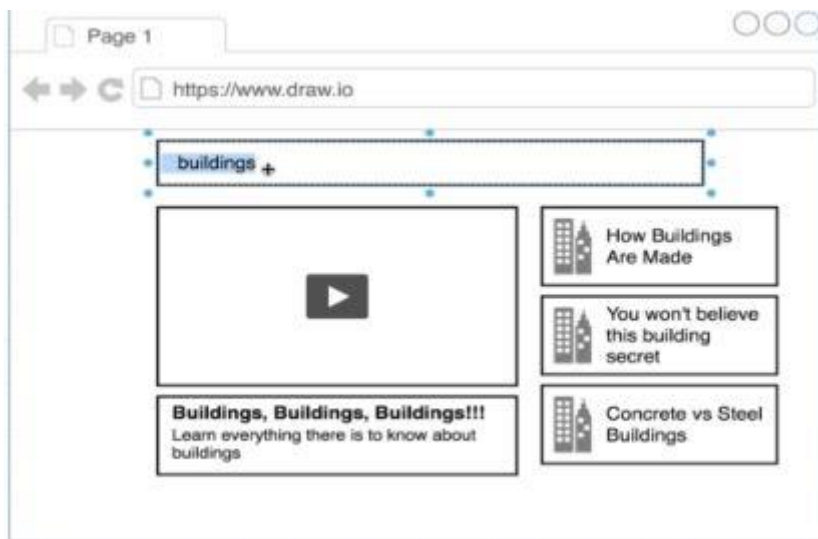
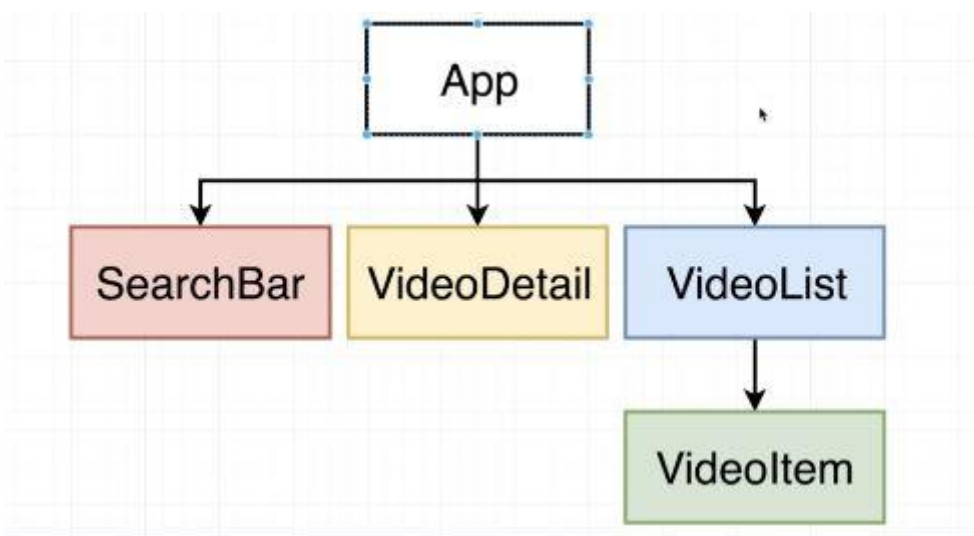


Proje de Yapılmak İstenilen



Proje Component Tasarımı



İlk Başta

“npx create-react-app videos” videos adında bir react projesi oluştururuz.

Sonra src klasör içindeki dosyaları silip boş bir projede sıfırdan projemizi yapmak için.

App Class Componenti oluşturup bunu DOM’da render etmek için

```
import React from 'react'

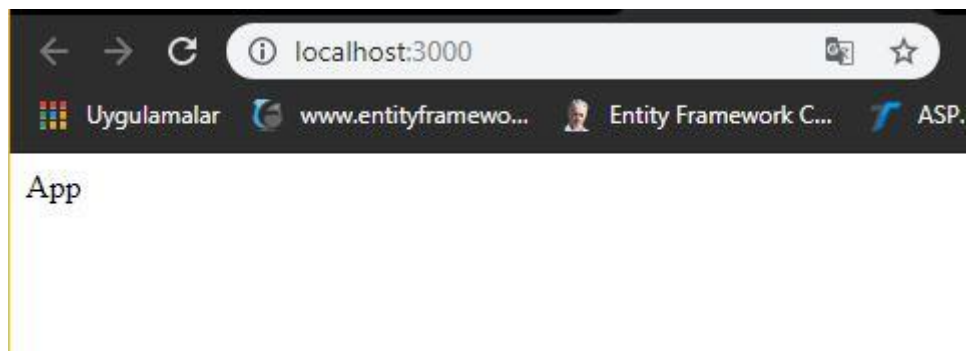
class App extends React.Component{
  render(){
    return (
      <div>
        App
      </div>
    )
  }
}

export default App
```

```
import React from 'react'
import ReactDOM from 'react-dom'
import App from './components/App'

ReactDOM.render(<App/>,document.querySelector('#root'))
```

Sonuc:



&&&

&&&

&&&

&&&

&&&

&&&

Şimdi SearchBar Class Componenti oluşturun, App Component'e çağırma. Yani Bir componenti başka componente nasıl çağırılır.

SearchBar.js çağırılacak component

```
import React from 'react'

class SearchBar extends React.Component{
  render(){
    return(
      <div>
        SearchBar
      </div>
    )
  }
}

export default SearchBar
```

App.js Çağırılacak yer olan component

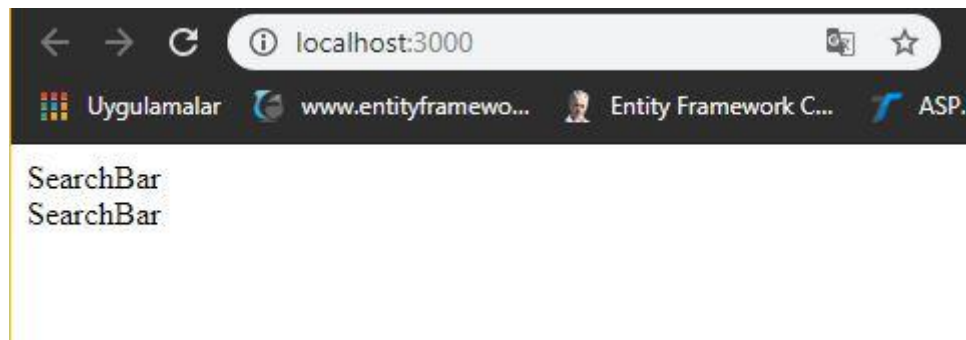
```
import React from 'react'
import SearchBar from './SearchBar'

class App extends React.Component{
  render(){
    return (
      <div>
        <SearchBar/>
        <SearchBar></SearchBar>
      </div>
    )
  }
}

export default App
```

App'de zaten DOM'a render ediliyordu.

Sonuc:



Projeye css dosyası ekleme

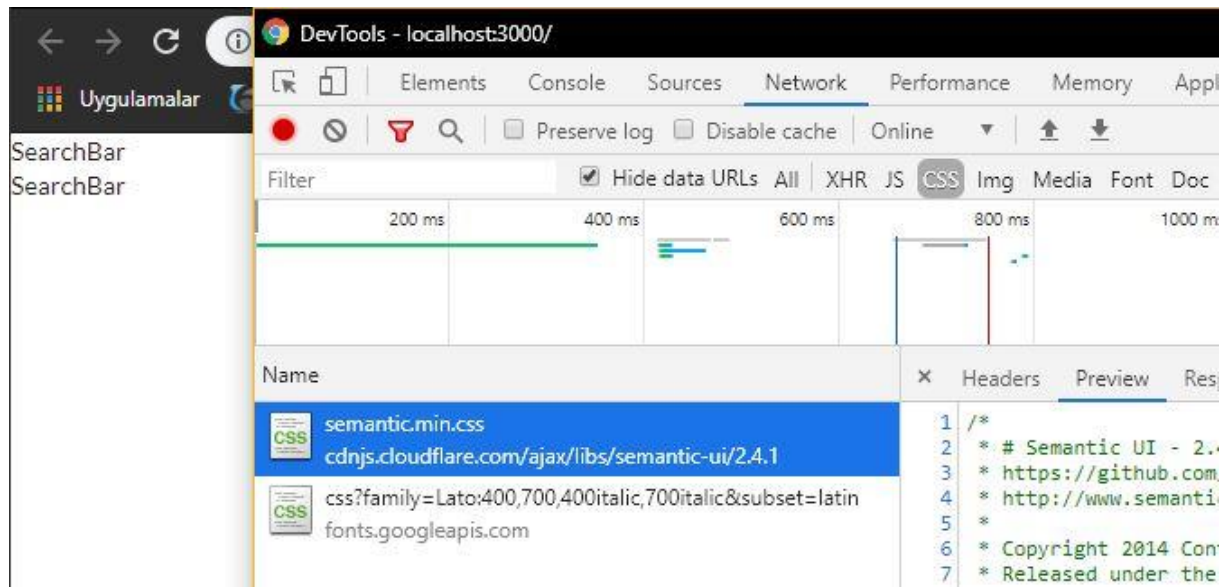
<https://cdnjs.com/libraries/semantic-ui> sitesinden

<https://cdnjs.cloudflare.com/ajax/libs/semantic-ui/2.4.1/semantic.min.css>

linki kopyalayıp aşağıdaki gibi projeye eklemeliyiz.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/semantic-
ui/2.4.1/semantic.min.css" />
    <title>React App</title>
  </head>
```

Sonuc:



&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

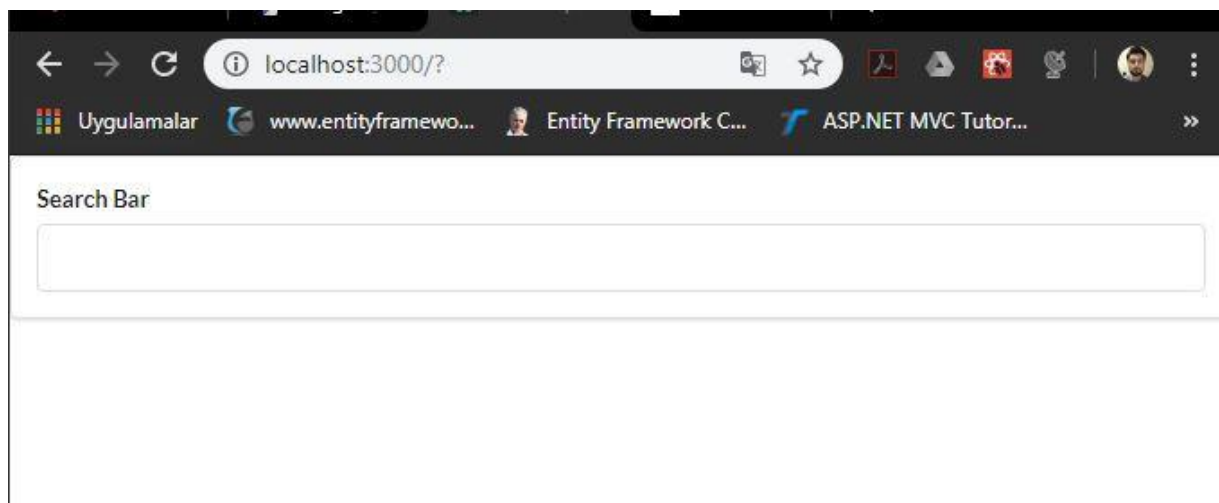
Şimdi SearchBar Componenti Css ile biraz değiştirelim ve bir input ekleyelim.

```
import React from 'react'

class SearchBar extends React.Component{
  render(){
    return(
      <div className="search-bar ui segment">
        <form className="ui form">
          <div className="field">
            <label>Search Bar</label>
            <input type="text"/>
          </div>
        </form>
      </div>
    )
  }
}

export default SearchBar
```

Sonuc:



&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

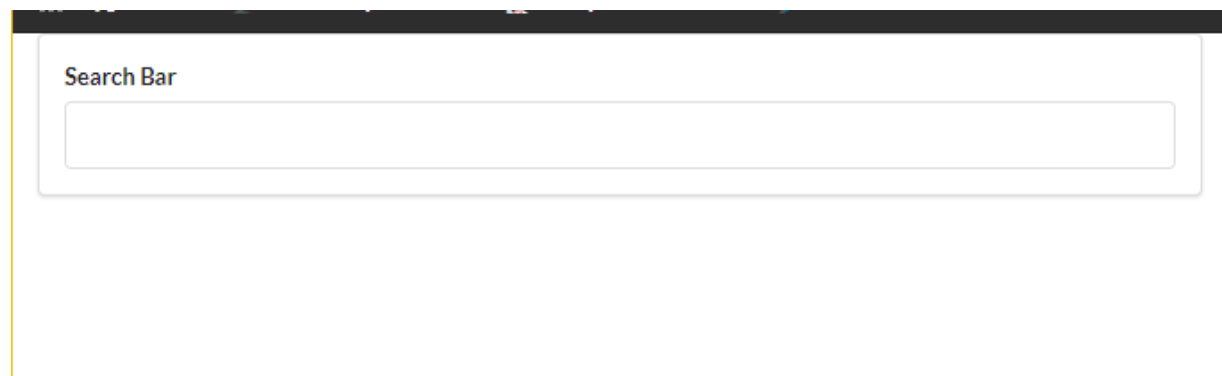
App componentin tamamını kapladığı görülür bunu biraz güzelleştirmek için

```
import React from 'react'
import SearchBar from './SearchBar'

class App extends React.Component{
  render(){
    return (
      <div className="ui container">
        <SearchBar/>
      </div>
    )
  }
}

export default App
```

Sonuc:



&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

Şimdi SearchBar da yazılan girdileri bir yerde tutmamız gerekir çünkü girilen değer ile birlikte api'de arama yapacağımızdan.

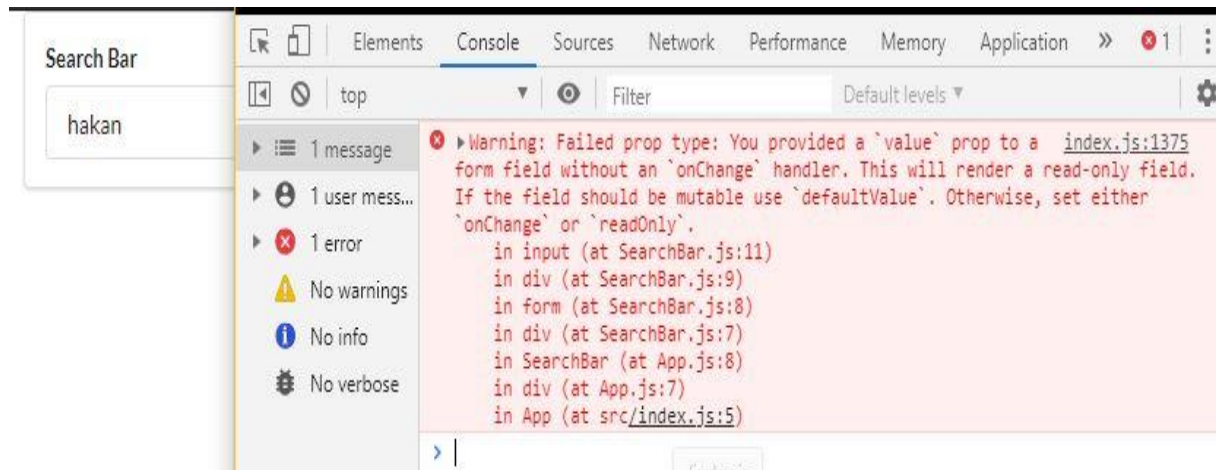
Kullanıcıdan girdi alınarak değeri tutma.

```
import React from 'react'

class SearchBar extends React.Component{
  state={term:'hakan'}
  render(){
    return(
      <div className="search-bar ui segment">
        <form className="ui form">
          <div className="field">
            <label>Search Bar</label>
            <input type="text" value={this.state.term}/>
          </div>
        </form>
      </div>
    )
  }
}

export default SearchBar
```

Sonuc: Böyle yaparsak value değeri yani inputa bir değer yazamamış oluruz bunun için onChange() metodu kullanmalıyız.



&&&

&&&

&&&

&&&

&&&

&&&

OnChange ile Input'a yazılanı değiştirme ve state de tutulan kullanıcı girdisini değiştirme.

```
import React from 'react'

class SearchBar extends React.Component{
  state={term:''} //state tanımı

  onInputChange = (event) => {
    this.setState({term:event.target.value}) //state ile değiştirme
  }

  render(){
    return(
      <div className="search-bar ui segment">
        <form className="ui form">
          <div className="field">
            <label>Search Bar</label>
            <input type="text" value= {this.state.term} //çağırma
            onChange= {this.onInputChange} />
          </div>
        </form>
      </div>
    )
  }
}

export default SearchBar
```

Sonuc: SearchBar da değişmeyen girdi şimdi değişebilir duruma geldi ve yazılan girdi bir değişkende state ile tutuluyor.

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

Şimdi işe Form'u submit etmeye geldi girilen değerleri submit etme.

```
import React from 'react'

class SearchBar extends React.Component{
  state={term:''}

  onInputChange = (event) => {
    this.setState({term:event.target.value})
  }

  onFormSubmit = event => {
    event.preventDefault() //Enter'a basınca sayfa yenilenmez
    //TODO:Make sure we call
    //callback from parent component
  }

  render(){
    return(
      <div className="search-bar ui segment">
        <form onSubmit= {this.onFormSubmit} className="ui form">
          <div className="field">
            <label>Search Bar</label>
            <input type="text" value={this.state.term}
onChange={this.onInputChange}/>
          </div>
        </form>
      </div>
    )
  }
}

export default SearchBar
```

Sonuc: Submit işlemi Api kurulduktan sonra devam edilecek. Önemli nokta Form dolduruldu enter yapınca sayfanın yenilenmemesi için preventDefault() eklendi.

&&&

&&&

&&&

&&&

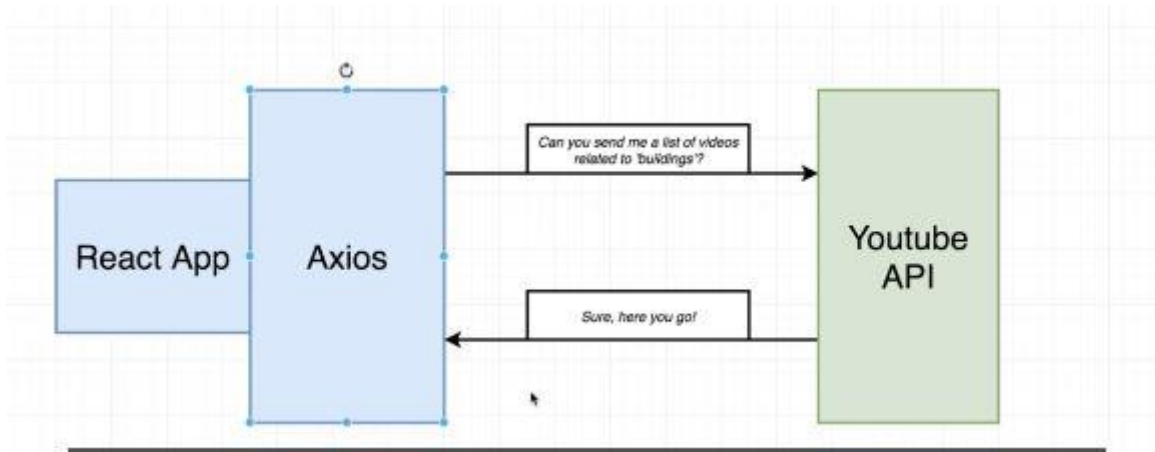
&&&

&&&

&&&

&&&

Api'den veri çekme mantığı



console.developers.google.com

<https://console.developers.google.com/apis/dashboard?project=my-project-1513557951684&angularJsUrl=>

Sol üstte : My Project + Yeni Proje + Project Name: Video Browser

Sağ üstte bildirimler de projenin oluşturulduğu görülür.

Aramda : YouTube Data API v3 seçeriz. Etkinleştiririz.

Sonra kimlik bilgilerini oluşturmalıyız.

Hangi Api kull => Youtube Data API v3

Api'ye nerden çağıracaksınız => Web tarayıcısı (javascript)

Hangi verileri işleyeceksiniz => Herkese açık

Sonra Hangi kimlik bilg ihtiyac var tıklıyoruz.

Sonra kimlik bilgileri alırsak => AIzaSyBBSYkulYKr_P6dr3Il7rSi-rPBapq4BDE

Sonra Anahtarı kısıtlayın, uyg kısıtlaması http yönlendirenler seç ve localhost:3000 ögesi eklenir.

&&&

&&&

&&&

&&&

&&&

&&&

&&&&

&&&

&&&

Src + apis + youtube.js

```
const key = 'AIzaSyBBSYkulYKr_P6dr3Il7rSi-rPBapq4BDE'
```

```
npm install axios@0.18.1
```

```
npm install - - save axios
```

yüklenir.

<https://developers.google.com/youtube/v3/docs/search/list>

adresine gidilir.

HTTP request => GET <https://www.googleapis.com/youtube/v3/search>

```
import Axios from 'axios';

const Key = 'AIzaSyBBSYkulYKr_P6dr3Il7rSi-rPBapq4BDE'

export default Axios.create({
  baseURL: 'https://www.googleapis.com/youtube/v3/',
  params: {
    part: 'snippet',
    maxResults: 5,
    key: Key
  }
})
```

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

Şimdi App Component'e çağırılan SearchBar Component'e Attr ekleyerek props ile App'dan veriyi SearchBar component'e taşıcaz. Çünkü App'da Api den aldığımız verileri yollıcaz.

App.js

```
import React from 'react'
import SearchBar from './SearchBar'
import youtube from '../apis/youtube'
class App extends React.Component{
  onTermSubmit = (term) => {
    console.log(term) //search barda yazılan değerler gözüktür.
    youtube.get('/search',{
      params:{
        q:term //listede o değeri aramak için
      }
    })
  }
  render(){
    return (
      <div className="ui container">
        <SearchBar onFormSubmit={this.onTermSubmit}/>
      </div>
    )
  }
}
export default App
```

SearchBar.js

```
import React from 'react'
class SearchBar extends React.Component{
  state={term:''}
  onInputChange = (event) => {
    this.setState({term:event.target.value})
  }
  onFormSubmit = event => {
    event.preventDefault()
    this.props.onFormSubmit(this.state.term) //OnFormSubmit'e term gider.
  }
  render(){
    return(
      <div className="search-bar ui segment">
        <form onSubmit= {this.onFormSubmit} className="ui form">
          <div className="field">
            <label>Search Bar</label>
            <input type="text" value={this.state.term}
onChange={this.onInputChange}/>
          </div>
        </form>
      </div>
    )
  }
}
```

```
}  
}  
  
export default SearchBar
```

SearchBar.js'de

```
this.props.onSubmit()
```

inputta yazılan veri

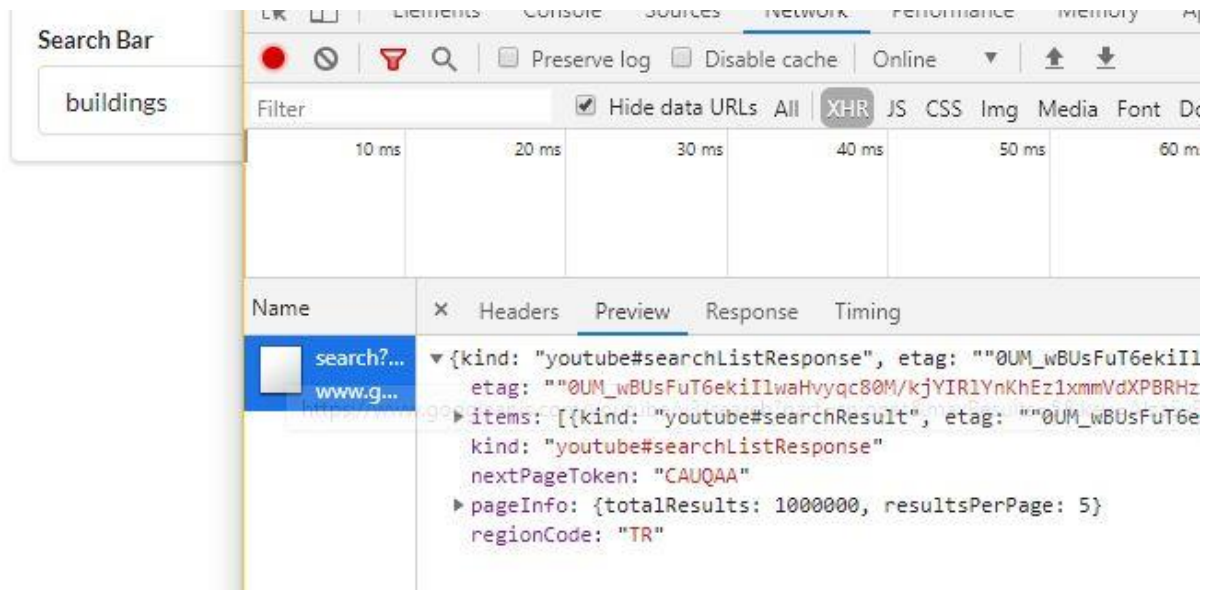
```
console.log(term)
```

de undefined iken

```
this.props.onSubmit(this.state.term)
```

böyle yaparsak da inputa ne yazılırsa o gelir.

Props ile fonksiyonu taşıma yukardaki örnek.



&&&

&&&

&&&

&&&

&&&

&&&

Aranılan değer Api den geri dönen Listeyi bir değişkende tutmak ve Asenkron bir işlem olduğundan Async-await kullandık.

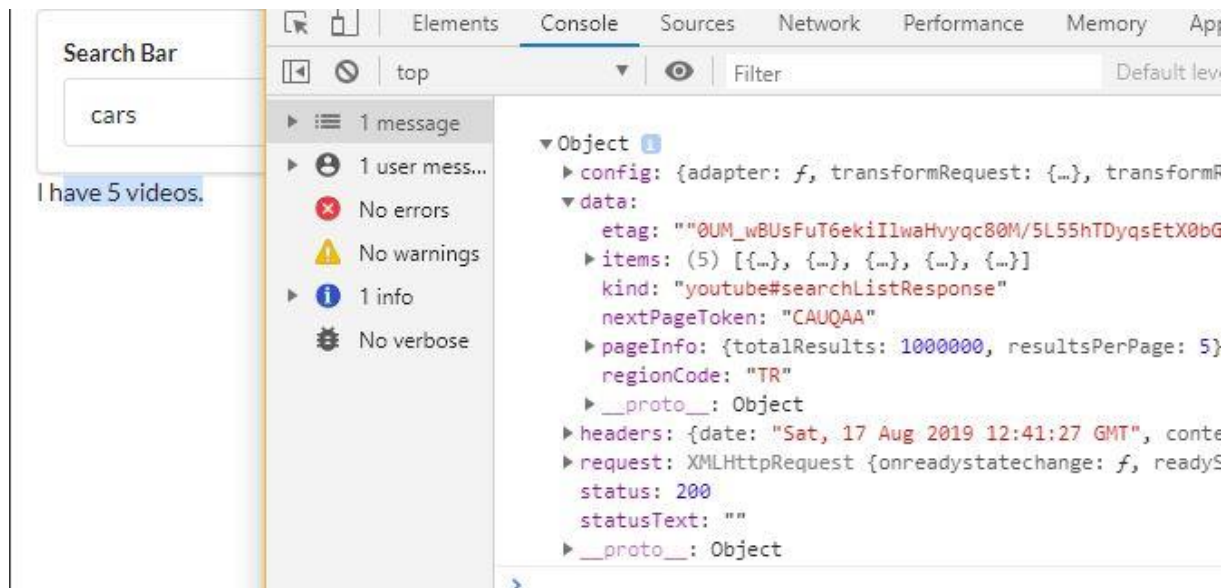
```
import React from 'react'
import SearchBar from './SearchBar'
import youtube from '../apis/youtube'

class App extends React.Component{
  state = {videos:[]}
  onTermSubmit = async (term) => {
    const response = await youtube.get('/search',{
      params:{
        q:term
      }
    })
    console.log(response)
    this.setState({videos:response.data.items})
  }

  render(){
    return (
      <div className="ui container">
        <SearchBar onFormSubmit={this.onTermSubmit}/>
        I have {this.state.videos.length} videos.
      </div>
    )
  }
}

export default App
```

console.log(response) çıktısı aşağıdaki gibidir.



The screenshot shows a web application interface on the left and a browser developer console on the right. In the web application, there is a search bar with the text 'cars' and a message below it that says 'I have 5 videos.' The browser developer console has tabs for Elements, Console, Sources, Network, Performance, Memory, and Application. The Console tab is active, showing a log message for the API response. The log message is an object with the following structure:

```
Object {
  config: {
    adapter: f,
    transformRequest: {...},
    transformResponse: {...},
  },
  data: {
    etag: '"0UM_wBU5FuT6ekiIlwaHvyqc80M/5L55hTDyqsEtX0bG"',
    items: (5) [{...}, {...}, {...}, {...}, {...}],
    kind: "youtube#searchListResponse",
    nextPageToken: "CAUQAA",
    pageInfo: {
      totalResults: 1000000,
      resultsPerPage: 5,
    },
    regionCode: "TR",
    __proto__: Object,
  },
  headers: {
    date: "Sat, 17 Aug 2019 12:41:27 GMT",
    content-type: "application/json",
  },
  request: XMLHttpRequest {onreadystatechange: f, readyState: 4, status: 200, statusText: ""},
  __proto__: Object
}
```

Şimdi geldik VideoList Fonk. Component'e

Bir öncekinde aradığımızda bir list elde ettik state de tuttuk bunu props ile VideoList component'e aktarmaya geldi.

```
import React from 'react'
import SearchBar from './SearchBar'
import youtube from '../apis/youtube'
import VideoList from './VideoList'

class App extends React.Component{
  state = {videos:[]}
  onTermSubmit = async (term) => {
    const response = await youtube.get('/search',{
      params:{
        q:term
      }
    })
    console.log(response)
    this.setState({videos:response.data.items})
  }

  render(){
    return (
      <div className="ui container">
        <SearchBar onFormSubmit={this.onTermSubmit}/>
        I have {this.state.videos.length} videos
        <br/>
        <hr/>
        <VideoList videosprops={this.state.videos}/>
      </div>
    )
  }
}
export default App
```

```
import React from 'react'

const VideoList = (props) => {
  return (
    <div>
      Video List {props.videosprops.length}
    </div>
  )
}

export default VideoList
```

Sonuc Listede kac eleman varsa onu döndürür.

Şimdi VideoList'de farklı video itemlar olabilir ve VideoItem(Fonk) componenti yapacağız. VideoList içinde VideoItem olmalı.

```
import React from 'react'

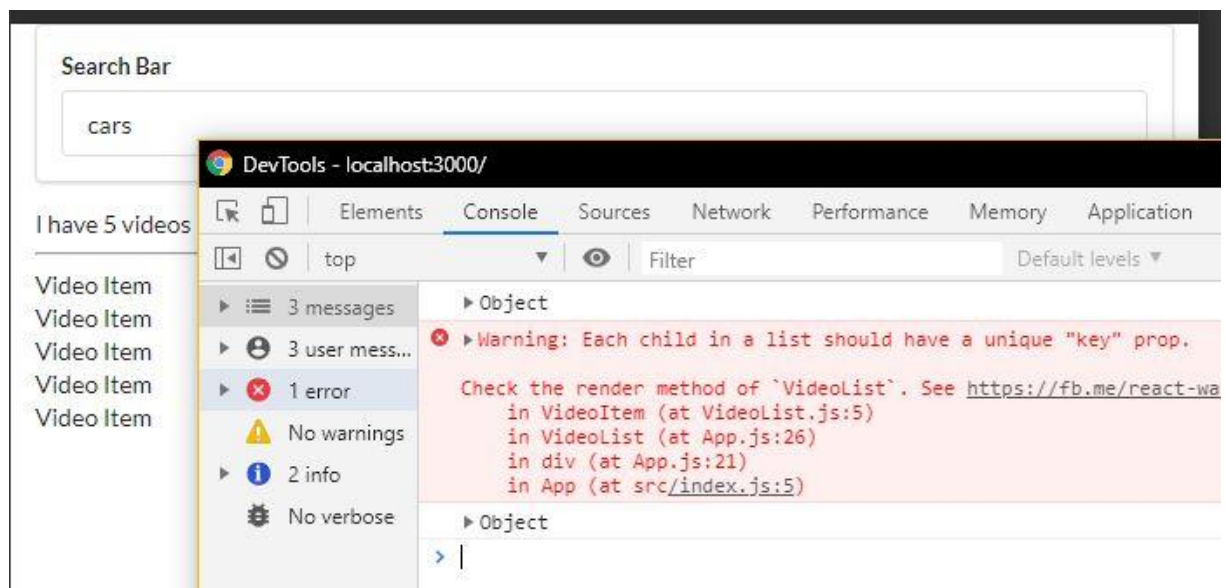
const VideoItem = (props) => {
  return(
    <div>
      Video Item
    </div>
  )
}

export default VideoItem
```

```
import React from 'react'
import VideoItem from './VideoItem'
const VideoList = (props) => {
  const Item = props.videosprops.map((vid)=> {
    return <VideoItem/>
  })
  return (
    <div>
      {Item}
    </div>
  )
}

export default VideoList
```

Listede kaçane var ise map ile dönülür ve componenti çağırırız şimdilik sonuç ve bir hata var



VideoItem'da Listeden title ve image değerlerini çekmeye geldi

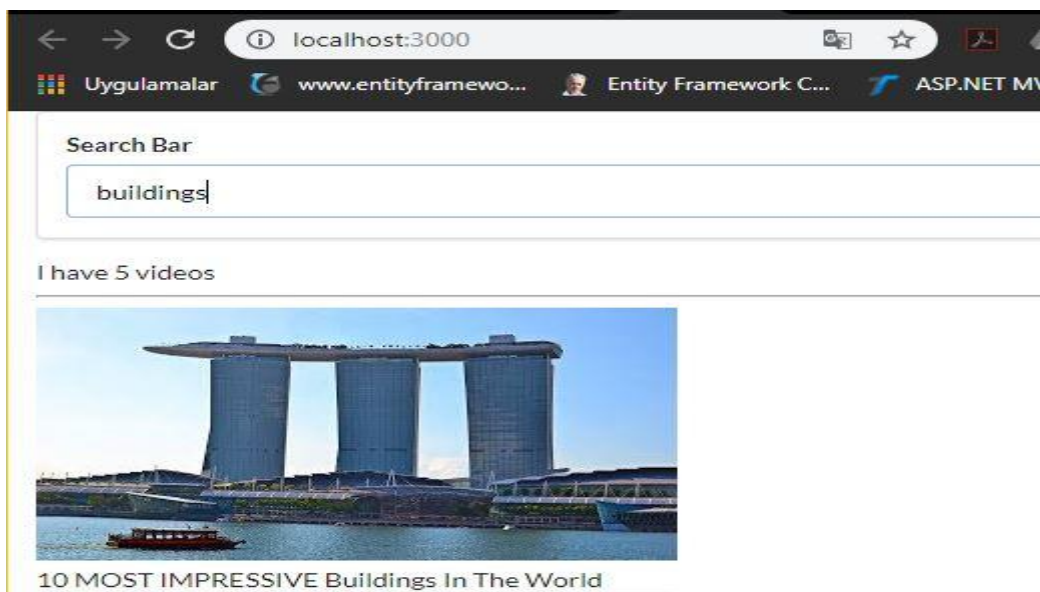
```
import React from 'react'
import VideoItem from './VideoItem'
const VideoList = (props) => {
  const Item = props.videosprops.map((vid)=> {
    return <VideoItem itemprops={vid} />
  })
  return (
    <div>
      {Item}
    </div>
  )
}

export default VideoList
```

```
import React from 'react'

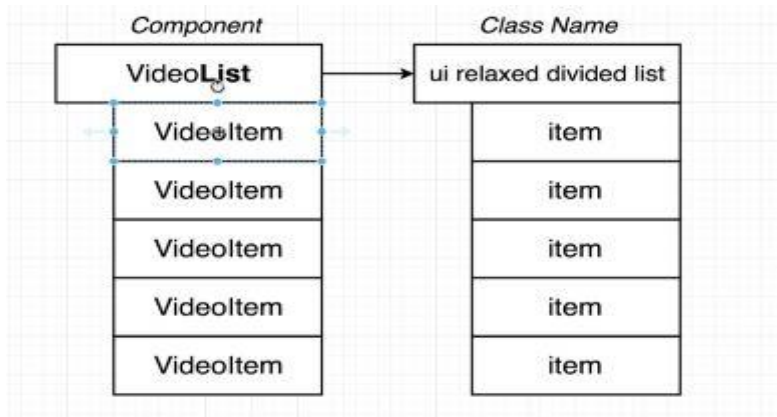
const VideoItem = (props) => {
  return(
    <div>
      <img src={props.itemprops.snippet.thumbnails.medium.url}/>
      <br/>
      {props.itemprops.snippet.title}
    </div>
  )
}

export default VideoItem
```



Verileri VideoList içinde VideoItemları listeledik. Biraz görseli güzelleştirmek için

<https://semantic-ui.com/elements/list.html>



```
import React from 'react'
import VideoItem from './VideoItem'
const VideoList = (props) => {
  const Item = props.videosprops.map((vid)=> {
    return <VideoItem itemprops={vid}/>
  })
  return (
    <div className="ui relaxed divided list" >
      {Item}
    </div>
  )
}

export default VideoList
```

```
import React from 'react'
import './css/VideoItem.css'
const VideoItem = (props) => {
  return(
    <div className="video-item item">
      <img className="ui image"
src={props.itemprops.snippet.thumbnails.medium.url}/>
      <div className="content">
        <div className="header">
          {props.itemprops.snippet.title}
        </div>
      </div>
    </div>
  )
}

export default VideoItem
```

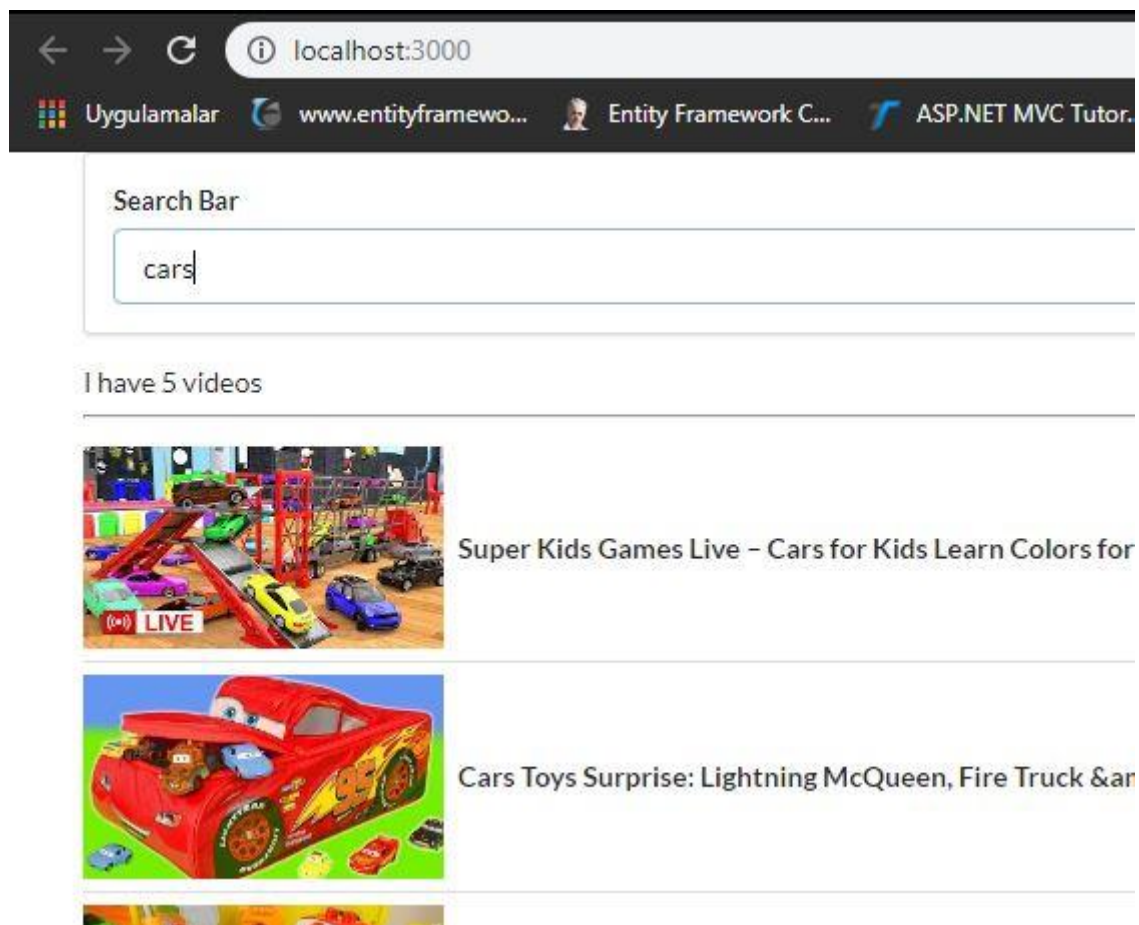
```

.video-item{
    display:flex !important;
    align-items:center !important;
    cursor:pointer;
}

.video-item.item img{
    max-width:180px;
}

```

Sonuc :



&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

Amaç



Item dan birine tıklayınca ortada büyük şekilde gösterilmesini sağlamak.

Cursor:pointer

Şimdilik Seçilem item özellikleri consolea basalım.

App.js

```
import React from 'react'
import SearchBar from './SearchBar'
import youtube from '../apis/youtube'
import VideoList from './VideoList'
class App extends React.Component{
  state = {videos:[], selectedVideo:null}
  onTermSubmit = async (term) => {
    const response = await youtube.get('/search',{
      params:{
        q:term
      }
    })
    this.setState({videos:response.data.items})
  }
  onVideoSelect = (vid) => {
    console.log('From the app!', vid)
  }
  render(){
    return (
      <div className="ui container">
        <SearchBar onFormSubmit={this.onTermSubmit}/>
        I have {this.state.videos.length} videos
        <br/>
        <hr/>
        <VideoList onVideoSelect={this.onVideoSelect}
videosprops={this.state.videos}/>
      </div>
    )
  }
}
export default App
```

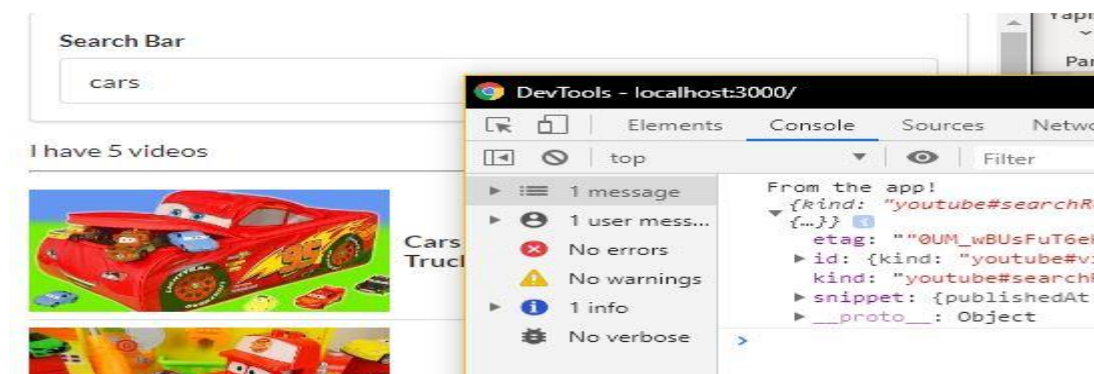
VideoList.js

```
import React from 'react'
import VideoItem from './VideoItem'
const VideoList = (props) => {
  const Item = props.videosprops.map((vid) => {
    return <VideoItem onVideoSelect={props.onVideoSelect}
itemprops={vid}/>
  })
  return (
    <div className="ui relaxed divided list" >
      {Item}
    </div>
  )
}
export default VideoList
```

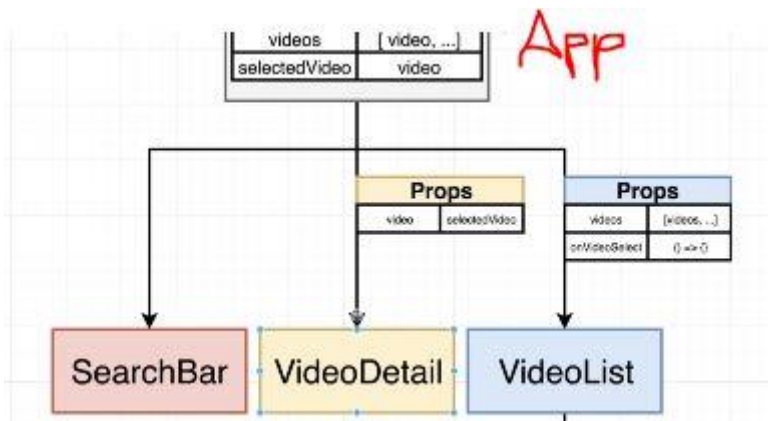
VideoItem.js

```
import React from 'react'
import '../css/VideoItem.css'
const VideoItem = (props) => {
  return(
    <div onClick={() => props.onVideoSelect(props.itemprops)}
className="video-item item">
      <img className="ui image"
src={props.itemprops.snippet.thumbnails.medium.url}/>
      <div className="content">
        <div className="header">
          {props.itemprops.snippet.title}
        </div>
      </div>
    </div>
  )
}
export default VideoItem
```

Listelenenden bir tane tıklayınca onun verileri consola yazdırdık.



Şimdi ise Seçileni render et seçilmeyeni render etme koşullu renderlama.



App.js

```
import React from 'react'
import SearchBar from './SearchBar'
import youtube from '../apis/youtube'
import VideoList from './VideoList'
import VideoDetail from './VideoDetail'

class App extends React.Component{
  state = {videos:[],selectedVideo:null}
  onTermSubmit = async (term) => {
    const response = await youtube.get('/search',{
      params:{
        q:term
      }
    })
    //console.log(response)
    this.setState({videos:response.data.items})
  }
  onVideoSelect = (vid) => {
    //console.log('From the app!', vid)
    this.setState({selectedVideo:vid})
  }
  render(){
    return (
      <div className="ui container">
        <SearchBar onFormSubmit={this.onTermSubmit}/>
        I have {this.state.videos.length} videos
        <VideoDetail videoDetail={this.state.selectedVideo}/>
        <VideoList onVideoSelect={this.onVideoSelect}
        videos={this.state.videos}/>
      </div>
    )
  }
}
export default App
```

VideoDetail.js

```
import React from 'react'

const VideoDetail = ({videoDetay}) => {
  return (
    <div>
      {videoDetay.snippet.title}
    </div>
  )
}

export default VideoDetail
```

Hata:

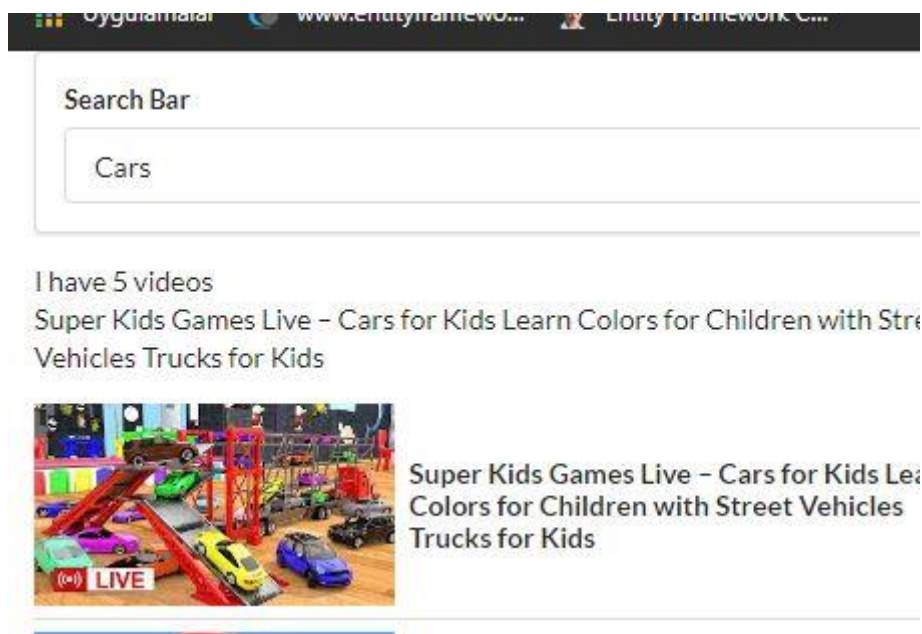
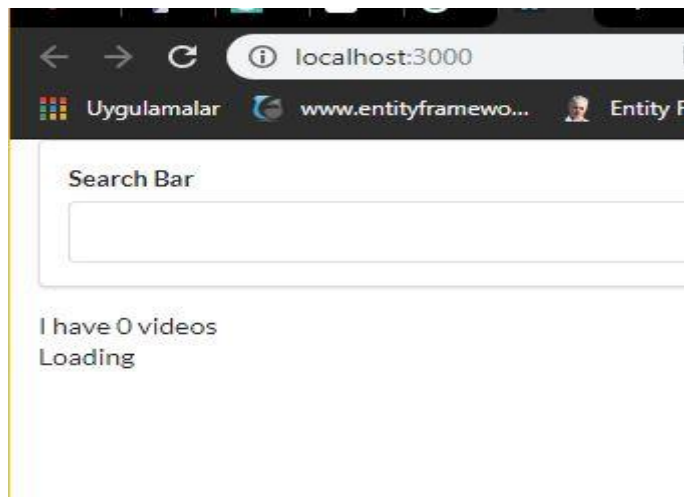
TypeError: Cannot read property 'snippet' of null

Çünkü selectedVideo default olarak null olduğundan snippet bulamıyor bunun önüne geçmek için

```
import React from 'react'

const VideoDetail = ({videoDetay}) => {
  if(!videoDetay){
    return <div>Loading</div>
  }
  return (
    <div>
      {videoDetay.snippet.title}
    </div>
  )
}

export default VideoDetail
```



&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

&&&

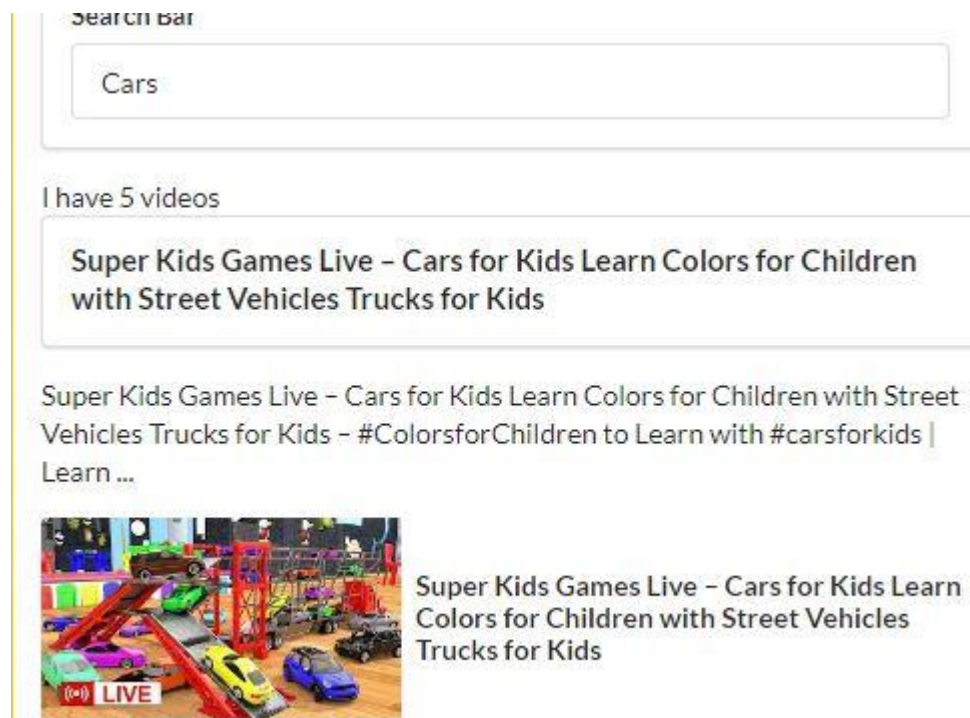
&&&

VideoDetail css biraz güzelleştirecez.

```
import React from 'react'

const VideoDetail = ({videoDetay}) => {
  if(!videoDetay){
    return <div>Loading</div>
  }
  return (
    <div >
      <div className="ui segment">
        <h4 className="ui header"> {videoDetay.snippet.title}</h4>
      </div>
      <p>{videoDetay.snippet.description}</p>
    </div>
  )
}

export default VideoDetail
```



&&&

&&&

&&&

&&&

&&&

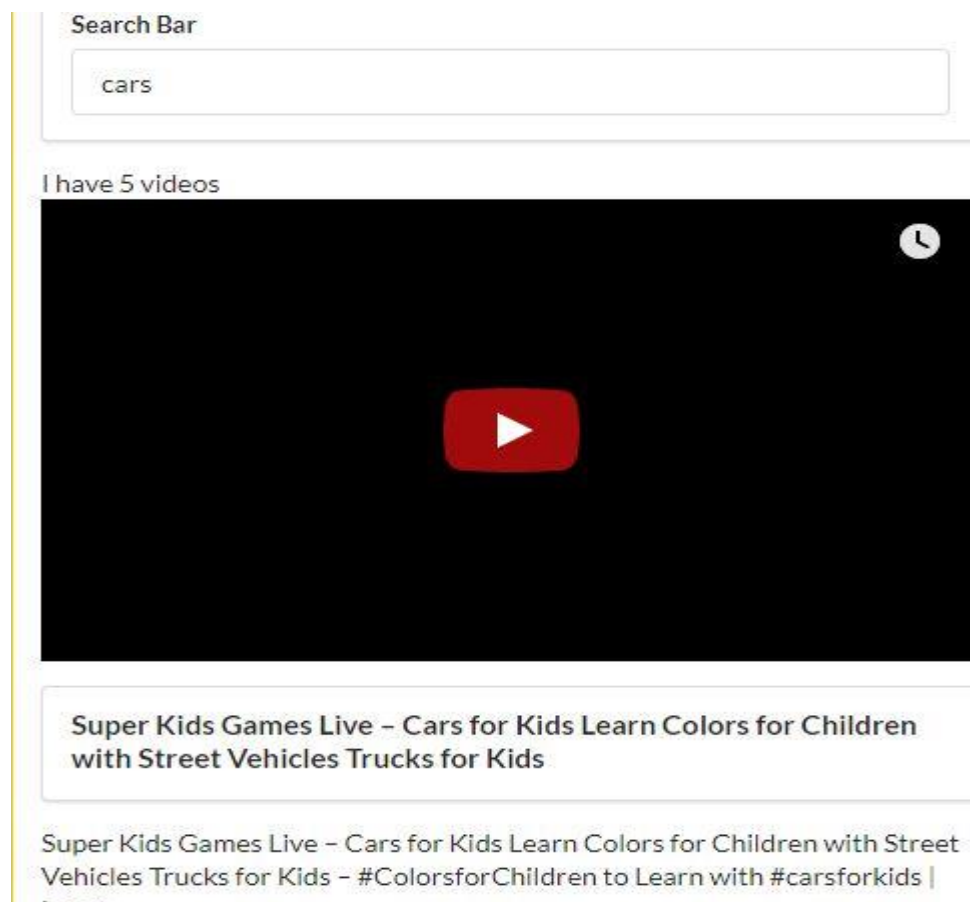
&&&

Şimdi ise Video playerı göstermeye geldi.

```
import React from 'react'

const VideoDetail = ({videoDetay}) => {
  if(!videoDetay){
    return <div>Loading</div>
  }
  const videoSrc = `https://www.youtube.com/embed/${videoDetay.id.videoId}`
  return (
    <div >
      <div className="ui embed">
        <iframe src={videoSrc} />
      </div>
      <div className="ui segment">
        <h4 className="ui header"> {videoDetay.snippet.title}</h4>
      </div>
      <p>{videoDetay.snippet.description}</p>
    </div>
  )
}

export default VideoDetail
```



<iframe> elements must have a unique title property jsx-a11y/iframe-has-title

```
<iframe title="video player" src={videoSrc} />
```

img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text

```
<img alt={props.itemprops.snippet.title} className="ui image"
src={props.itemprops.snippet.thumbnails.medium.url}/>
```

Each child in a list should have a unique "key" prop.

Check the render method of `VideoList`

```
return <VideoItem key={vid.id.videoId} onVideoSelect={props.onVideoSelect}
itemprops={vid}/>
```

Görseli güzelleştirme işlemleri

<https://semantic-ui.com/collections/grid.html>

App.js

```
return (
  <div className="ui container">
    <SearchBar onFormSubmit={this.onTermSubmit}/>
    <div className="ui grid">
      <div className="ui row">
        <div className="eleven wide column">
          <VideoDetail
videoDetay={this.state.selectedVideo}/>
        </div>
        <div className="five wide column ">
          <VideoList onVideoSelect={this.onVideoSelect}
videosprops={this.state.videos}/>
        </div>
      </div>
    </div>
  </div>
)
```

localhost:3000

Uygulamalar www.entityframewo... Limited time offer... Entity Framework C... ASP.NET MVC Tutor... TutorialsTeacher On...

Search Bar

Fenerbahçe

Loading

18 AĞUSTOS FENERBAHÇE TRANSFER GÜNDEMİ

ARMAN SOL BEK DIEGO LAXALT

LAXALT ANALİZİ

FENERBAHÇE SPOR KULÜBÜ

1907

Fenerbahçe Transfer Gündemi | Fenerbahçe'a Tek Serkan Reçber | 18 ağustos

Fenerbahçe SK

EĞLENCELİ FOTOĞRAF ÇEKİMİ

AVIS

Bizimkilerin Eğlenceli Fotoğraf Çekimi 🤔 #BackstageVlog

FENERBAHÇE GELİYOR!

#OmuzOmuzSampiyonluğa

Fenerbahçe Geliyor! #OmuzOmuzSa

localhost:3000

Uygulamalar www.entityframewo... Limited time offer... Entity Framework C... ASP.NET MVC Tutor... TutorialsTeacher On...

Search Bar

Fenerbahçe

Bizimkilerin Eğlenceli Fotoğraf Çekimi 🤔 #Ba...

Daha sonra izle

Paylaş

EĞLENCELİ FOTOĞRAF ÇEKİMİ

AVIS

Bizimkilerin Eğlenceli Fotoğraf Çekimi 🤔 #BackstageVlog

Fenerbahçemiz, Samandıra Can Bartu Tesisleri'nde yeni sezon sosyal medya forma çekimlerini gerçekleştirdi. En eğlenceli anlar sizi bekliyor! KANALA ...

18 AĞUSTOS FENERBAHÇE TRANSFER GÜNDEMİ

ARMAN SOL BEK DIEGO LAXALT

LAXALT ANALİZİ

FENERBAHÇE SPOR KULÜBÜ

1907

Fenerbahçe Transfer Gündemi | Fenerbahçe'a Tek Serkan Reçber | 18 ağustos

Fenerbahçe SK

EĞLENCELİ FOTOĞRAF ÇEKİMİ

AVIS

Bizimkilerin Eğlenceli Fotoğraf Çekimi 🤔 #BackstageVlog

FENERBAHÇE GELİYOR!

#OmuzOmuzSampiyonluğa

Fenerbahçe Geliyor! #OmuzOmuzSa

Arama yapılırca default olarak bir video seçili gelmesini istersek,

Açılır açılmaz default olarak gelmesini istiyorsak, componentDidMount

```
import React from 'react'
import SearchBar from './SearchBar'
import youtube from '../apis/youtube'
import VideoList from './VideoList'
import VideoDetail from './VideoDetail'

class App extends React.Component{
  state = {videos:[],selectedVideo:null}

  componentDidMount(){
    this.onTermSubmit('Fenerbahçe') //bir parametrelili //açılınca gelir.
  }

  onTermSubmit = async (term) => {
    const response = await youtube.get('/search',{
      params:{
        q:term
      }
    })
    //console.log(response)
    this.setState({
      videos:response.data.items,
      selectedVideo:response.data.items[0]
    })
  }

  onVideoSelect = (vid) => {
    //console.log('From the app!', vid)
    this.setState({selectedVideo:vid})
  }

  render(){
    return (
      <div className="ui container">
        <SearchBar onFormSubmit={this.onTermSubmit}/>
        <div className="ui grid">
          <div className="ui row">
            <div className="eleven wide column">
              <VideoDetail
videoDetay={this.state.selectedVideo}/>
            </div>
            <div className="five wide column ">
              <VideoList onVideoSelect={this.onVideoSelect}
videosprops={this.state.videos}/>
            </div>
          </div>
        </div>
      </div>
    )
  }
}
```



```
export default App
```

