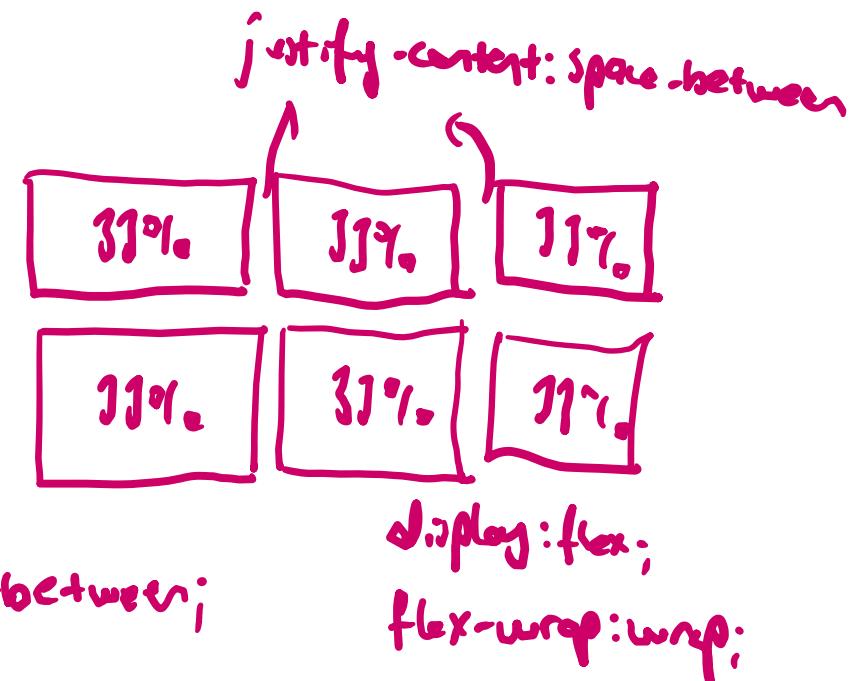
const photos = [
 { src: "....."},
 { src: "..."},
 :
];

Maillat Footer

hotel.css

```
.hotelContainer {  
    display: flex;  
    justify-content: center;  
}  
margin-top: 20px;  
  
.hotelWrapper {  
    width: 100%;  
    max-width: 1000px;  
    display: flex;  
    flex-direction: column;  
    gap: 10px;  
}  
  
.hotelImageWrapper {  
    width: 33%;  
}  
  
.hotelImg {  
    width: 100%;  
    object-fit: cover;  
}
```

```
.hotelImages {  
    display: flex;  
    flex-wrap: wrap;  
    justify-content: space-between;  
}
```



```
.hotelDetails {  
    display: flex;  
    justify-content: space-between;  
    gap: 20px;  
    margin-top: 20px;  
}
```

```
.hotelDetailsText {  
    flex: 1;  
}  
.hotelDetailsPrice {  
    flex: 1;  
}
```

SLIDER

```
const [slideNumber, setSlideNumber] = useState(0)  
<div className="hotelImage">  
    {photos.map((photo, i) => (

<img onClick={() => setSlideNumber(i)} className="hotelImg" src={photo.src}/>

)})  
</div>
```

```
const [open, setOpen] = useState(false)
```

```
const handleOpen = (i) => {
```

```
  setSlideNumber(i)
```

```
  setOpen(true)
```

```
<img onClick={()=>handleOpen(i)} className="holderImg" />
```

```
const handleMove = (direction) => {
```

```
  let newSlideNumber;
```

```
  if (direction === "l") {
```

```
    newSlideNumber = slideNumber == 0 ? 5 : slideNumber - 1
```

```
  } else {
```

```
    newSlideNumber = slideNumber == 5 ? 0 : slideNumber + 1
```

```
  }
```

```
  setSlideNumber(newSlideNumber)
```

```
}
```

```
const handleMove = (direction) => {
```

```
  let newSlideNumber;
```

```
  if (direction === "l") {
```

```
    newSlideNumber = slideNumber == 0 ? 5 : slideNumber - 1;
```

```
  } else {
```

```
    newSlideNumber = slideNumber == 5 ? 0 : slideNumber + 1
```

```
  }
```

```
  setSlideNumber(newSlideNumber);
```

```
}
```

```
const handleMove = (direction) => {
```

```
    let newSlideNumber;
```

```
    if (direction == "l") {
```

```
        newSlideNumber = slideNumber == 0 ? 5 : slideNumber - 1;
```

```
    } else {
```

```
        newSlideNumber = slideNumber == 5 ? 0 : slideNumber + 1
```

```
}
```

```
    setSlideNumber(newSlideNumber)
```

```
}
```

```
const handleMove = (direction) => {
```

```
    let newSlideNumber;
```

```
    if (direction == "l") {
```

```
        newSlideNumber = slideNumber == 0 ? 5 : slideNumber - 1;
```

```
    } else {
```

```
        newSlideNumber = slideNumber == 5 ? 0 : slideNumber + 1;
```

```
}
```

```
    setSlideNumber(newSlideNumber)
```

```
}
```

```
{open drd()
<div className="slider">
  <FontAwesomeIcon icon={faCircleXmark} className="close" onClick={()=>setOpen(false)} />
  <FontAwesomeIcon icon={faCircleArrowLeft} className="arrow" onClick={()=>handleOpen("l")}/>
  <div className="sliderWrapper">
    <img src={photos[slideNumber].src} className="sliderImg"/>
    </div>
    <FontAwesomeIcon icon={faCircleArrowRight} className="arrow" onClick={()=>handleOpen("r")}/>
  </div>
}
{open drd()
<div className="slider">
  <FontAwesomeIcon icon={faCircleXmark} className="close" onClick={()=>setOpen(false)} />
  <FontAwesomeIcon icon={faCircleArrowLeft} className="arrow" onClick={()=>handleMore("l")}/>
  <div className="sliderWrapper">
    <img src={photos[slideNumber].src} className="sliderImg"/>
    </div>
    <FontAwesomeIcon icon={faCircleArrowRight} className="arrow" onClick={()=>handleMore("r")}/>
  </div>
}
}
```

```
.slider{  
    position: sticky;  
    top: 0;  
    left: 0;  
    width: 100vw;  
    height: 100vh;  
    z-index: 999;  
    background-color: rgba(0,0,0,0.426);  
    display: flex;  
    align-items: center;  
}  
.sliderWrapper{  
    width: 100%;  
    height: 100%;  
    display: flex;  
    justify-content: center;  
    align-items: center;  
}  
.sliderImg{  
    width: 80%;  
    height: 80vh;  
}
```

```
.slider{  
    position: sticky;  
    top: 0;  
    left: 0;  
    width: 100vw;  
    height: 100vh;  
    z-index: 999;  
    background-color: rgba(0,0,0,0.426);  
    display: flex;  
    align-items: center;  
}  
.sliderWrapper{  
    width: 100%;  
    height: 100%;  
    display: flex;  
    justify-content: center;  
    align-items: center;  
}  
.sliderImg{  
    width: 80%;  
    height: 80vh;  
}
```

```
.close {  
    position: absolute;  
    top: 20px;  
    left: 20px;  
    font-size: 30px;  
    color: lightgray;  
}
```

```
.arrow {  
    margin: 10px;  
    font-size: 50px;  
    color: lightgray;  
}
```

```
const handleMove = (direction) => {  
    let newSlideNumber;  
    if (direction === "l") {  
        newSlideNumber = slideNumber === 0 ? 5 : slideNumber - 1;  
    } else {  
        newSlideNumber = slideNumber === 5 ? 0 : slideNumber + 1;  
    }  
    setSlideNumber(newSlideNumber)  
}
```

