

EPL 421: Systems Programming

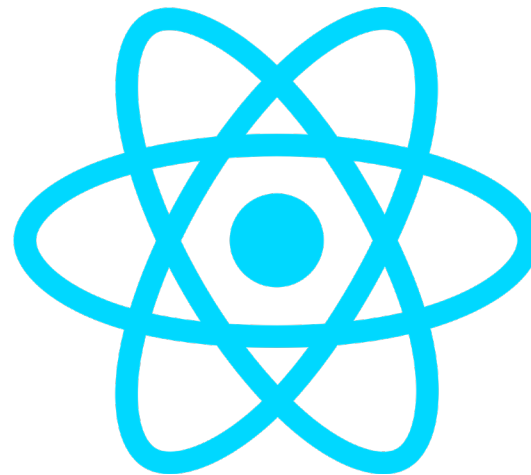
React.js Front-End Library FTP Client Implementation

Canciu Ionut – Cristian
(icanci01@ucy.ac.cy)

Ilias Kalaitzidis
(ikalai01@ucy.ac.cy)



University of Cyprus
Department of Computer
Science



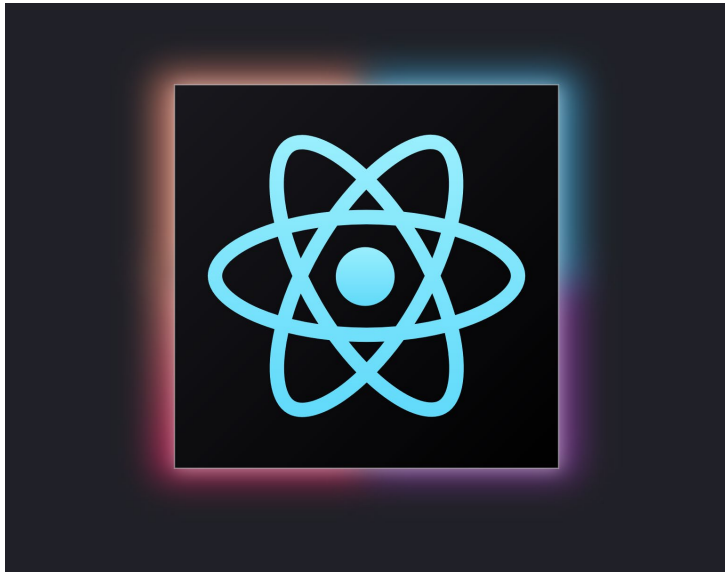
React

12 – MARCH - 2022

ΠΕΡΙΕΧΟΜΕΝΑ ΠΑΡΟΥΣΙΑΣΗΣ

- Τι είναι η React και ιστορική αναδρομή
- Επεξήγηση τεχνολογιών που χρησιμοποιούνται
- JavaScript ή TypeScript
- Χρήση της έννοιας ‘εφαρμογή μίας σελίδας’
- Ικανότητες της βιβλιοθήκης react
 - Πληθώρα ενσωματωμένων εξαρτήσεων, εργαλείων κ.α
 - React και JSX
 - Αντικείμενα, Κατάσταση, Ιδιότητες, Hooks
- MERN framework stack για δημιουργία εφαρμογών στην react
- Μερικές διαθέσιμες επιλογές για back-end
- Εγκατάσταση εργαλείων
- Η πρώτη εφαρμογή στην react και παράδειγμα “Hello World”
- Η δική μας υλοποίηση του FTP client
- Συμπεράσματα – Θετικά και Αρνητικά από την δική μας εμπειρία
- Εναλλακτικές επιλογές
- Βιβλιογραφία

ΤΙ ΕΙΝΑΙ Η REACT ΚΑΙ ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ



REACT

(A JavaScript library for building user interfaces)

- “**React** (*also known as React.js or ReactJS*) is a free and open-source front-end JavaScript library for building user interfaces based on UI components.”
- “It is maintained by **Meta** (formerly **Facebook**) and a community of individual developers and companies.” -*Wikipedia*
 - Original Author: **Jordan Walke**
 - Initial Release: **May 29, 2013**
 - Latest Stable Version: **18.0.0 / 29 March 2022**

ΕΠΕΞΗΓΗΣΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ (1)

Διαδίκτυο, HTML, CSS, JavaScript, JavaScript HTML DOM

- Το **Διαδίκτυο** (αγγλικά: Internet) είναι παγκόσμιο σύστημα διασυνδεδεμένων δικτύων υπολογιστών, οι οποίοι χρησιμοποιούν καθιερωμένη ομάδα πρωτοκόλλων, η οποία συχνά αποκαλείται "TCP/IP" – Wikipedia
- Όπως καλά γνωρίζουμε οι ιστοσελίδες που επισκεπτόμαστε στο διαδίκτυο τις πλείστες φορές αποτελούνται από:
 - + **HTML – (HyperText Markup Language)** αρχεία. (π.χ. index.html)
Είναι η βασική γλώσσα δόμησης σελίδων του World Wide Web που επισκεπτόμαστε. Τα αρχεία αυτά δύναται να έχουν και εφαρμοσμένη μορφοποίηση CSS και ενσωματωμένο κώδικα *JavaScript*.
 - + **CSS – (Cascading Style Sheets)** αρχεία. (π.χ. index.css)
Γλώσσα προορισμένη να αναπτύσσει στυλιστικά μια ιστοσελίδα προσθέτοντας διάφορα χαρακτηριστικά, χρώματα, στοίχιση, κ.α.
 - + **JavaScript** αρχεία. (π.χ. index.js)
Γλώσσα προορισμένη στην επεξεργασία της επικοινωνίας του χρήστη με το γραφικό περιβάλλον, user interface, μιας ιστοσελίδας.



ΕΠΕΞΗΓΗΣΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ (2)

HTML – (HyperText Markup Language)

```
*index.html - Notepad
File Edit Format View Help
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <title>This is an example!</title>
  </head>
  <body>
    <h1>My First Heading 1!</h1>
    <h2>My First Heading 2!</h2>
    <p>My First Paragraph.</p>
    <div>
      <table>
        <tr>
          <th></th>
          <th></th>
          <th></th>
        </tr>
      </table>
    </div>

    <form>
      <input type="text" name="name" placeholder="Your name" />
      <input type="email" name="email" placeholder="Your email" />
      <input type="submit" value="Submit" />
    </form>
  </body>
</html>
```



ΕΠΕΞΗΓΗΣΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ (3)

CSS – (Cascading Style Sheets)

```
index.css - Notepad
File Edit Format View Help
body {
  background-color: #b35f5f;
  padding: 10px;
}

h1 {
  color: #fff;
  font-size: 2em;
  margin: 0;
}

p {
  color: #2bff00;
  font-size: 1em;
  margin: 0;
}

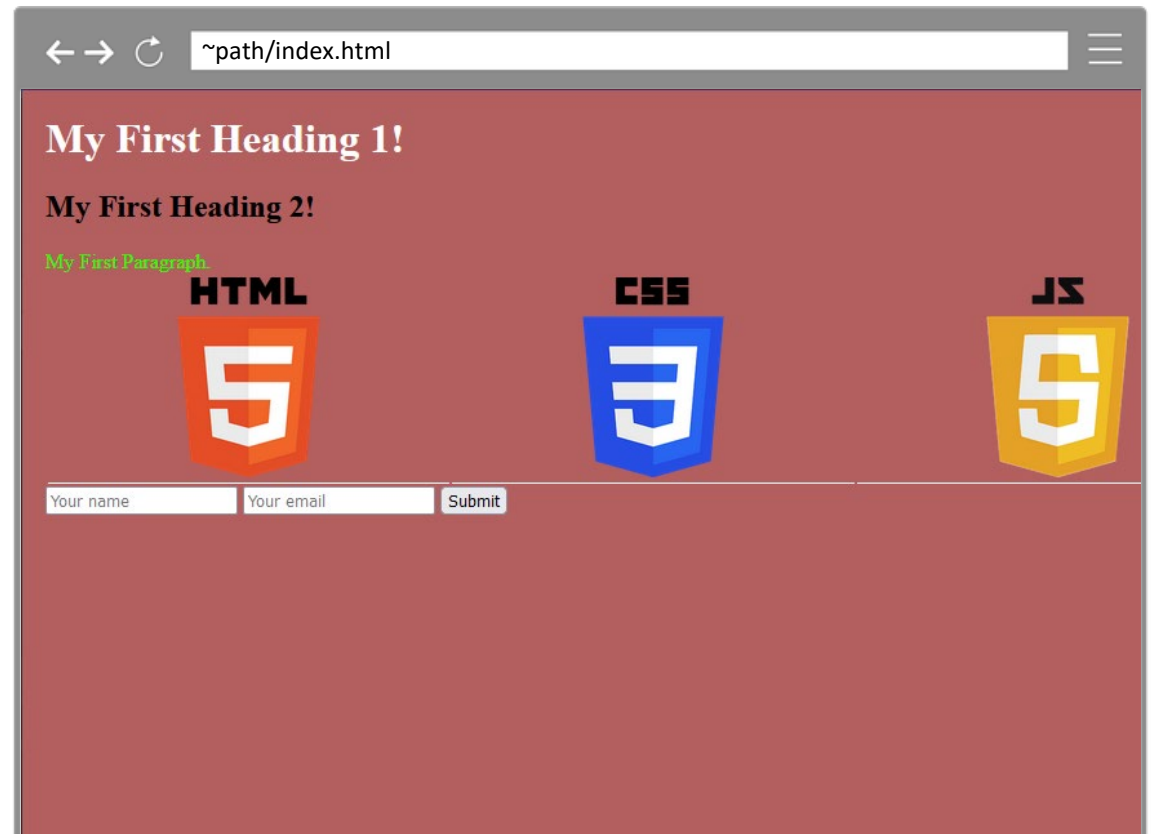
img {
  width: 50%;
  height: auto;
}

th {
  border-bottom: 1px solid #fff;
}
```

index.html

```
<meta name="viewport" content="width=device-
<link rel="stylesheet" href="index.css">

<title>This is an example!</title>
head>
```



ΕΠΕΞΗΓΗΣΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ (4)

JavaScript

```
index.js - Notepad
File Edit Format View Help
function validateForm () {
    var val_name = document.getElementById("name");
    var val_email = document.getElementById("email");
    if (val_name.value == ""){
        // change the style of the input box to red
        val_name.style.backgroundColor = "red";
        val_email.style.backgroundColor = "red";
        // change the text of the paragraph
        changeParagraph();
    }
}

function changeParagraph() {
    var para = document.getElementById("para");
    para.innerHTML = "WRONG!!!!";
    para.style.color = "red";
}

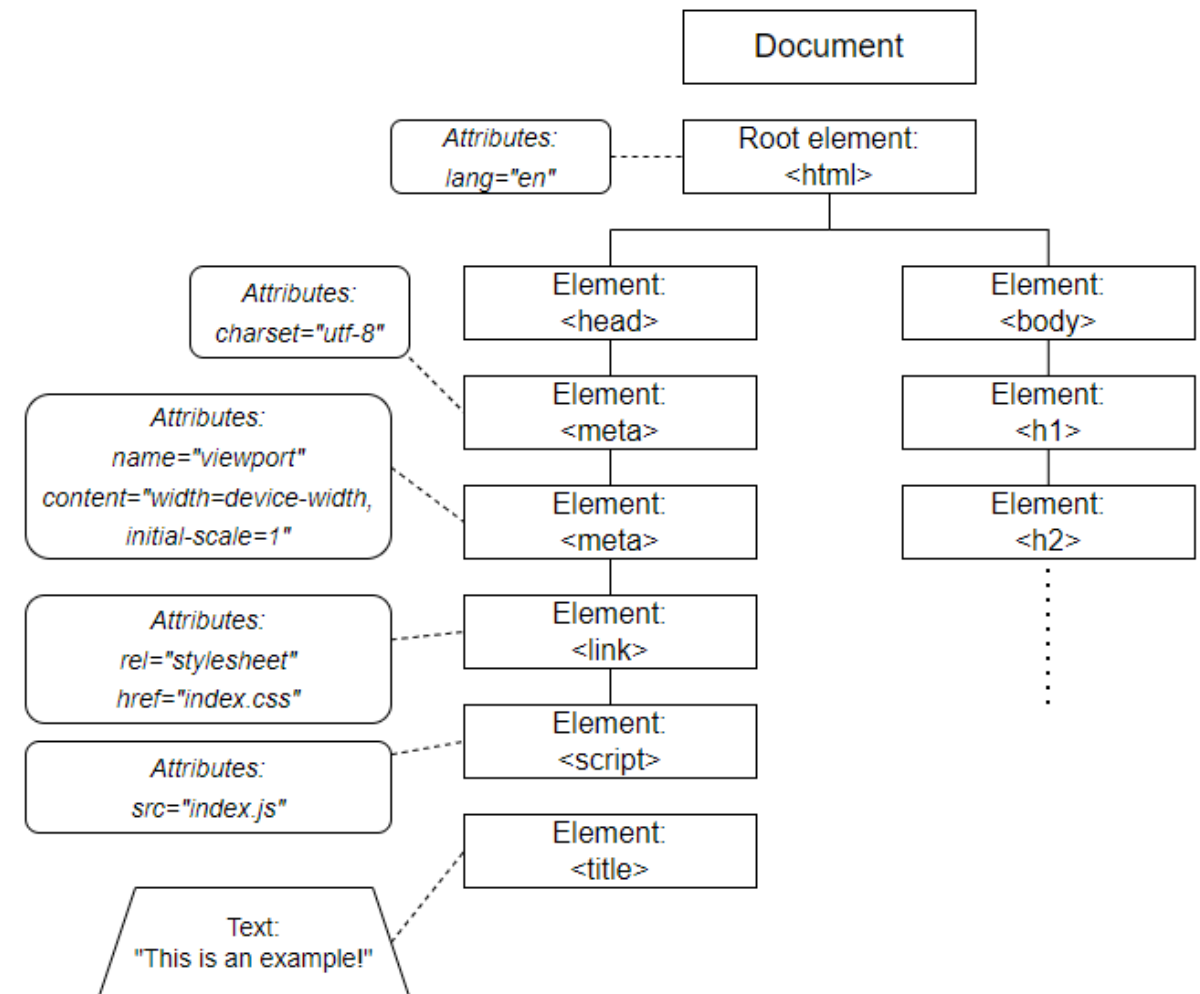
<meta name= viewport content= width=device-w
<link rel="stylesheet" href="index.css">
<script src="index.js"></script>
<title>This is an example!</title>
<h2>My First Heading 2!</h2>
<p id="para">My First Paragraph.</p>
<div>
</div>
<button onclick="validateForm()">Click Me!</button>
<form>
    <input type="text" id="name" name="name" placeholder
```



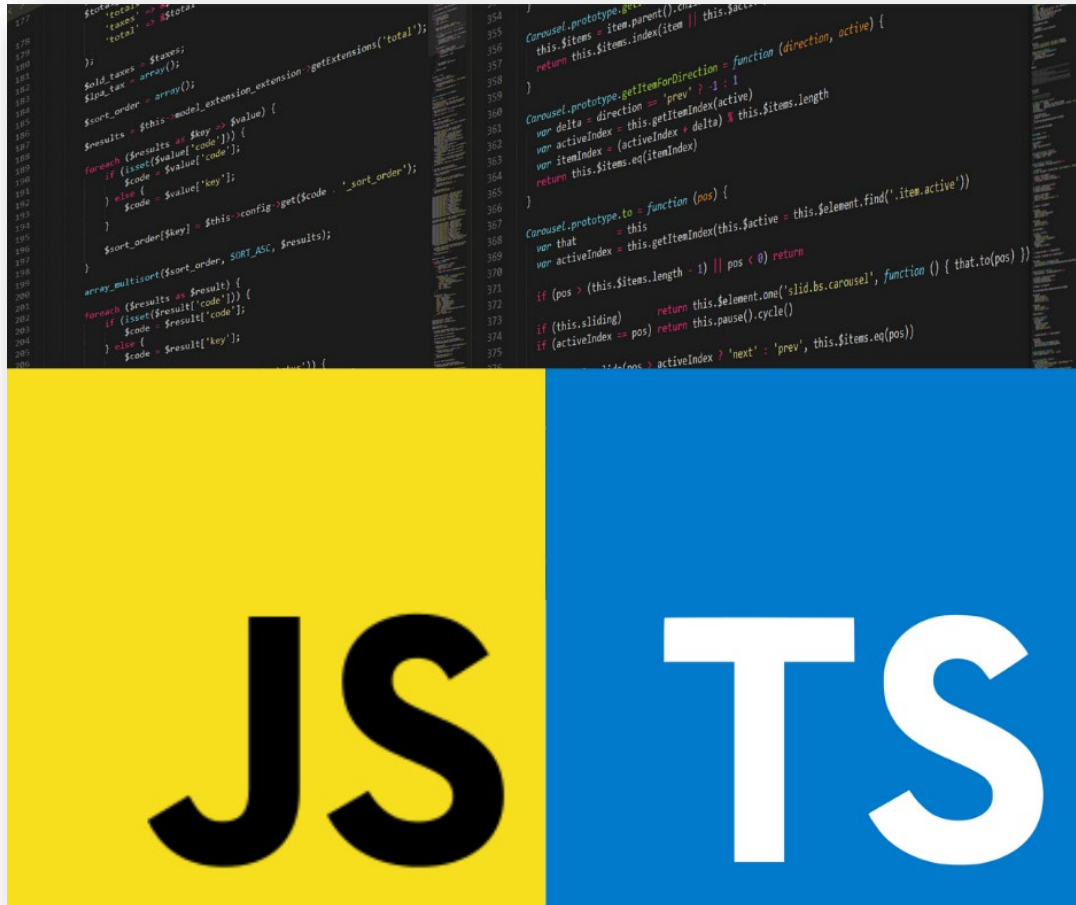
ΕΠΕΞΗΓΗΣΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ (5)

JavaScript HTML DOM

- Το **HTML DOM** αποτελεί την δομή ενός αρχείου HTML, και την σύνδεση που έχουν τα διάφορα στοιχεία που παρουσιάζονται σε ένα τέτοιο αρχείο. Ένα αρχείο JavaScript, μπορεί να εισέλθει στην δομή του DOM, και να την τροποποιεί.
- Αναφερόμαστε σε αυτό γιατί η React, είναι ικανή να επεξεργάζεται και να τροποποιεί αυτή την δομή αρχείου για να δημιουργήσει μία διαδραστική εμπειρία πλοήγησης στο διαδίκτυο. Λέγεται **React Render DOM**. *



JAVASCRIPT Η TYPESCRIPT (1)



JavaScript ή TypeScript Η επιλογή του κάθε προγραμματιστή

Η βιβλιοθήκη React, αφού είναι επέκταση της **JavaScript**, είναι αναμενόμενο να υποστηρίζει αυτή την γλώσσα.

Υπάρχει όμως η δυνατότητα ο κώδικας να είναι γραμμένος και σε **TypeScript**

Η **TypeScript** προσφέρει type safety, αφού ορίζουμε το είδος της μεταβλητής.

Χρειάζεσαι έναν αριθμό, τότε τον δηλώνεις και η μεταβλητή θα είναι πάντα αριθμός, συμβολοσειρά; Το ίδιο!

Προσφέρει και άλλες επεκτάσεις στην JavaScript.

JAVASCRIPT H TYPESCRIPT (2)

JavaScript

```
example.js - Notepad
File Edit Format View Help
function greeter(firstName, lastName) {
    return "Hello " + firstName + " " + lastName;
}

var firstName = "John";
var lastName = "Doe";

console.log(greeter(firstName, lastName));
```

Ln 1, Col 1 100% Windows (CRLF)

```
[Running] node "c:\Users\
Hello John Doe
```

JS

TypeScript

```
example.ts - Notepad
File Edit Format View Help
function greeter(firstName: string, lastName: string) {
    return "Hello, " + firstName + " " + lastName;
}

let firstName: string = "Jane";
let lastName: string = "Doe";

console.log(greeter(firstName, lastName));
```

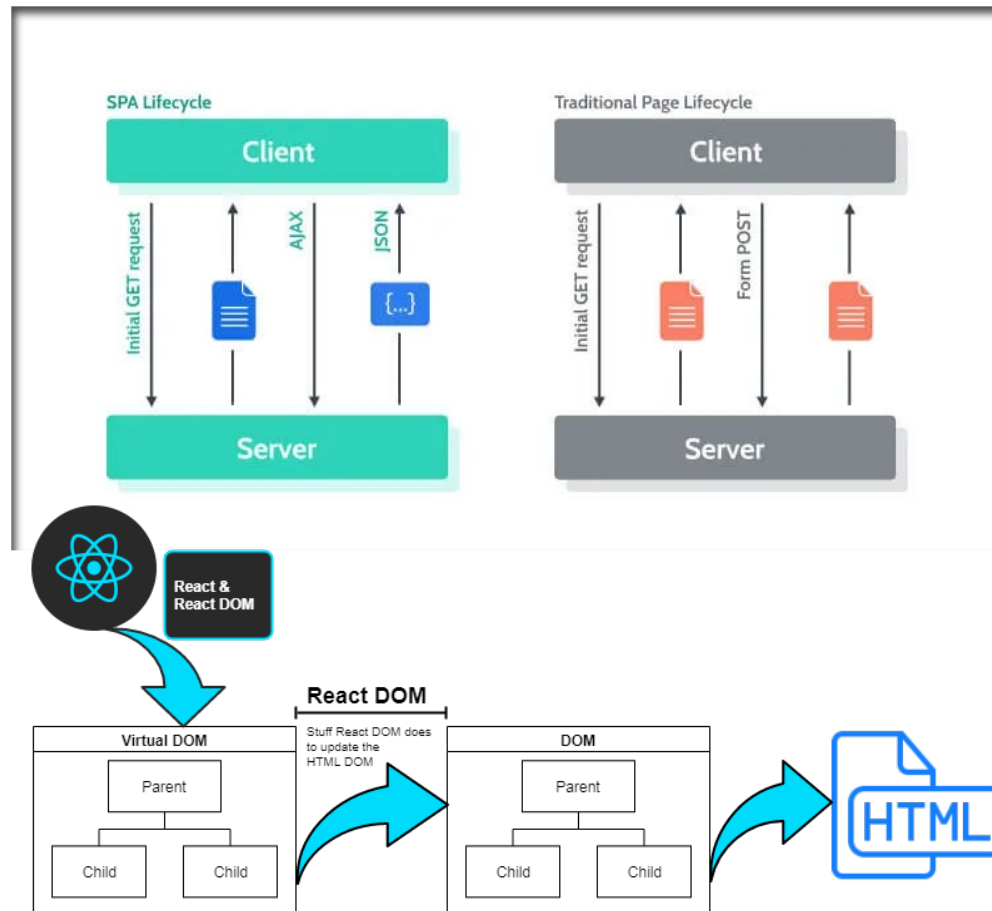
Ln 8, Col 43 100% Windows (CRLF)

```
[Running] ts-node "c:\Users\
Hello, Jane Doe
```

TS

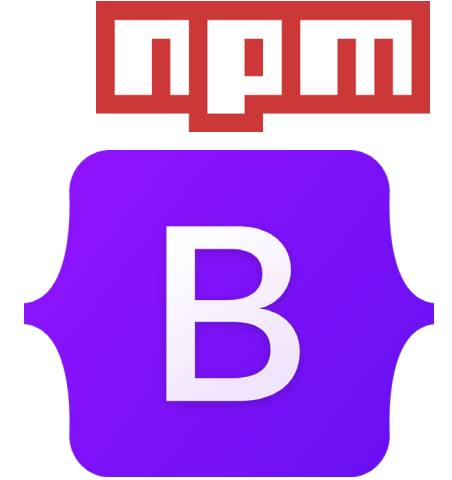
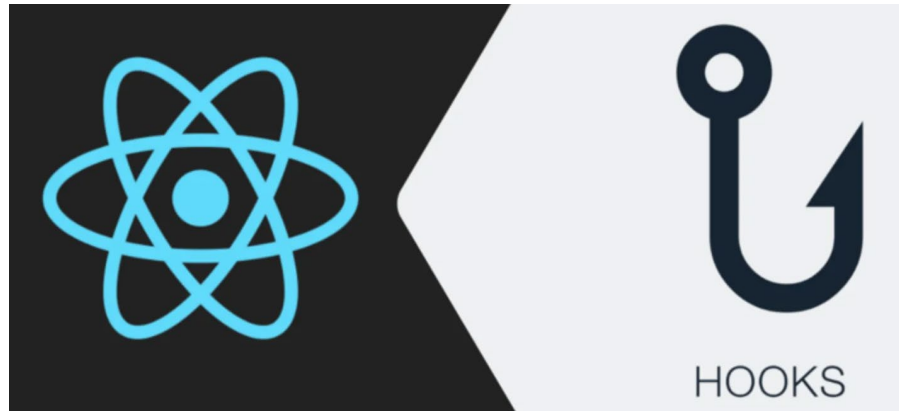
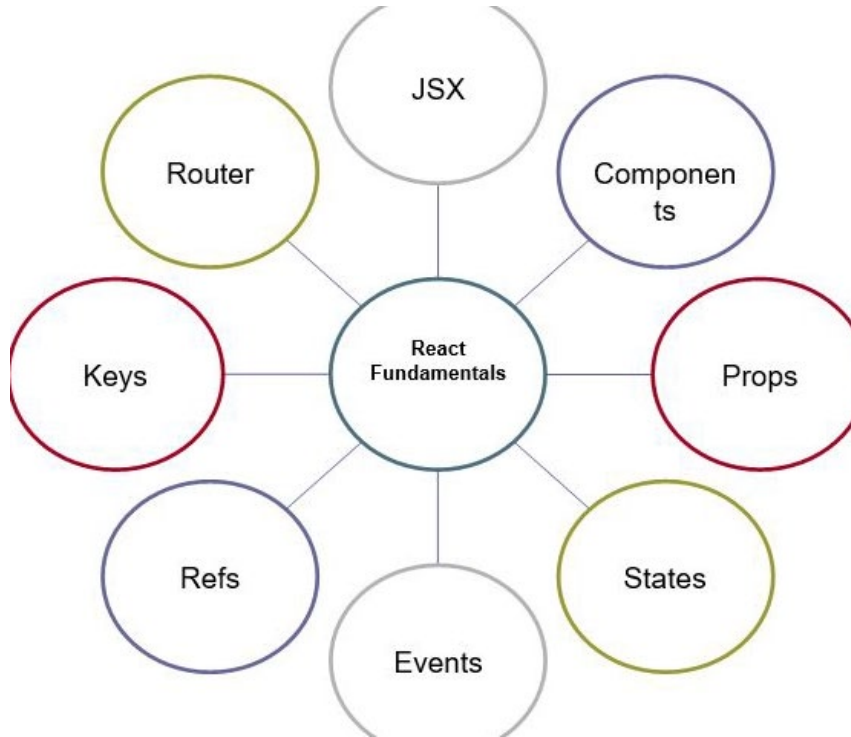
ΧΡΗΣΗ ΤΗΣ ΕΝΝΟΙΑΣ ΕΦΑΡΜΟΓΗ ΜΙΑΣ ΣΕΛΙΔΑΣ

Single Page Application - SPA



- ▶ Η React λειτουργεί ως μια Εφαρμογή μιας Σελίδας, όπου ο χρήστης αλληλοεπιδρά με τα στοιχεία στην οθόνη του. Σε αντίθεση με ένα κανονικό ιστότοπο HTML, στις εφαρμογές React δεν απαιτείται η 'αλλαγή' σελίδων. (π.χ. *index.html*, *contact_us.html* κ.α.)
- ▶ Το περιεχόμενο του κάθε στοιχείου, αποδίδεται και παρουσιάζεται (render) σε μια προεπιλεγμένη και σταθερή σελίδα.
- ▶ **web-application NOT website**
- ▶ **React Render.DOM**

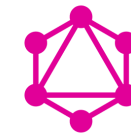
ΙΚΑΝΟΤΗΤΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ REACT (1)



Redux

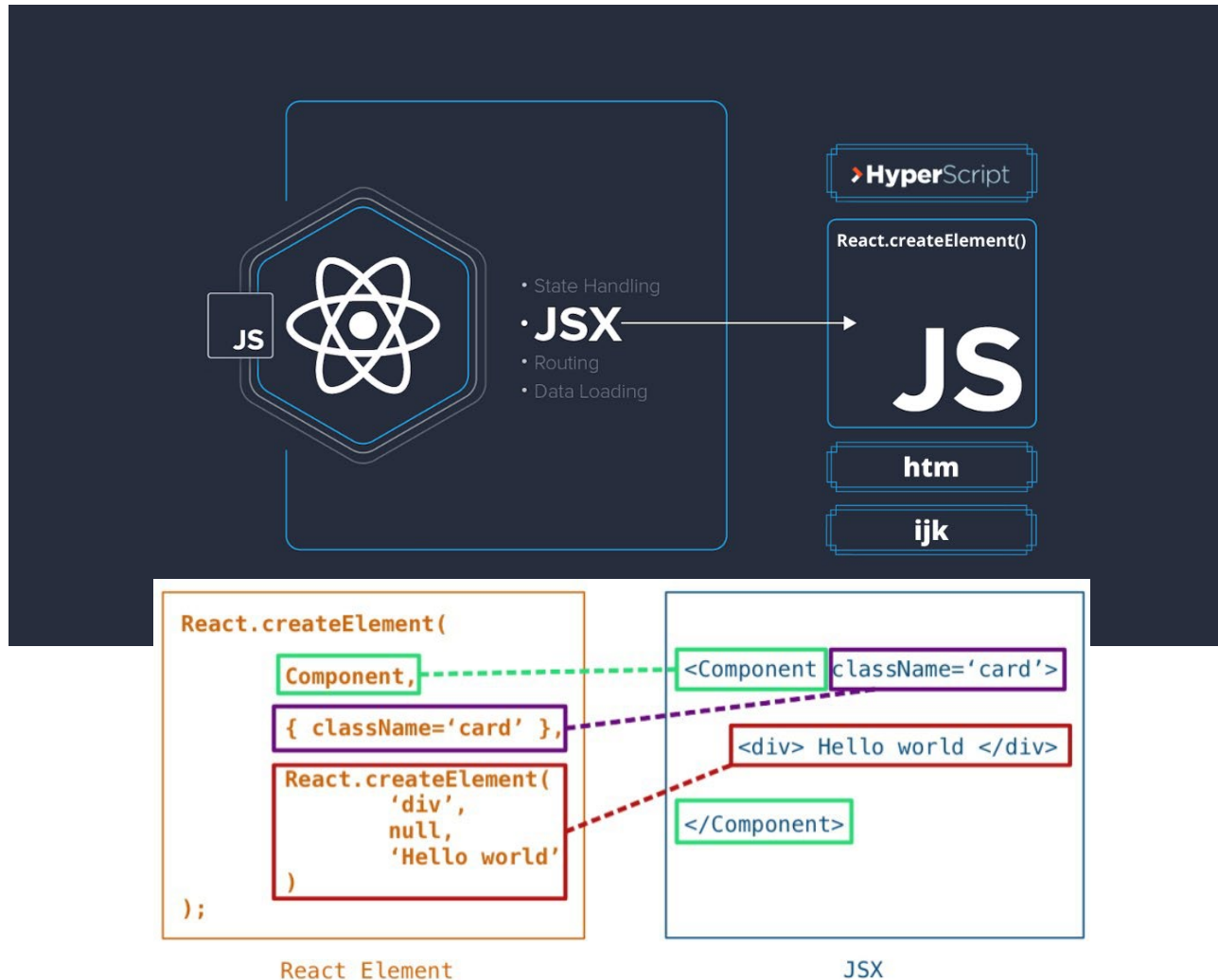


chakra



GraphQL

ΙΚΑΝΟΤΗΤΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ REACT (2)

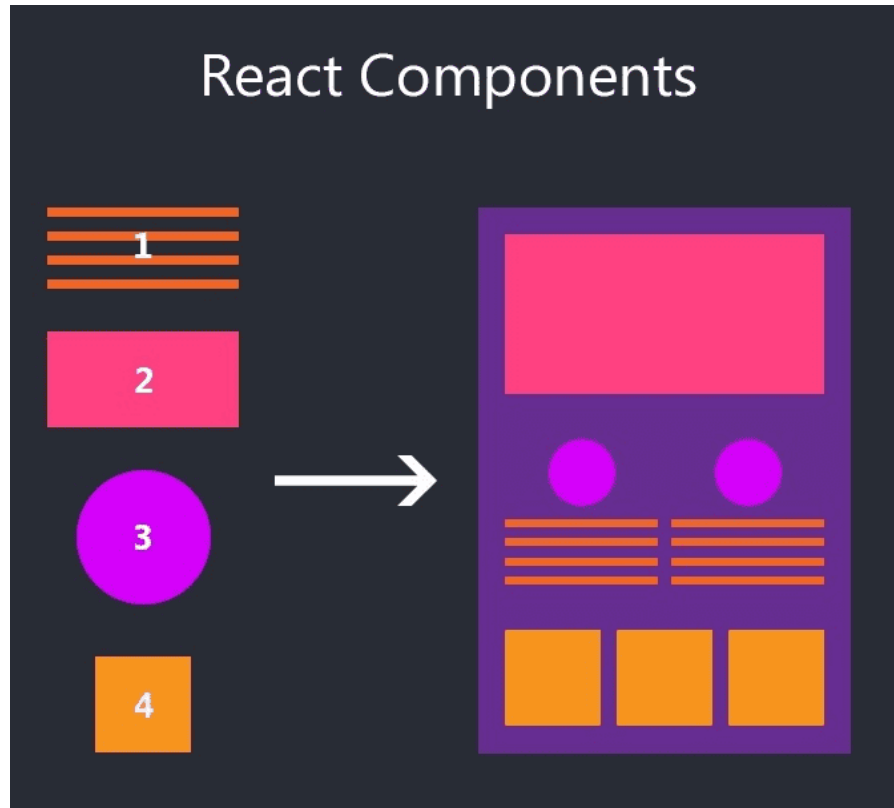


REACT ΚΑΙ JSX

- Η React αποτελεί επέκταση της γλώσσας προγραμματισμού JavaScript.
- Βασίζεται στην δημιουργία αντικειμένων που μπορούν να αλλάξουν κατάσταση και να παρουσιάζουν διάφορα εφέ, κινήσεις, μορφοποιήσεις, κ.α.
- Αποτελεί υπερσύνολο της JavaScript

ΙΚΑΝΟΤΗΤΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ REACT (3)

REACT ΚΑΙ ΤΑ COMPONENTS (1)



- Τα Components, στοιχεία στην React είναι τα βασικά δομικά στοιχεία. Ένα στοιχείο μπορεί να οριστεί μία φορά και να χρησιμοποιηθεί πολλές φορές. Κάθε στοιχείο έχει τη δική του ταυτότητα, που δίνεται από το όνομα της συνάρτησης που εξάγεται και για να το αναγνωρίσει η React ως στοιχείο θα πρέπει να δηλωθεί με το πρώτο γράμμα κεφαλαίο. Σύνταξη CamelCase.
- Όπως φαίνεται στην εικόνα, μόλις δημιουργηθεί ένα στοιχείο, μπορεί να χρησιμοποιηθεί σε πολλές θέσεις της εφαρμογής.
- Κάθε στοιχείο μπορεί να έχει τη δική του κατάσταση ή μπορεί να μοιράζεται μια κατάσταση μεταξύ 2 τοποθεσιών όπου αποδίδεται, rendered.
- Υπάρχουν και στοιχεία που δεν έχουν κατάσταση, stateless.

ΙΚΑΝΟΤΗΤΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ REACT (4)

```
helloworld.component.js - Notepad
File Edit Format View Help
import React from "react";
import "../helloworld.component.css";

function changeColor() {
  document.getElementById("hello-world").style.color = "green";
}

class HelloWorld extends React.Component {
  render() {
    return (
      <div className="hello-world">
        <h1>Hello World</h1>
        <button onClick={changeColor()}>Click Me!</button>
      </div>
    );
  }
}
```

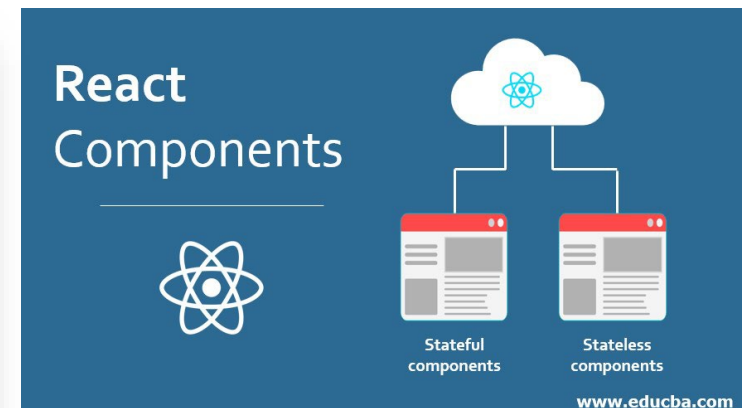
JavaScript

HTML

```
helloworld.component.css - Notepad
File Edit Format View Help
h1 {
  color: red;
  font-weight: bold;
  font-size: 1.5em;
}
```

REACT ΚΑΙ ΤΑ COMPONENTS (2)

Παράδειγμα React και χρήση
HTML, JSX, CSS και JavaScript



ΙΚΑΝΟΤΗΤΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ REACT (5)

```
nestingjsx.js - Notepad
File Edit Format View Help
import React from "react";

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

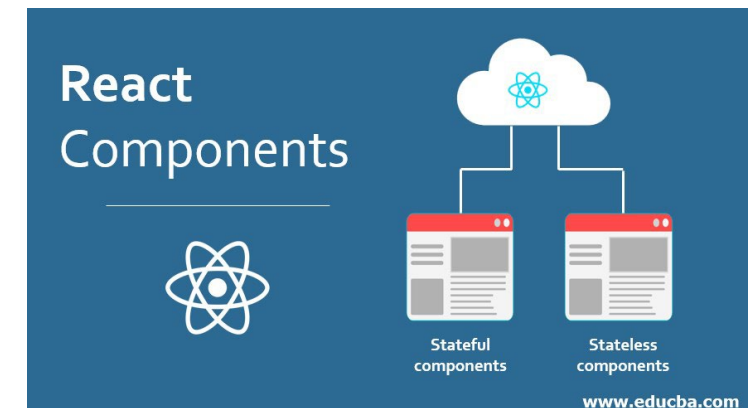
class Header extends React.Component {
  render() {
    return (
      <div>
        <h1>Header</h1>
      </div>
    )
  }
}

class Content extends React.Component {
  render() {
    return (
      <div>
        <h1>Content</h1>
      </div>
    )
  }
}
```

REACT ΚΑΙ ΤΑ COMPONENTS (3)

Παράδειγμα χρήσης React
και JSX component nesting

App |
| - Header
| - Content



ΙΚΑΝΟΤΗΤΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ REACT (6)

REACT ΚΑΙ ΤΑ COMPONENTS (3)

```
import * as React from 'react';
import { Component } from 'react';
```

```
class ListFilesComponent extends React.Component {
  [x: string]: any;
```

```
  state = {
    files: [
      {
        name: 'file1.txt',
        size: '1.2 MB',
        type: 'text/plain'
      },
      {
        name: 'file2.txt',
        size: '1.2 MB',
        type: 'text/plain'
      }
    ]
  }
```

```
  constructor() {
    super();
    this.state = { files: [] };
  }
```

```
  componentDidMount() {
    fetchFiles(files => this.setState({ files }));
  }
```

```
  render() {
    return (
      <div>
        <h1>List of files</h1>
        <ul>
          {this.state.files.map(file =>
            <li key={file.name}>
              <span>{file.name}</span>
              <span>{file.size}</span>
              <span>{file.type}</span>
            </li>
          )}
        </ul>
      </div>
    );
  }
```

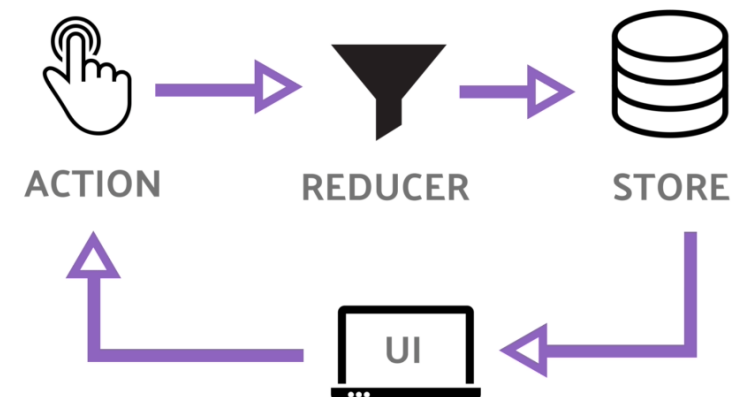
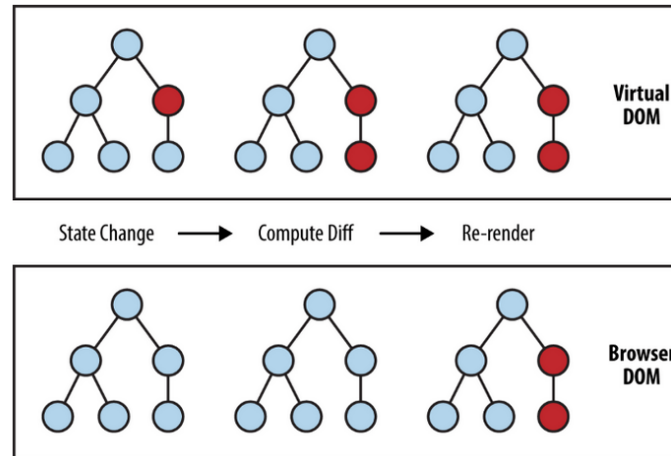
```
function fetchFiles(arg0: (files: any) => any) {
  throw new Error('Function not implemented.');
```

examples/component.example.tsx

ΙΚΑΝΟΤΗΤΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ REACT (7)

REACT ΚΑΙ ΚΑΤΑΣΤΑΣΗ

- Τα στοιχεία React μπορούν να διατηρήσουν μια κατάσταση.
- Το **Redux** είναι ένα container κατάστασης που διαχειρίζεται τις καταστάσεις των στοιχείων React.
- Το **React Render.DOM** απλώς αποδίδει τα διαφορετικά στοιχεία ανάλογα με την κατάσταση που έχουν σε κάποια στιγμή t_x.



ΙΚΑΝΟΤΗΤΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ REACT (8)

REACT ΚΑΙ ΙΔΙΟΤΗΤΕΣ (PROPS)

- Τα στηρίγματα στοιχείων είναι οι ιδιότητες που μεταβιβάζονται στο στοιχείο React και αλλάζουν την κατάσταση του στοιχείου.



```
AbsoluteCenter.tsx - Notepad
File Edit Format View Help
import * as React from "react";
import { Box, Flex } from "@chakra-ui/react";

export function AbsoluteCenter({
  children,
}): {
  children: React.ReactNode;
}): JSX.Element {
  return (
    <Box
      position="absolute"
      left="50%"
      top="50%"
      transform="translate(-50%, -50%)"
    >
      <Flex h="full" w="full" justify="center" align="center">
        {children}
      </Flex>
    </Box>
  );
}
```

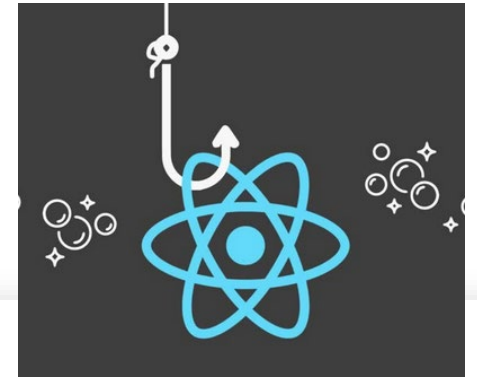
PROPS
(EVEN ANOTHER
COMPONENT)

inline css example

ΙΚΑΝΟΤΗΤΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ REACT (9)

REACT ΚΑΙ ΤΑ HOOKS

- Τα Hooks είναι λειτουργίες που επιτρέπουν στους προγραμματιστές να «αγκιστρωθούν» στις λειτουργίες κατάστασης και κύκλου ζωής ενός στοιχείου της React. Τα hook δεν λειτουργούν μέσα σε κλάσεις — σας επιτρέπουν να χρησιμοποιείτε το React χωρίς κλάσεις.
- Το React παρέχει μερικά ενσωματωμένα άγκιστρα όπως **useState**, **useContext**, **useReducer**, **useMemo** και **useEffect**. Άλλα τεκμηριώνονται στο Hooks API Reference. Το **useState** και το **useEffect**, που χρησιμοποιούνται πιο συχνά, προορίζονται για τον έλεγχο της κατάστασης και των ενεργειών, αντίστοιχα.



```
hook.example.js - Notepad
File Edit Format View Help
import React, { Component, useState } from "react";

function HelloWorldHook() {
  const [mssg, setMssg] = useState("Hello World");

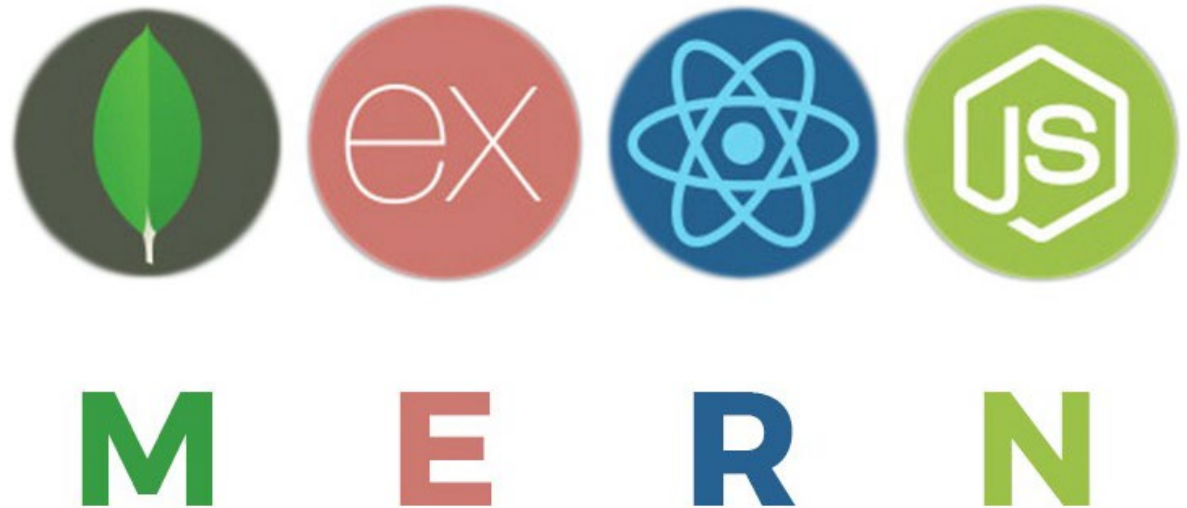
  return (
    <div className="hello-world">
      <h1>{mssg}</h1>
      <button onClick={setMssg("Changed!")}>Click Me!</button>
    </div>
  );
}
```

MERN FRAMEWORK STACK ΓΙΑ ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΩΝ ΣΤΗΝ REACT

- Ένα πολύ γνωστό πλαίσιο πλήρους στοίβας (framework stack*) που χρησιμοποιείται συχνά κατά τη δημιουργία εφαρμογών React είναι η στοίβα MERN.
 - M: MongoDB (database storage)
 - E: Express.js (back-end server)
 - R: React.js (front-end)
 - N: Node.js (Επικοινωνία API)
- Αναφερόμαστε σε αυτό το πλαίσιο (framework), γιατί αυτό χρησιμοποιήσαμε και εμείς χωρίς όμως την ανάγκη της σύνδεσης με κάποια βάση δεδομένων.

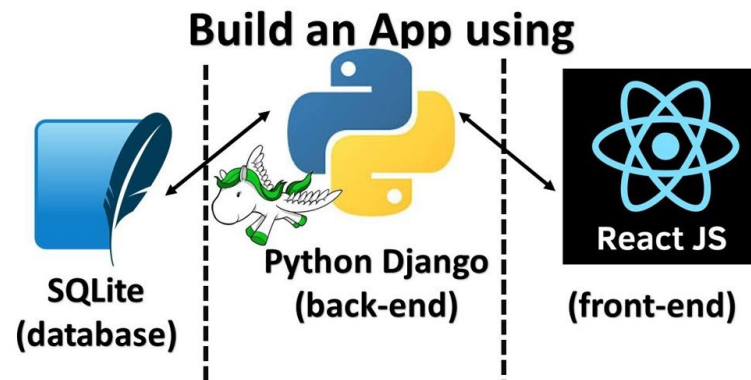
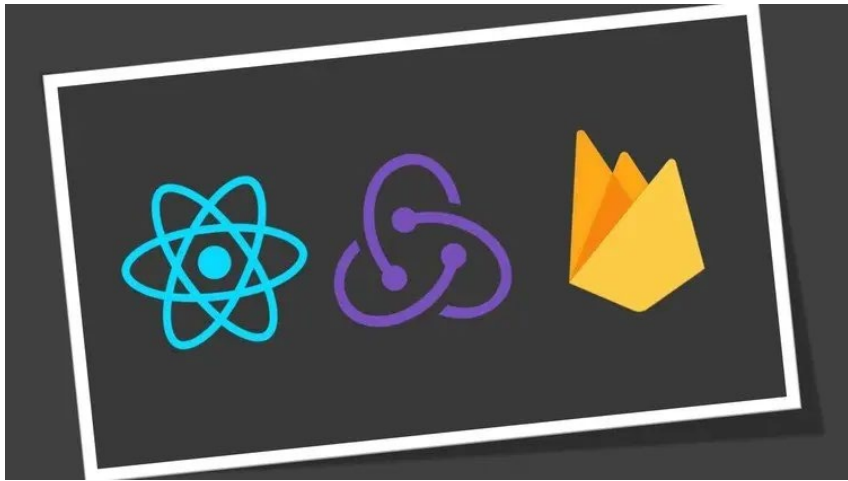
***Framework stack:** A framework in programming is a **tool that provides ready-made components or solutions that are customized in order to speed up development.** A framework may include a library but is defined by the principle of inversion of control (IoC).

REACT IS NOT A FRAMEWORK



ΜΕΡΙΚΕΣ ΔΙΑΘΕΣΙΜΕΣ ΕΠΙΛΟΓΕΣ ΓΙΑ BACK-END

- Ωστόσο, υπάρχουν ατελείωτες επιλογές που μπορούν να χρησιμοποιηθούν, π.χ., FReMP (Flask, React, MongoDB, Python)
- Για μια πιο σταθερή εφαρμογή, προτείνεται η χρήση ενός backend που μπορεί να επεξεργάζεται πολλαπλά αιτήματα ταυτόχρονα, π.χ. Java Vert.x (microservices).



VERT.X

ΕΓΚΑΤΑΣΤΑΣΗ ΕΡΓΑΛΕΙΩΝ

ΠΡΟΑΠΑΙΤΟΥΜΕΝΑ:

Node.js

Node Package Manager (npm)

Χρήσιμα εργαλεία για τη δημιουργία εφαρμογής πλήρους στοίβας:

- Yarn Package Manager
- MongoDB – (Database Storage)
- Docker – (Containers)
- GitHub Desktop – (Version Control)
- Postman – (API testing)
- κ.α.



```
bash-$ sudo apt install nodejs
```

```
bash-$ node -v (get node version)
```

```
bash-$ npm -v (get npm version)
```

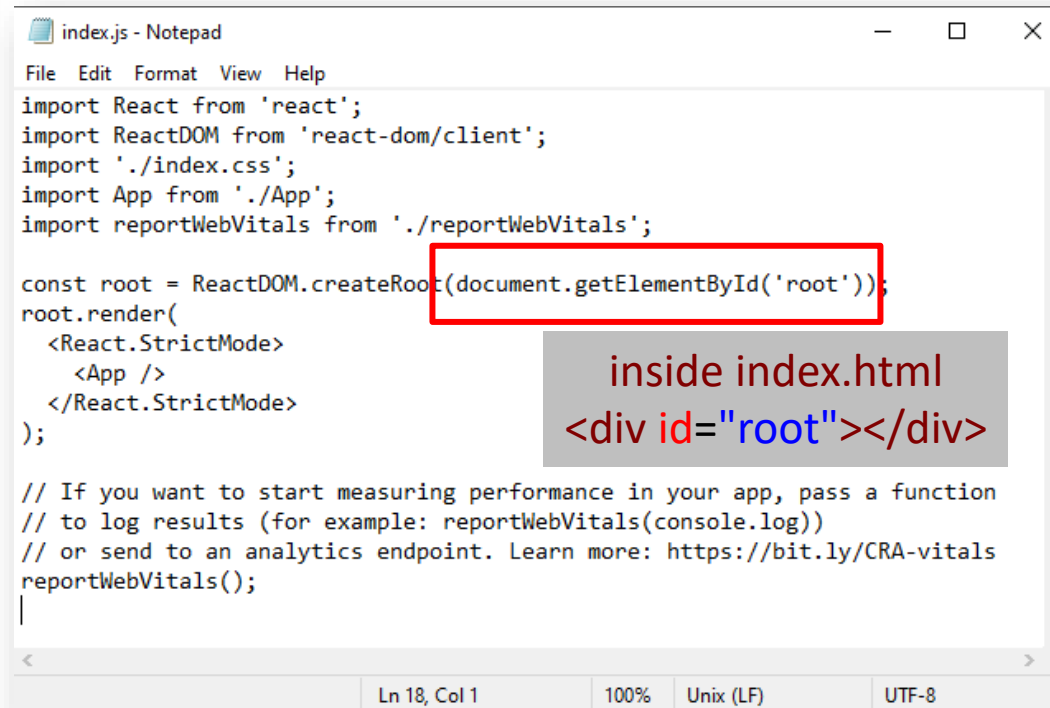
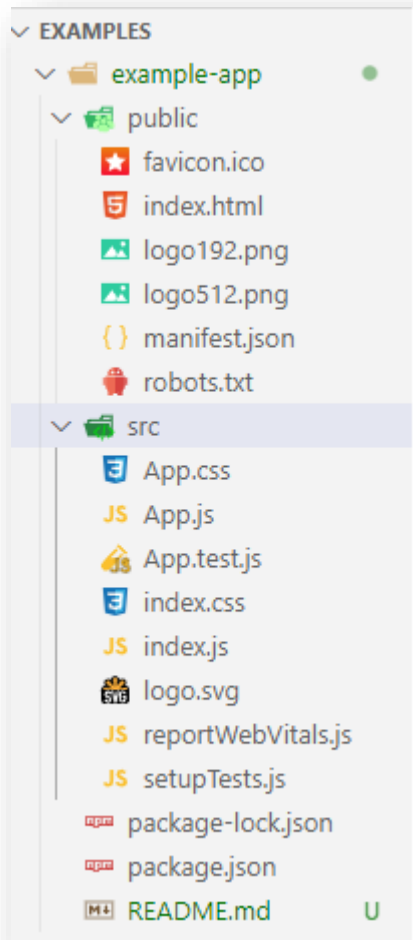


```
bash-$ npm install -g npm
```


Η ΠΡΩΤΗ ΕΦΑΡΜΟΓΗ ΣΤΗΝ REACT ΚΑΙ ΠΑΡΑΔΕΙΓΜΑ “HELLO WORLD”

ΑΡΧΙΚΗ ΔΗΜΙΟΥΡΓΙΑ ΔΟΜΗ ΜΙΑΣ ΕΦΑΡΜΟΓΗΣ ΣΤΗΝ REACT

➤ `npx create-react-app epl421-presentation-react`

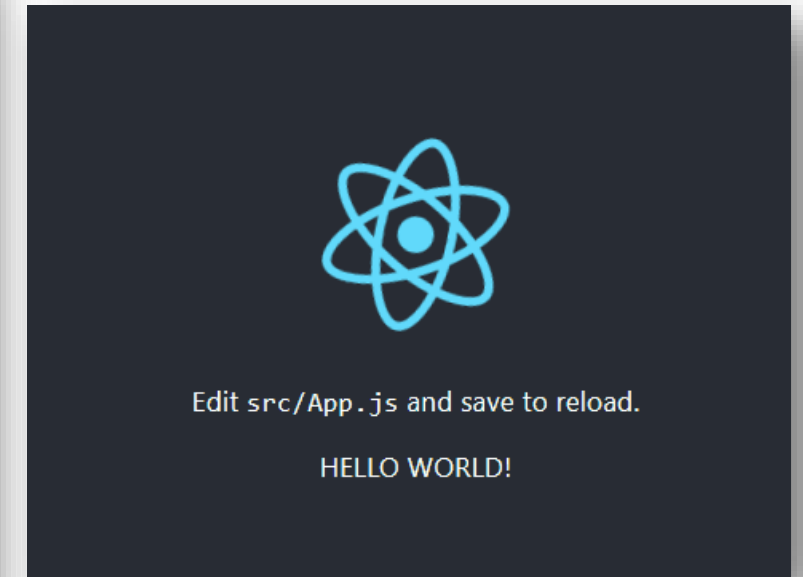


```
index.js - Notepad
File Edit Format View Help
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);

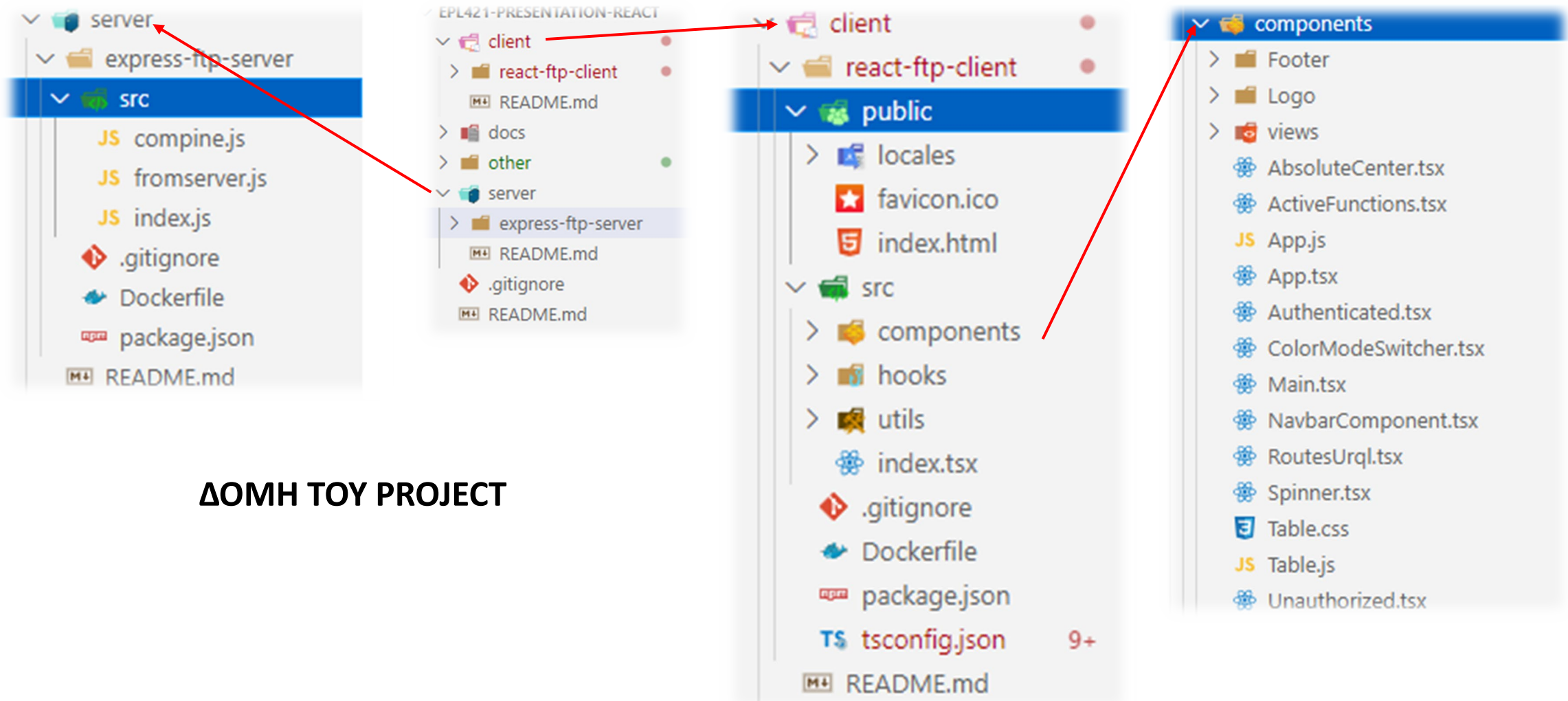
// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

inside index.html
`<div id="root"></div>`



Για να τρέξουμε την εφαρμογή χρησιμοποιούμε την εντολή
`bash-$ npm start`

Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (1)



ΔΟΜΗ ΤΟΥ PROJECT

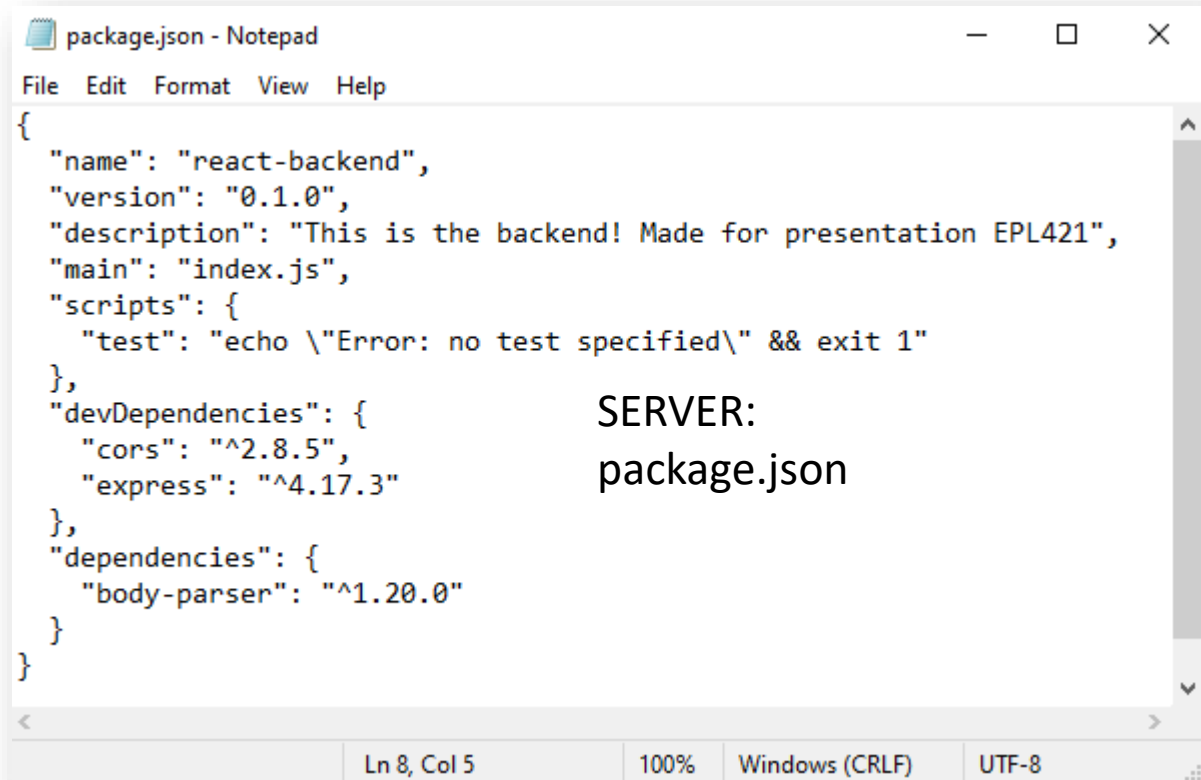
Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (2)



```
package.json - Notepad
File Edit Format View Help
{
  "name": "react-ftp-client",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "@chakra-ui/icons": "^1.1.7",
    "@chakra-ui/react": "^1.8.8",
    "@emotion/react": "^11.9.0",
    "@emotion/styled": "^11.8.1",
    "@hookform/resolvers": "^2.8.8",
    "@testing-library/jest-dom": "^5.16.4",
    "@testing-library/react": "^10.4.9",
    "@testing-library/user-event": "^12.8.3",
    "@types/jest": "^25.2.3",
    "@types/node": "^12.20.47",
    "@types/react": "^16.14.24",
    "@types/react-dom": "^16.9.14",
    "framer-motion": "^4.1.17",
    "i18next-browser-languagedetector": "^6.1.4",
    "i18next-xhr-backend": "^3.2.2",
    "react": "^18.0.0",
    "react-dom": "^18.0.0",
    "react-error-boundary": "^3.1.4",
    "react-i18next": "^11.16.2",
    "react-icons": "^3.11.0",
    "react-router-dom": "^6.3.0",
    "react-scripts": "5.0.0",
    "typescript": "^4.6.3",
    "use-subscription": "^1.6.0",
    "web-vitals": "^0.2.4",
    "webpack": "^5.72.0",
    "webpack-cli": "^4.9.2",
    "yup": "^0.32.11"
  }
}
```

CLIENT:
package.json

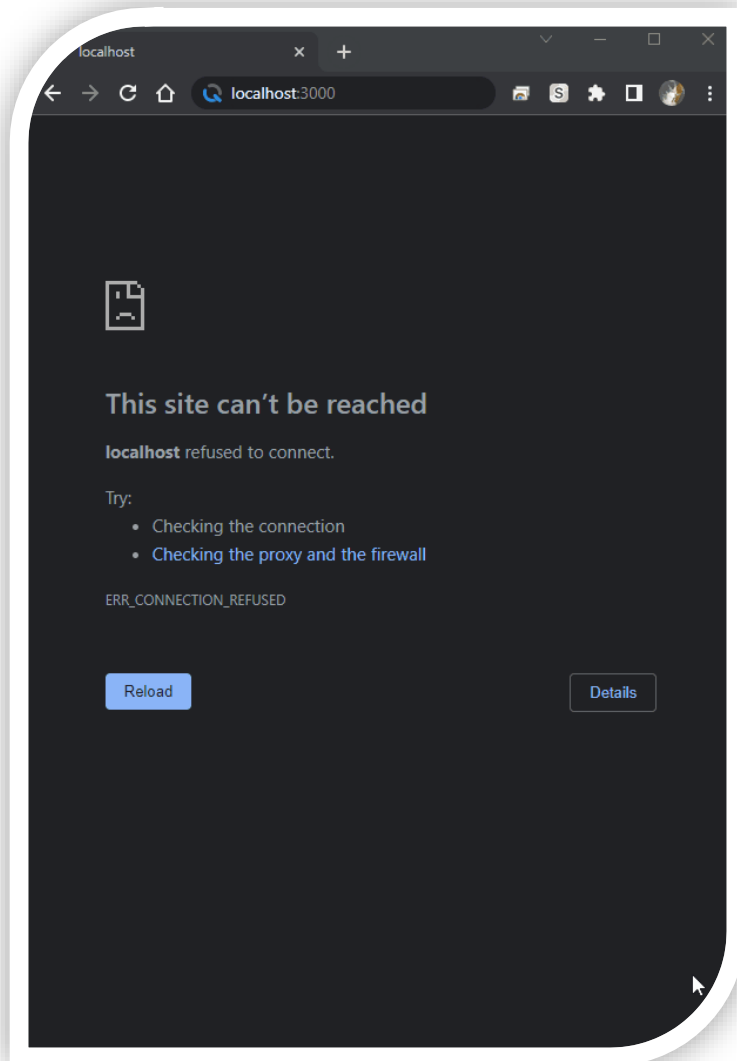
ΕΞΑΡΤΗΣΕΙΣ



```
package.json - Notepad
File Edit Format View Help
{
  "name": "react-backend",
  "version": "0.1.0",
  "description": "This is the backend! Made for presentation EPL421",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "devDependencies": {
    "cors": "^2.8.5",
    "express": "^4.17.3"
  },
  "dependencies": {
    "body-parser": "^1.20.0"
  }
}
```

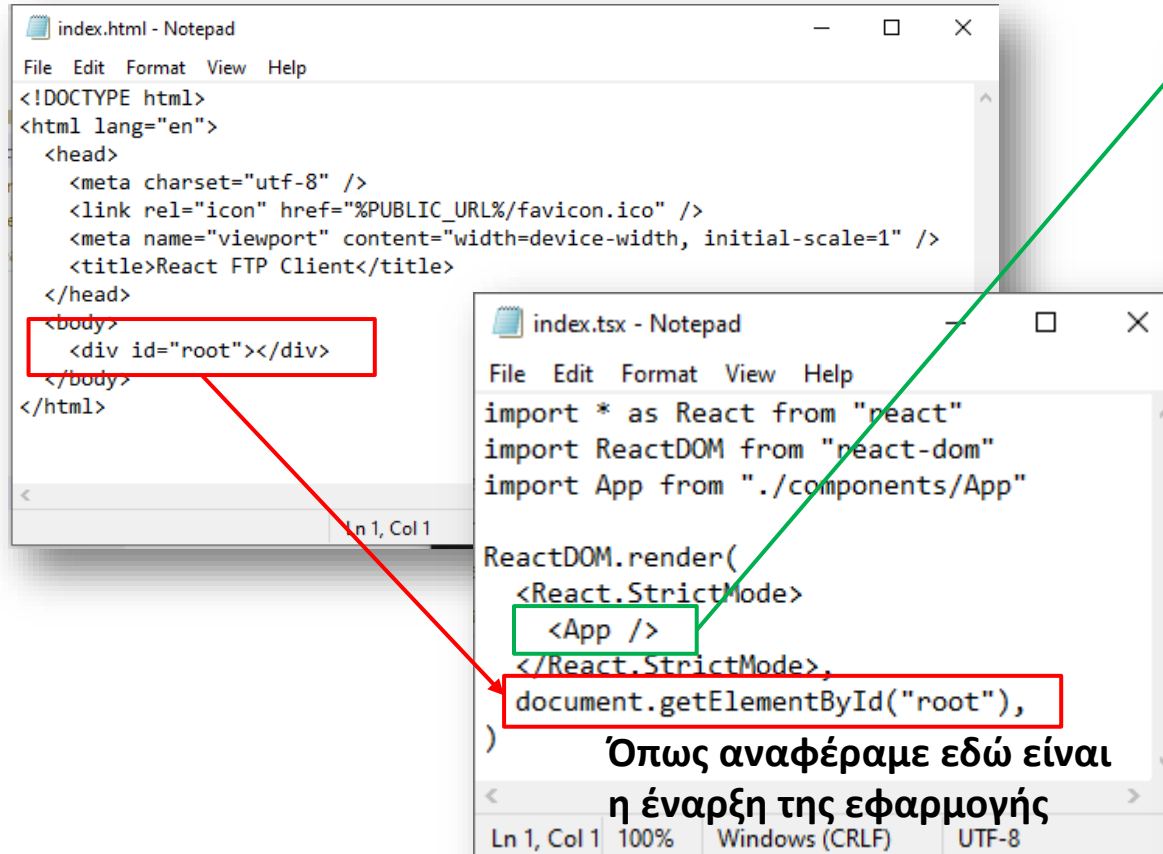
SERVER:
package.json

Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (3)



Η ΔΙΚΗ ΜΑΣ ΦΟΡΜΑ ΕΙΣΟΔΟΥ

Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (4)



The image shows two Notepad windows. The first window, titled 'index.html - Notepad', contains the following HTML code:

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <title>React FTP Client</title>
  </head>
  <body>
    <div id="root"></div>
  </body>
</html>

```

The second window, titled 'index.tsx - Notepad', contains the following TypeScript code:

```

import * as React from "react"
import ReactDOM from "react-dom"
import App from "../components/App"

ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById("root"),
)

```

Annotations in the image include:

- A red box around the `<div id="root"></div>` in the HTML file, with a red arrow pointing to the `document.getElementById("root")` in the TypeScript file.
- A green box around the `<App />` in the TypeScript file, with a green arrow pointing to the `function App(): JSX.Element {` in the adjacent code block.

Όπως αναφέραμε εδώ είναι
η έναρξη της εφαρμογής

```

function App(): JSX.Element {
  return (
    <IconContext.Provider value={{ size: "24px" }}>
      <ChakraProvider theme={theme}>
        <ColorModeScript />
        <ErrorBoundary FallbackComponent={ErrorFallback}>
          <I18nextProvider i18n={i18n}>
            <Box minW="100vw" minH="100vh">
              <React.Suspense
                fallback={
                  <AbsoluteCenter>
                    <Spinner description="Loading language.." />
                  </AbsoluteCenter>
                }
              >
                <Main />
              </React.Suspense>
            </Box>
          </I18nextProvider>
        </ErrorBoundary>
      </ChakraProvider>
    </IconContext.Provider>
  );
}

export default App;

```

Επιπλέον overlay
σε περίπτωση
σφάλματος

TOKEN EXCHANGE WITH BACK-END
Token sent if FTP connection successful

Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (5)

```
Main.tsx - Notepad
File Edit Format View Help
import * as React from "react";
import { useState } from "react";
import { getToken } from "../utils/authToken";
import { Authenticated } from "../Authenticated";
import { Unauthorized } from "../Unauthorized";

export function Main() {
  const [token, setToken] = useState();

  React.useEffect(() => {
    return setToken(getToken());
  }, []);

  return token ? <Authenticated /> : <Unauthorized />;
}
```

```
Unauthorized.tsx - Notepad
File Edit Format View Help
import * as React from "react";
import { Box } from "@chakra-ui/react";
import { Login } from "../views/Login";

export function Unauthorized( ) {
  return (
    <Box h="80vh" bg="blue.500">
      <Login />
    </Box>
  );
}
```

```
export function Login() {
  const { t } = useTranslation();
  const [, setToken] = useToken();
  const [submissionError, setSubmissionError] = React.useState<string>();
  const login = useLogin();
  const { register, handleSubmit, formState } = useForm<{
    hostname: string;
    username: string;
    password: string;
  }>({
    resolver: yupResolver(
      yup.object().shape({
        hostname: yup.string().label(t("hostname")).required(),
        username: yup.string().label(t("username")).required(),
        password: yup.string().label(t("password")).required(),
      })
    ),
  });
}
```

TOKEN EXCHANGE WITH BACK-END

Token sent if FTP connection successful

Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (6)

```
function useLogin() {
  const { t } = useTranslation();
  return React.useCallback(
    (hostname: string, username: string, password: string) => {
      const data = new URLSearchParams();
      data.append("hostname", hostname);
      data.append("username", username);
      data.append("password", password);

      return fetch("/login", {
        method: "POST",
        body: data,
      })
        .then((fetchResult) => {
          if (!fetchResult.ok) {
            console.error("Error logging user", fetchResult);
            return Promise.reject(t("wrong_hostname"));
          } else return fetchResult.json();
        })
        .then((token: Token) => {
          return token;
        });
    },
    [t]
  );
}
```

useLogin() - Hook

SERVER LOGIN REQUEST

```
index.js - Notepad
File Edit Format View Help
const express = require("express");
const bodyParser = require("body-parser");
const cors = require("cors");
const app = express();
const port = 5000;

app.use(cors());

app.use(bodyParser.json()); // support json encoded bodies
app.use(bodyParser.urlencoded({ extended: true })); // support encoded bodies

app.use("/login", (req, res) => {
  console.log("Received Request!");
  console.log(req.body);
  validUser(req.body) ? res.send({
    token: "test123",
  }) : res.send("Failure");
});

function validUser(req) {
  return req.username === "test" && req.password === "test";
}

app.listen(port, () => console.log(`Example app listening on port ${port}!`));
```

SERVER RESPONSE

TOKEN EXCHANGE WITH BACK-END

Token sent if FTP connection successful

Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (7)

Κατάλογος αρχείων του χρήστη στον FTP εξυπηρετητή

Name	Date modified	Type	Size
folder1	4/7/2022 5:51 PM	File folder	
onetextfile.txt	4/10/2022 9:31 PM	Text Document	2 KB
twotextfile.txt	4/10/2022 9:32 PM	Text Document	2 KB

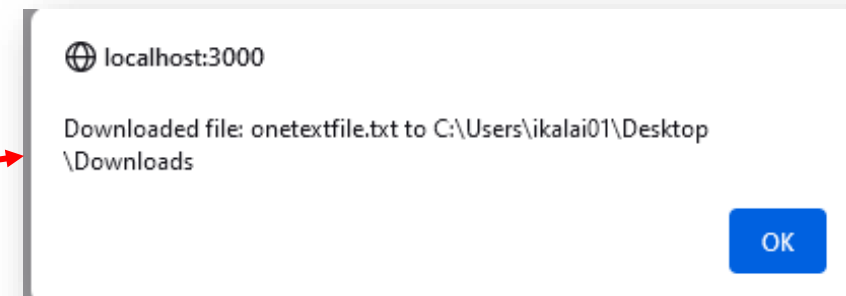
Εμφάνιση αρχείων στο γραφικό περιβάλλον
REACT FRON-END

Server: 127.0.0.1

File Name	File Type
..	
folder1	d
onetextfile.txt	-
twotextfile.txt	-

ΕΜΦΑΝΙΣΗ ΑΡΧΕΙΩΝ ΚΑΙ ΚΑΤΕΒΑΣΜΑ LIST FILES AND DOWNLOAD

Τα αρχεία κατεβαίνουν πατώντας πάνω στο
υφιστάμενο αρχείο.
Μας εμφανίζεται μήνυμα επιβεβαίωσης



Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (8)

```
app.use("/list", (req, res) => {
  console.log("Received Request!");
  path = req.body.path;
  c.connect({
    host: "127.0.0.1",
    user: "test",
    password: "test",
  });

  c.on("ready", function () {
    c.cwd(path, function () {});
    c.list(path, function (err, list) {
      if (err) throw err;
      c.end();
      if (res.headersSent) return;
      return res.send(list);
    });
  });
});
```

Nodejs backend

Post request to
node server to
get the data

```
export function Home() {
  const { t } = useTranslation();
  const [dataTable, setDataTable] = useState([]);

  useEffect(() => {
    const requestOptions = {
      method: "POST",
      headers: { "Content-Type": "application/json" },
      body: JSON.stringify({ path: "." }),
    };
    console.log("ddd");
    fetch("/list", requestOptions).then((response) =>
      response
        .json()
        .then((data) => ({
          data: data,
          status: response.status,
        })))
      .then((res) => {
        setDataTable(res.data);
      })
    );
  }, []);

  const column = [
    { heading: "File Name", value: "name" },
    { heading: "File Type", value: "type" },
  ];
```

Home Component

```
return (
  <>
    <Flex direction="column" justify="top" bg="blue.100">
      <NavbarComponent />
    </Flex>

    <Box bg={useColorModeValue("blue.200", "gray.900")} px={4}>
      <Text>Welcome to the {t("app_name")}</Text>
    </Box>
    <Spinner description={"Loading"} />
    <Box>
      <h1>Dynamic Table</h1>
      <Table data={dataTable} column={column} />
    </Box>
  </>
);
```

Render Component

```
Tablejs - Notepad
File Edit Format View Help
import "../Table.css";

let path = "";

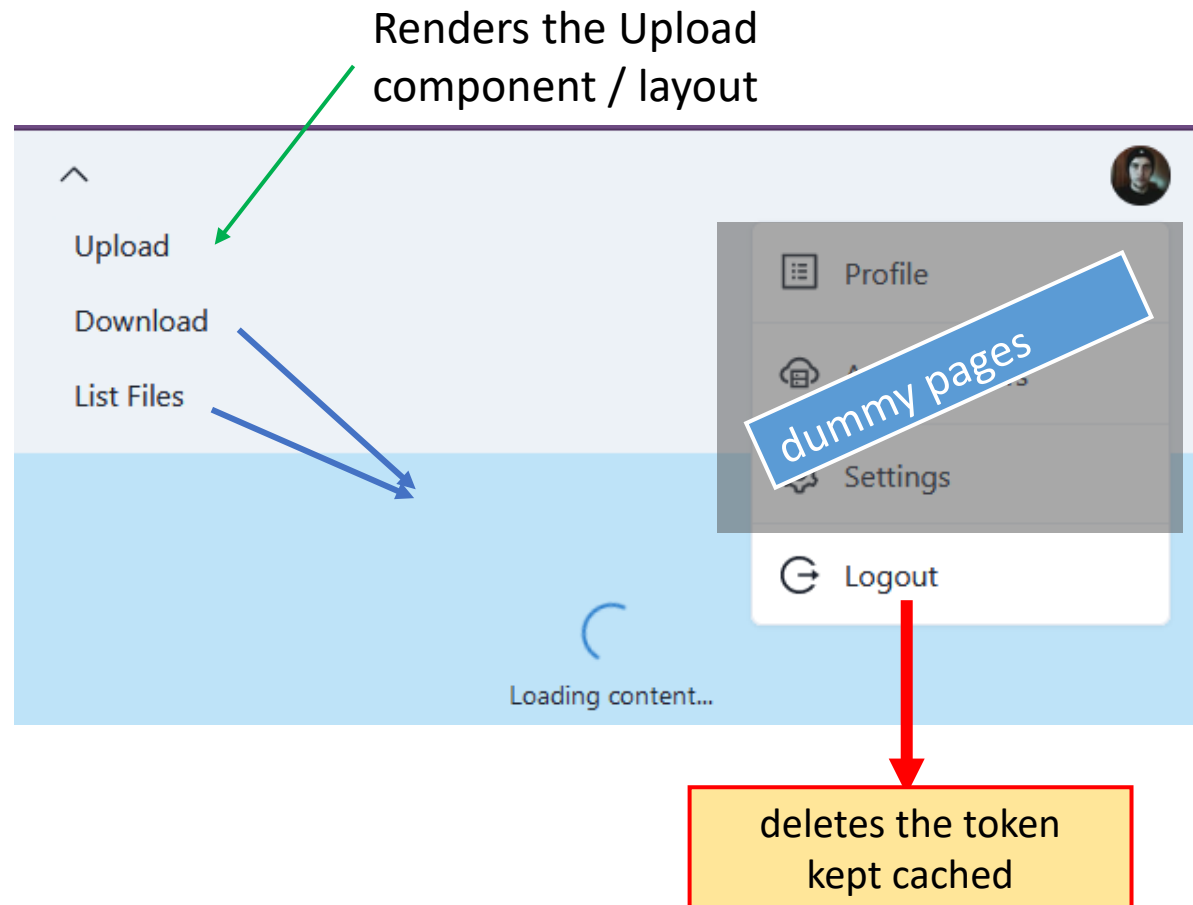
const Table = ({ data, column }) => {
  return (
    <table>
      <td onClick={() => goBack()}>..</td>
      <thead>
        <tr>
          {column.map((item, index) => (
            <TableHeadItem item={item} />
          ))}
        </tr>
      </thead>
      <tbody>
        {data.map((item, index) => (
          <TableRow item={item} column={column} />
        ))}
      </tbody>
    </table>
  );
};

const TableHeadItem = ({ item }) => <th>{item.heading}</th>;
```

Table Component

Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (9)

HOME Component Rendered



Η ΔΙΚΗ ΜΑΣ ΥΛΟΠΟΙΗΣΗ ΤΟΥ FTP CLIENT (10)

Upload Component Rendered



Ανοίγει pop-up για επιλογή
αρχείου / φακέλου προς
ανέβασμα

File Uploaded

test1.png

Browse

100%

Upload

ΣΥΜΠΕΡΑΣΜΑΤΑ – ΘΕΤΙΚΑ ΚΑΙ ΑΡΝΗΤΙΚΑ ΑΠΟ ΤΗΝ ΔΙΚΗ ΜΑΣ ΕΜΠΕΙΡΙΑ

ΘΕΤΙΚΑ

- Μεγάλο όγκο διαθέσιμων παραδειγμάτων στο διαδίκτυο.
- Μεγάλο όγκο διαθέσιμων tutorials στο YouTube, Udemy, Coursera, κ.α.
- Εύκολη εγκατάσταση εργαλείων που χρειαστήκαμε.
- Εύκολη πρόσθεση νέων packages και εξαρτήσεων
- Δουλεύει στους πλείστους φυλλομετρητές.
- Εύκολη δημιουργία πρώτου project,
`npm create-react-app <app-name>`
- Επαναχρησιμοποίηση έτοιμων στοιχείων (components).
- Διαδραστική αίσθηση πλοήγησης.
- Ικανοποιητικός χρόνος μεταγλώττισης και έναρξης εφαρμογής.
- Άμεση ανταπόκριση στις ενέργειες του χρήστη.

ΑΡΝΗΤΙΚΑ

- Μπορεί να είναι δύσκολη στην μάθηση για κάποιον που δεν ασχολείται ιδιαίτερα με το web-development.
- Προαπαιτείτε γνώσης τεχνολογιών και εννοιών HTML, CSS, JavaScript, JSX, WEB, Document DOM, κ.α.
- Πληθώρα από επιλογές διάφορων packages για ενσωμάτωση στην εφαρμογή.
- Τα αρχεία node_modules καταναλώνουν μεγάλο χώρο αποθήκευσης.

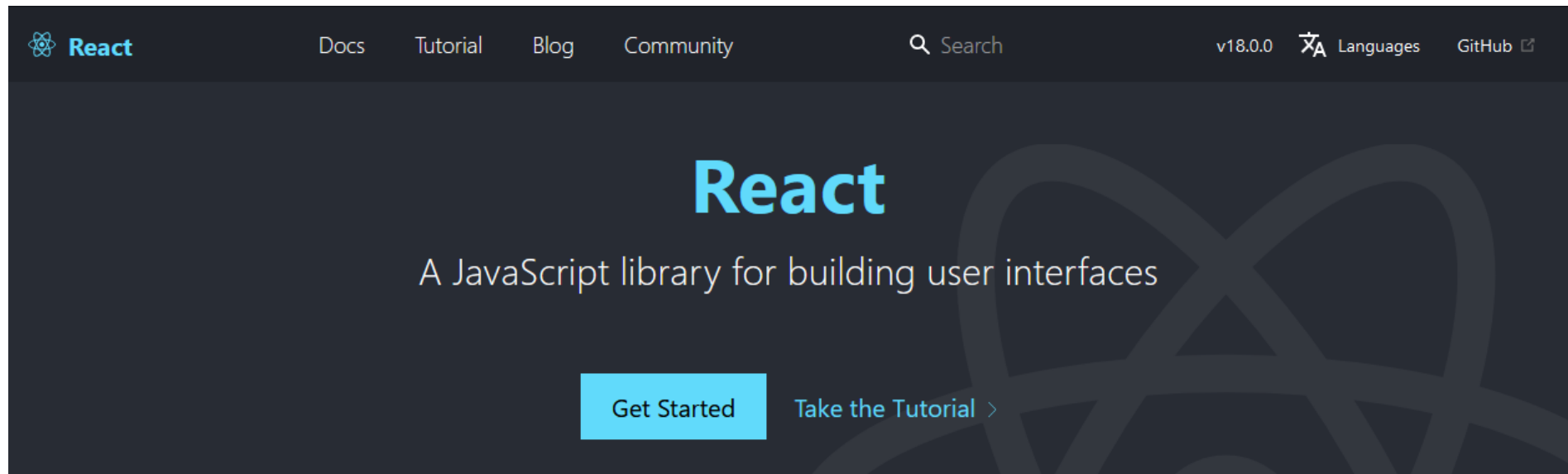
ΕΝΑΛΛΑΚΤΙΚΕΣ ΕΠΙΛΟΓΕΣ



ΧΡΗΣΙΜΕΣ ΠΛΗΡΟΦΟΡΙΕΣ

Documentation at: <https://reactjs.org/>

Source code at: <https://github.com/facebook/react/>



PROJECT SOURCE CODE AT:

<https://github.com/icanci01/epl421-presentation-react>

ΒΙΒΛΙΟΓΡΑΦΙΑ

- <https://el.wikipedia.org/wiki/%CE%94%CE%B9%CE%B1%CE%B4%CE%AF%CE%BA%CF%84%CF%85%CE%BF> - (Διαδίκτυο)
- <https://en.wikipedia.org/wiki/React> - (Wikipedia - React)
- <https://www.universalclass.com/articles/computrs/javascript/how-to-script-forms-with-javascript.htm>
- <https://reactjs.org/> - React (Presentation Topic)
- <https://v5.reactrouter.com/web/api/> - React Router (Path manager, context switching)
- <https://nodejs.org/en/> - Node (Working Environment)
- <https://www.npmjs.com/> - NPM (Node Package Manager)
- <https://expressjs.com/> - Express (Back-end Integration)
- <https://yarnpkg.com/> - Yarn (Package Manager)
- <https://filezilla-project.org/> - FileZilla (FTP Server Provider)
- <https://chakra-ui.com/> - Chakra-UI (Front-end Components)
- <https://react-bootstrap.github.io/getting-started/introduction/> - Bootstrap (Front-end CSS)
- <https://www.w3schools.com/react/> - W3Schools (React Tutorials)
- <https://www.npmjs.com/package/basic-ftp> - FTP API Provider (Connection to FileZilla)
- <https://www.youtube.com/watch?v=b6Oe2puTdMQ> – UPLOAD
- <https://paypalpaul2436.medium.com/10-main-core-concept-you-need-to-know-about-react-303e986e1763> – Extra Information

ΕΥΧΑΡΙΣΤΟΥΜΕ

ΕΡΩΤΗΣΕΙΣ;