

## React Fiber nədir?

- React Fiber, React kitabxanasının arxitekturasında edilən böyük bir yeniləmədir və əsasən performansın yaxşılaşdırılması və daha çevik renderləmə prosesini təmin etmək məqsədi daşıyır. Fiber, React 16 versiyasından etibarən təqdim edilmişdir və React-in əvvəlki arxitekturası olan "Stack Reconciler"-i əvəz edir.

## Niyə React Fiber və üstünlükləri nədir?

- **Asinxron və Prioritetli Renderləmə:**

Fiber asinxron renderləməni dəstəkləyir, yeni uzun sürən renderləmə işləri zamanı tətbiqin donmasının qarşısını alır. Bu, komponentlərin renderlənməsi prosesini kiçik hissələrə bölərək hər birinə fərqli prioritetlər təyin edir. Daha vacib işlər (məsələn, istifadəçi ilə qarşılıqlı əlaqə) daha sürətli işlənir, az vacib işlər isə sonraya saxlanır.

- **UI Performansının Artırılması:**

Fiber UI-nin performansını artırmaq üçün optimallaşdırılıb. Bu, xüsusilə mürəkkəb və böyük tətbiqlərdə, çoxlu yenidən renderləmə əməliyyatları baş verən yerlərdə önə çıxır. Fiber istifadəçilərin klikləri və skrolları kimi interaktivləri daha yüksək prioritetdə saxlayır, beləliklə UI-nin axıcı qalmasına kömək edir.

- **Geri çəkile bilən (Interruptible) Renderləmə:**

Fiber əvvəlki arxitektura ilə müqayisədə daha çevikdir və renderləmə prosesini yarıda kəsmək (interruption) imkanı verir. Əgər daha vacib bir iş (məsələn, istifadəçinin girişi) ortaya çıxarsa, mövcud iş yarıda dayandırılabilir və sonra davam etdirilə bilər. Bu, komponentlər arasında keçidlərin daha axıcı və rahat olmasına şərait yaradır.

- **Geri Qayıtmaq və Yenidən Renderləmək:**

Fiber ilə müəyyən renderləmə işləri ləğv edilə bilər və ya əvvəlki vəziyyətə qaytarılabilir. Bu, UI vəziyyətlərinin idarə olunmasını daha asan edir və tətbiqlərdə daha böyük nəzarət imkanı verir.

- **Yenidən İstifadə Edilə bilən Kod:**

Fiber, React komponentlərinin daha çevik olmasını təmin edir. Yəni bir komponentin render prosesi yarıda kəsilərək daha sonra davam etdirilə bilər, kod təkrar istifadə oluna bilər və sistemin resursları daha optimal şəkildə idarə olunur.

- **Daha Yaxşı Animasiya Dəstəyi:**

Fiber ilə animasiyalar daha səmərəli işləyir, çünki yüksək prioritetli interaktiv əməliyyatlar (məsələn, animasiyalar) gecikmədən icra edilir. Bu da istifadəçi təcrübəsini daha axıcı və mükəmməl edir.

- **Daha Yaxşı Genişləndirilə bilənlik (Scalability):**

Fiber, böyük və mürəkkəb tətbiqləri daha yaxşı idarə edir və miqyaslanabilir bir renderləmə modelinə malikdir. React Fiber-in çevik təbiəti onun mürəkkəb UI-ləri dəstəkləməsini asanlaşdırır və tətbiqlərin böyüməsinə mane olmur.

- **Daha Ağıllı Hata İdarəetməsi:**

Fiber React-in hata idarəetmə modelini də təkmilləşdirir. Komponentlərdəki hatalar daha yaxşı idarə olunur və müəyyən bir komponentdə baş verən hata bütün tətbiqi dayandırmır.

## Polymorphism nədir?

- Polymorphism (Çoxşəkillilik) obyekt yönümlü proqramlaşdırmanın (OOP) mühüm prinsiplərindən biridir və fərqli obyektlərin eyni interfeysi istifadə edərək müxtəlif yollarla davranmasını təmin edir. React isə JavaScript ilə yazılan və komponent əsaslı bir istifadəçi interfeysi kitabxanasıdır. Polymorphism, React-in özündə birbaşa təqdim edilməsə də, React komponentləri ilə işləyərkən istifadə olunan mühüm prinsiplərdən biri kimi qəbul edilə bilər.

## Niyə polymorphism istifadə olunmalıdır və üstünlükləri nədir?

- React-də polymorphism anlayışını əsasən komponentlər və onların istifadəsi vasitəsilə görürük. Polymorphism React-də aşağıdakı yollarla təzahür edir:

- **Fərqli Komponentlərin Eyni Interfeysdən İstifadəsi:**

Polymorphism-in əsas prinsiplərindən biri bir neçə komponentin eyni interfeysi tətbiq etməsi və fərqli davranmasıdır. React-də bu, daha çox təkrarlanan komponent strukturlarında və ya dinamik olaraq bir komponentdən digərinə keçərkən olur.

- **Komponentlər Arasında Təkrar İstifadə və Fərqli Funksionallıq:**

Polymorphism React komponentlərinin müxtəlif yerlərdə təkrar istifadəsini asanlaşdırır. Müxtəlif komponentlər eyni bazanı (məsələn, abstract component) paylaşaraq özünəməxsus davranışlarını təmin edə bilərlər. Məsələn, bir neçə düymə növü üçün bir əsas düymə komponenti yaradıb, onu fərqli məqsədlər üçün istifadə edə bilərsiniz.

- **Render Props və Higher-Order Components (HOC):**

Polymorphism anlayışı React-də daha yüksək səviyyədə istifadə oluna bilər. Render props və Higher-Order Components (HOC) kimi üsullar React-də polymorphism prinsiplərini əks etdirir. Bu üsullar komponentlərin funksionallığını təkrar istifadə etməyə və eyni interfeys üzərindən fərqli davranışlar təmin etməyə imkan verir.

- **Children Props:**

React komponentlərində children props-u vasitəsilə polymorphism təmin edilə bilər. Fərqli komponentlər eyni children props-u ilə müxtəlif şəkildə davranır.

- **Flexible və Adaptiv Tətbiqlər:** Polymorphism React-də komponentləri daha çevik edir. Eyni bazadan istifadə edərək, tətbiqin müxtəlif hissələrində fərqli davranış göstərən komponentlər hazırlamaq mümkündür.

## **CDN nədir?**

- CDN (Content Delivery Network), məzmun çatdırılma şəbəkəsi olaraq tərcümə olunur. Bu, coğrafi olaraq müxtəlif yerlərdə yerləşdirilən serverlərdən ibarət bir şəbəkədir və internet istifadəçilərinə veb saytlar, videolar, şəkillər, JavaScript faylları kimi məzmunu daha sürətli və etibarlı şəkildə çatdırmaq üçün istifadə olunur.

## **Niyə CDN istifadə olunmalıdır və üstünlükləri nədir?**

- **Yükləmə Sürətinin Artırılması:**

CDN-lər məzmunu istifadəçilərə ən yaxın olan serverlərdən çatdırdığı üçün yükləmə sürətini xeyli artırır. Bu, xüsusilə böyük media faylları, şəkillər və videolar üçün vacibdir, çünki coğrafi məsafə yükləmə vaxtına birbaşa təsir edir. Məzmun daha yaxın bir serverdən gəldiyindən gecikmə azalır və sayt daha sürətli yüklənir.

- **Daha Yaxşı İstifadəçi Təcrübəsi:**

Daha sürətli yüklənən veb saytlar və tətbiqlər istifadəçilərə daha rahat və problemsiz bir təcrübə təmin edir. Bu, ziyarətçilərin saytınızda daha uzun müddət qalmasına və yenidən geri dönməsinə səbəb olur.

- **Beynəlxalq İstifadəçilərə Daha Yaxşı Xidmət:**

Əgər saytınızın və ya tətbiqinizin beynəlxalq istifadəçiləri varsa, CDN bu istifadəçilərin daha sürətli xidmət almasını təmin edir. Məzmun dünya üzrə müxtəlif yerlərdə yerləşən serverlərdən təqdim olunduğu üçün hər yerdə istifadəçi təcrübəsi bərabər səviyyədə olur.

- **Ana Serverə Düşən Yükün Azaldılması:**

CDN-lər, məzmunu server şəbəkələri arasında yaydığı üçün ana serverə düşən yükü azaldır. Bu, serverin həddindən artıq yüklənməsinin qarşısını alır və veb saytın daha stabil işləməsinə kömək edir.

- **Trafik Partlayışları Zamanı Performansın Yüksəldilməsi:**

Əgər veb saytınızda ani trafik artımları (məsələn, yeni məhsul buraxılışı zamanı) baş verərsə, CDN bu trafiki yayaraq serverin işini dəstəkləyir. Beləliklə, saytınız çoxlu ziyarətçi gələndə belə çökməz və sürətli qalmağa davam edir.

- **Etibarlılıq və Yedəkləmə:**

Məzmun bir neçə serverdə saxlanıldığından, bir serverdə nasazlıq baş verdikdə digər serverlərdən xidmət davam edir. Bu da veb saytınızın və ya tətbiqinizin həmişə əlçatan olmasını təmin edir.

- **SEO Üçün Müsbət Təsir:**

Axtarış motorları (məsələn, Google) sürətli yüklənən saytları daha yüksək qiymətləndirir. CDN istifadə etməklə veb saytınızın

- yüklənmə sürətini artırdığınız üçün SEO baxımından da müsbət təsir göstərmiş olursunuz. Sürətli saytlar daha yüksək sıralamalara malik ola bilər.
- **DDOS Hücumlərinə Qarşı Müdafiə:**  
CDN-lər şəbəkələrində güclü təhlükəsizlik protokolları saxlayır və DDOS (Distributed Denial of Service) hücumlarını süzgəcdən keçirə bilirlər. Bu, saytınızı zərərli hücumlardan qoruyur və işləmə müddətinin pozulmasının qarşısını alır.