

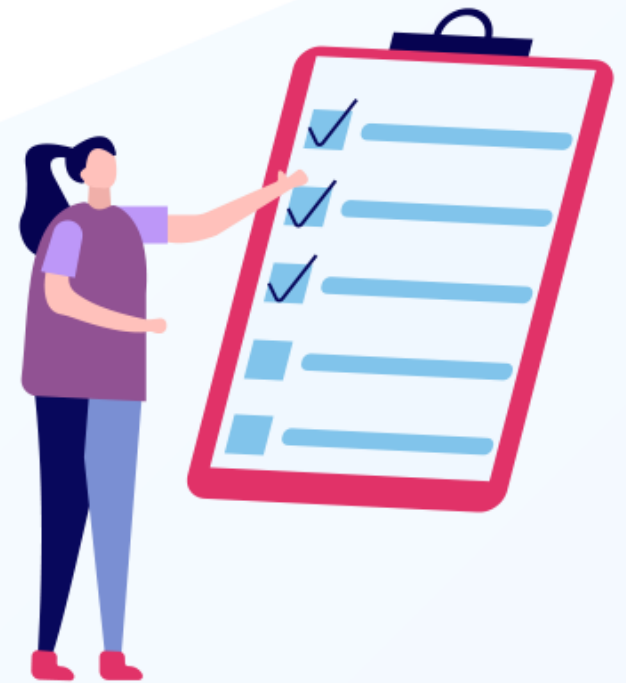
# Basics of Next.js with React



# Learning Objectives

By the end of this lesson, you will be able to:

- 👁️ Identify the features and benefits of Next.js for building efficient web applications with React
- 👁️ Create Next.js welcome and login applications
- 👁️ Outline server-side rendering (SSR) and demonstrate routing in Next.js
- 👁️ Create a Rest API using Next.js





# Introduction to Next.js

# What Is Next.js?

It is a React framework for building full-stack web applications.

# NEXT.js

- It abstracts and automatically configures tooling needed for React, like bundling, compiling, and more.
- It helps in building interactive, dynamic, and fast React applications.

It is known for providing the best developer experience when building production-ready applications with all the required features.

# Key Features of Next.js

Data fetching

Built-in Cascading  
Style Sheets (CSS)

Server-side  
rendering (SSR)

Dynamic routing

TypeScript support

Image optimization

Layouts

Automatic code  
splitting

Fast refresh

# Key Features of Next.js

## Data fetching

- It provides developers with various methods to fetch and render data in their applications.
- It aligns with different rendering strategies like server-side rendering (SSR), static site generation (SSG), and client-side rendering (CSR).

## Built-in Cascading Style Sheets (CSS)

Next.js supports importing CSS files directly into JavaScript or React components.

# Key Features of Next.js

## Server-side rendering (SSR)

Next.js allows pages to be rendered on the server side, thereby improving performance and SEO.

## Dynamic routing

It is a file-system-based routing mechanism where pages get automatically routed based on their file names in the pages directory.

## TypeScript support

It is a built-in support that makes it easier to build robust applications with type-checking.

# Key Features of Next.js

## Image optimization

This helps to resize, optimize, and serve images in modern formats like WebP.

## Layouts

- It allows developers to create reusable layout components that can wrap around pages or groups of pages.
  - This is particularly useful for maintaining consistent design elements like headers, footers, and sidebars across different pages.



# Key Features of Next.js

## Fast refresh

- It provides instant feedback on edits made to React components.
- It preserves the state of the components while updating the code, which enhances the development experience.

## Automatic code splitting

- This feature enhances application performance by splitting the JavaScript code into smaller chunks.
- This process happens at the page level; each page in the pages directory gets its own JavaScript bundle.
  - This approach optimizes the initial load of the application, as users download only the code they need for the page they are visiting, rather than the entire application's code.

# Benefits of Next.js

Improved performance

Better SEO

Enhanced user experience

Simplified routing

Full stack capabilities

Ease of deployment

Flexibility with rendering techniques

Scalability

Rich ecosystem

Community and support

Developer experience

Zero configuration startup

# Creating Next.js Welcome Application



## Problem Statement:

**Duration: 15 min**

You have been assigned a task to create the Next.js application to display a simple welcome message.

# Assisted Practice: Guidelines



Steps to be followed:

1. Set up the next.js project
2. Create the Project
3. Run the application



## Server-Side Rendering (SSR) in Next.js

# SSR in Next.js

It refers to the process of rendering React components on the server to generate HTML for a page request, which is then sent to the client's browser.

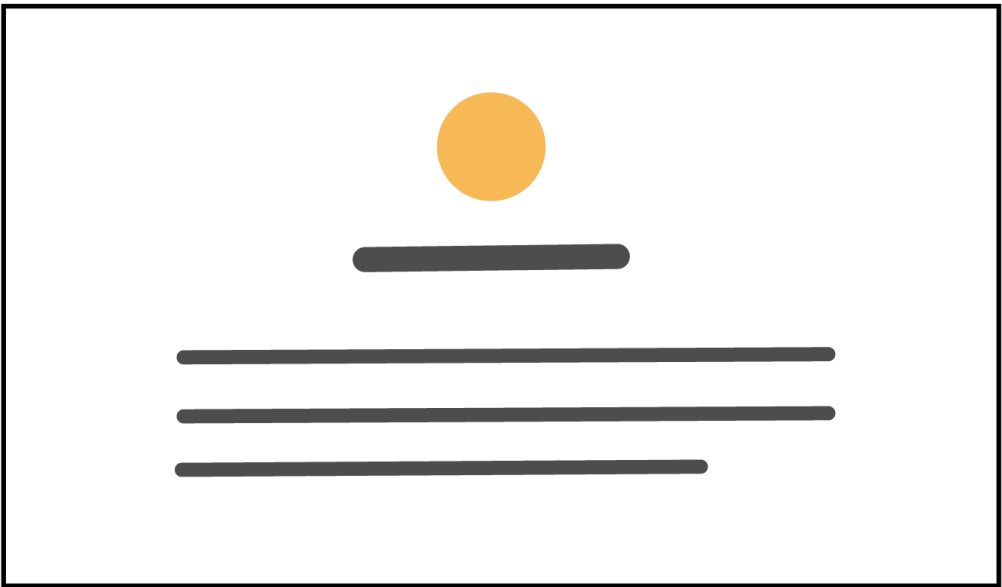
## Static generation

The HTML is generated at *built-time* and reused for each request.

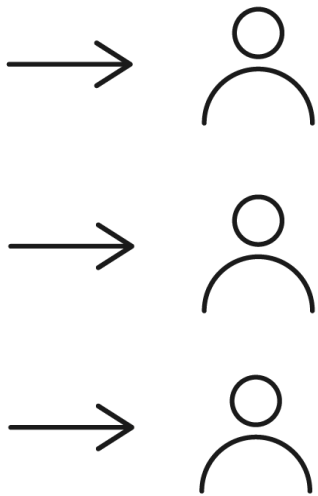
**NEXT**.JS

Next, build →

Build the app for production



The HTML is generated.



Reused for each request

# Create a React Next.js Login Application



## Problem Statement:

**Duration: 15 min**

You have been assigned a task to create the Next.js application by creating server-side components to do login with file and testing with the React component.

# Assisted Practice: Guidelines



Steps to be followed:

1. Create and set up the Next.js project
2. Create a file that contains a set of login details
3. Create a server folder that holds a server-side component
4. Create a client folder
5. Run the application





# Routing in Next.js

# Routing in Next.js

It refers to the process of managing navigation between different pages in a web application based on URL paths.

## Types of routes:

### Index routes

The index.js files in the pages directory get automatically routed to the root directory.

### Nested routes

It involves organizing files into subdirectories within the pages directory and enables the creation of nested URL structures.

### Dynamic routes

It uses square brackets (`[ ]`) to indicate dynamic segments in the URL.

# Creating Next.js Routing Application



## Problem Statement:

**Duration: 15 min**

You have been assigned a task to create the Next.js server-side routing application based upon the path provided by browser.

# Assisted Practice: Guidelines



Steps to be followed:

1. Create and set up a Next.js project
2. Create components for routing
3. Run the application

# Creating a Rest API Using Next.js



## Problem Statement:

**Duration: 15 min**

You have been assigned a task to create a Next.js server-side component for a REST API that interacts with a file module for efficient data storage and retrieval, accessible across various technologies.

# Assisted Practice: Guidelines

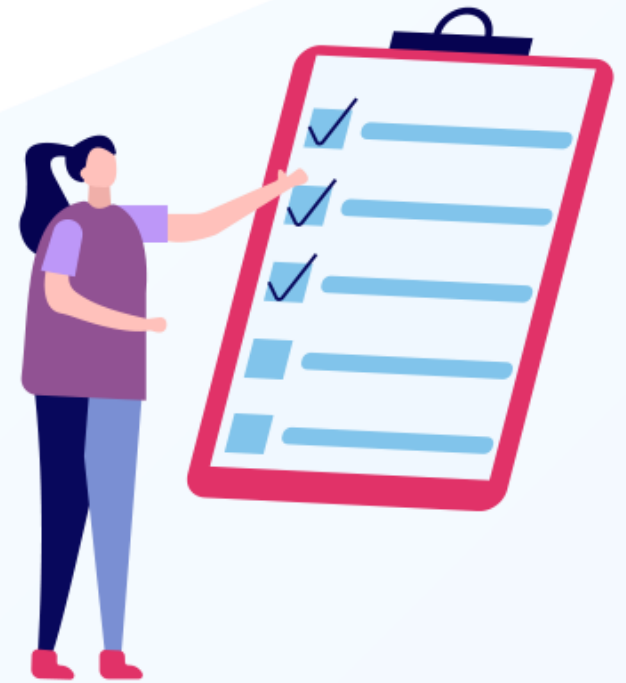


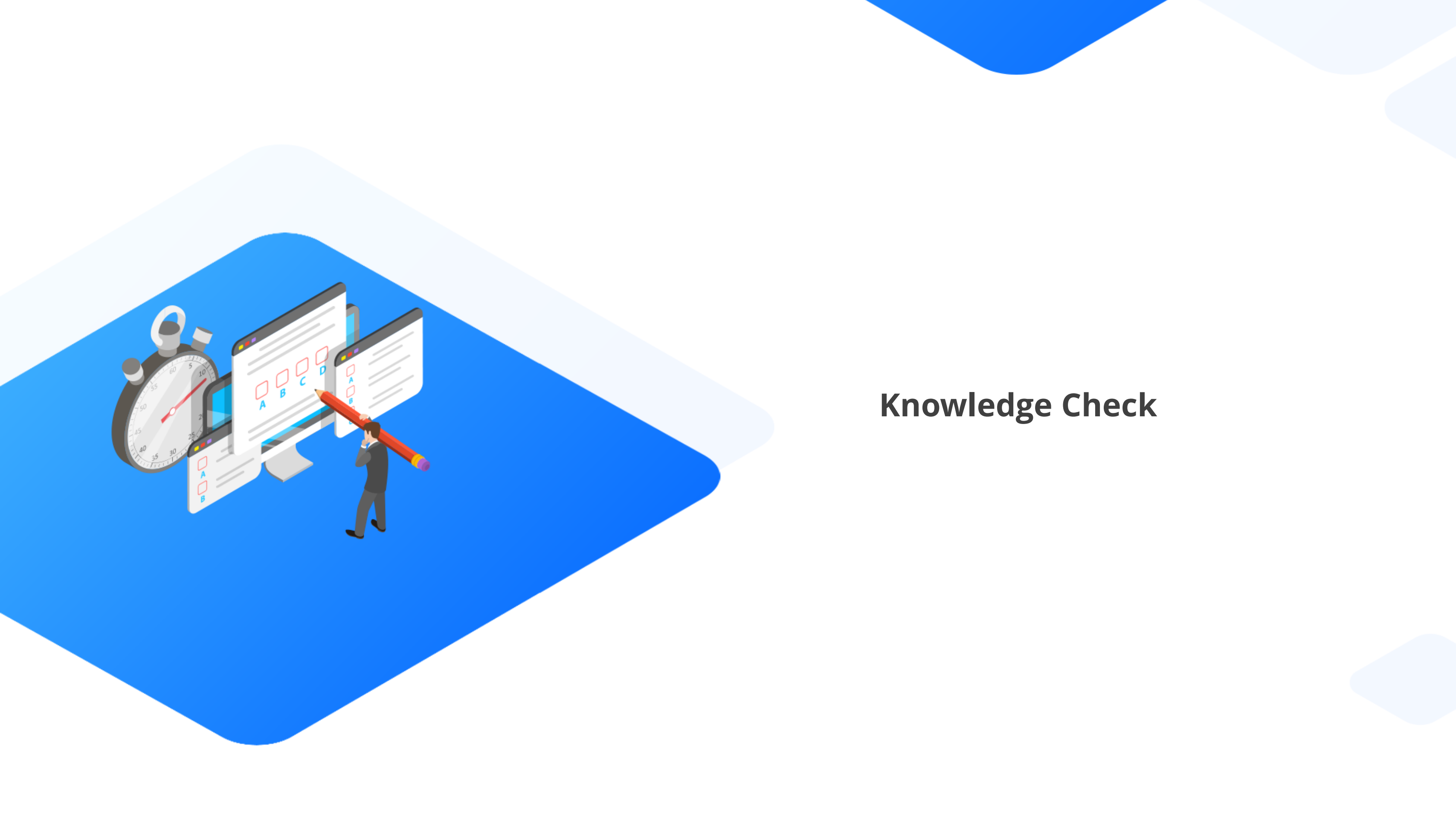
Steps to be followed:

1. Create and set up a Next.js project
2. Create a JSON file for an employee
3. Create a routing component
4. Run the application

# Key Takeaways

- Next.js is a React framework for building full-stack web applications as well as interactive, dynamic, and fast React applications.
- SSR in Next.js refers to rendering React components on the server to generate HTML for a page request, which is then sent to the client's browser.
- Routing in Next.js refers to the process of managing navigation between different pages in a web application based on URL paths.





## Knowledge Check



## Knowledge Check

1

Which among the following is a built-in support that makes it easier to build robust applications with type-checking?

- A. Server-side rendering
- B. Fast refresh
- C. TypeScript
- D. Data fetching



## Knowledge Check

1

Which among the following is a built-in support that makes it easier to build robust applications with type-checking?

- A. Server-side rendering
- B. Fast refresh
- C. TypeScript
- D. Data fetching



---

The correct answer is **C**

---

**TypeScript is a built-in support that makes it easier to build robust applications with type-checking.**