

React + Ts

Loops & Conditional Rendering

```
import Son from "./Son";

const Father = () => {
  return (
    <>
      <h1>Father</h1>
      <Son name="Joe" age={25} />
      <Son name="Bob" age={30} />
      <Son name="Luci" age={35} />
    </>
  );
};

export default Father;
```

Conditional Rendering

- מהו רינדור מותנה (Conditional Rendering)?
- במה זה מסייע לנו?
- כיצד נוכל לקבוע Style לאלמנט בהתאם לתנאי נתון?
- כיצד נוכל לקבוע האם להציג אלמנט בהתאם לתנאי נתון?
- באיזה מקרים ternary יכול לסייע לנו?

```
const Son = (props: SonProps) => {
  const { name, age } = props;

  const titleColor: string = age > 30 ? "red" : "blue";
  const isBiggerThen30: boolean = age > 30;

  return (
    <>
      <h1 style={{ color: titleColor }}>{name}</h1>
      {isBiggerThen30 && (
        <h3>
          hello, my name is {name}, my age is {age}
        </h3>
      )}
    </>
  );
};
```

Father

Joe

Bob

Luci

hello, my name is Luci, my age is 35

תרגול - Conditional Rendering

צרו פרויקט React חדש לטובת התרגיל.

```
import RandomNum from "../components/RandomNum";

const App = () => {
  return (
    <main>
      <RandomNum min={1} max={10} />
      <RandomNum min={5} max={100} />
      <RandomNum min={15} max={20} />
      <RandomNum min={30} max={60} />
    </main>
  );
};

export default App;
```

Random Number
this is Odd Number

Random Number
100

Random Number
18

Random Number
this is Odd Number

- צרו קומפוננטה חדשה בשם RandomNum.
- קומפוננטה זו תדע לקבל ב-props 2 פרמטרים: min, max.
- בקומפוננטה יוגרל מספר **שלם** רנדומלי בין הטווחים שהתקבלו ב-props.
- במידה והמספר שהוגרל מתחלק ב-2 – המספר המתקבל יוצג על גבי המסך (יש להשתמש במודולו %), אחרת, תוצג הודעה שהמספר אינו מתחלק ב-2.
- קיראו לקומפוננטה זו להצגה 4 פעמים באמצעות שליחת ערכים שונים ובידקו האם אכן מוצגים לכם רק מספרים זוגיים (מספרים שמתחלקים ב-2).

```
export const users = [
  {
    name: 'Bob',
    age: 25,
    email: 'bob@email.com'
  },
  {
    name: 'Joe',
    age: 35,
    email: 'joe@email.com'
  },
  {
    name: 'Luci',
    age: 45,
    email: 'luci@email.com'
  }
];
```

```
const LoopsDemo = () => {
  return (
    <div>
      <h1>Loop Demo</h1>
      <div>
        {users.map((user, index) => {
          return (
            <div key={index}>
              <h3>{user.name}</h3>
              <p>{user.email}</p>
              <p>age: {user.age}</p>
            </div>
          );
        })}
      </div>
    </div>
  );
};

export default LoopsDemo;
```

לולאות ו-map

- כיצד נוכל להשתמש בלולאה על מנת להציג רצף של אותה הקומפוננטה כמספר הריצות של הלולאה?
- במה המתודה map יכולה לעזור לנו בנושא?
- כיצד נוכל לתת מזהה ייחודי לכל איבר במערך?
- מדוע עלינו לעשות זאת?
- כיצד נוכל להשתמש בפרמטר ה-index של פונקציית ה-callback לצורך זה?

Loop Demo

Bob

bob@email.com

age: 25

Joe

joe@email.com

age: 35

Luci

luci@email.com

age: 45

Loops - תרגול

```
import { users } from "../data/users";
import Card from "../Card";
import Styles from "../LoopDemo.module.css";

const LoopsDemo = () => {
  return (
    <div className={Styles.container}>
      <h1 className={Styles.header}>Loop Demo</h1>
      <div className={Styles.container}>
        {users.map((user, index) => {
          return <Card key={index} user={user} className={Styles.card} />;
        })}
      </div>
    </div>
  );
};

export default LoopsDemo;
```

- העתיקו והריצו את הדוגמא מהשקף הקודם.
- הפרידו את תוכן ה-div אשר מוצג בלולאה לקומפוננטה נפרדת בשם Card והעבירו לה את פרטי המשתמש דרך ה-props.
- תנו לקומפוננטה עיצוב דומה ככל הניתן לדוגמא המוצגת.
- הוסיפו עוד 3 משתמשים לרשימה.
- הציגו רק את המשתמשים שגילם הוא **אי-זוגי**.
- הוסיפו כפתור ל-Card.
- בעת לחיצה על הכפתור שם המשתמש יודפס לקונסול.

Loop Demo

Bob bob@email.com age: 25	Joe joe@email.com age: 35	Luci luci@email.com age: 45
--	--	--

שיעורי בית

- יש לחזור ולסיים לענות על כל התרגילים שבמצגת.

תודה על ההקשבה

אני וצוות המכללה כאן עבורכם