Pagination

সহজ কথায়, **Pagination (প্যাগিনেশন)** হলো একটি বিশাল তথ্য বা ডাটার তালিকাকে ছোট ছোট অনেকগুলো ভাগে বা পাতায় (Pages) ভাগ করে দেখানো।

আপনি যখন Google-এ কিছু সার্চ করেন, তখন নিচে **1, 2, 3, 4, Next** লেখা থাকে—এটাই হলো প্যাগিনেশন। গুগল একসাথে কোটি কোটি রেজাল্ট এক পাতায় না দেখিয়ে প্রতি পাতায় ১০-২০টি করে রেজাল্ট দেখায়।

১. প্যাগিনেশন কেন প্রয়োজন?

- **Performance (গতি):** আপনার ডাটাবেসে যদি ১ লাখ প্রোডাক্ট থাকে, আর আপনি যদি একবারে সব লোড করতে চান, তবে আপনার ওয়েবসাইট ক্রাশ করবে বা অনেক সময় নেবে। প্যাগিনেশন করলে ওয়েবসাইট দ্রুত লোড হয়।
- **User Experience (ব্যবহারকারীর অভিজ্ঞতা):** একসাথে অনেক তথ্য থাকলে মানুষ কনফিউজ হয়ে যায়। অল্প অল্প তথ্য দেখালে সেটি পড়তে বা বুঝতে সুবিধা হয়।
- **Server Load কমানো:** প্রতিবার সার্ভার থেকে মাত্র ১০ বা ২০টি ডাটা আনতে হয় বলে সার্ভারের ওপর চাপ অনেক কম থাকে।

২. প্যাগিনেশন কীভাবে কাজ করে? (Technical Logic)

প্যাগিনেশন কাজ করার জন্য মূলত দুটি প্যারামিটার বা মান প্রয়োজন হয়:

1. **Limit (বা Page Size):** প্রতি পাতায় কয়টি আইটেম থাকবে? (যেমন: ১০টি)।
2. **Offset (বা Page Number):** কত নাম্বার আইটেম থেকে দেখানো শুরু হবে?

উদাহরণ: ধরুন আপনি প্রতি পাতায় ১০টি করে ডাটা দেখাতে চান।

- **Page 1:** ১ থেকে ১০ পর্যন্ত ডাটা। (Offset = 0, Limit = 10)

- **Page 2:** ১১ থেকে ২০ পর্যন্ত ডাটা। (Offset = 10, Limit = 10)
- **Page 3:** ২১ থেকে ৩০ পর্যন্ত ডাটা। (Offset = 20, Limit = 10)

৩. প্যাগিনেশনের ধরন (Types of Pagination)

ক. Offset-based Pagination (সবচেয়ে কমন)

এটি ডাটাবেসের LIMIT এবং OFFSET ব্যবহার করে কাজ করে।

- **সুবিধা:** এটি তৈরি করা খুব সহজ।
- **অসুবিধা:** যদি ডাটাবেস খুব বড় হয় (কোটি কোটি ডাটা), তবে শেষের দিকের পাতাগুলো লোড হতে অনেক সময় লাগে।

খ. Cursor-based Pagination (আধুনিক পদ্ধতি)

ফেসবুক বা ইনস্টাগ্রামের নিউজফিড স্ক্রল করলে যে নতুন পোস্ট আসে, তা এই পদ্ধতিতে কাজ করে। এখানে কোনো পেজ নাম্বার থাকে না, বরং শেষ আইটেমের একটি **ID** বা **Timestamp** (যাকে Cursor বলে) ব্যবহার করে পরের ডাটাগুলো আনা হয়।

- **সুবিধা:** এটি অনেক দ্রুত এবং ডাটা ডুপ্লিকেট হওয়ার ভয় থাকে না।
- **অসুবিধা:** সরাসরি ৩ বা ৪ নাম্বার পেজে লাফ দিয়ে যাওয়া যায় না।

৪. প্যাগিনেশন বনাম ইনফিনিট স্ক্রলিং (Infinite Scrolling)

- **Pagination:** যেখানে নির্দিষ্ট পেজ নম্বর থাকে (যেমন: ই-কমার্স সাইট Amazon বা দারাজ)। ইউজার জানে সে কত নাম্বার পাতায় আছে।
- **Infinite Scrolling:** নিচে নামতে থাকলে অটোমেটিক নতুন ডাটা আসতেই থাকে (যেমন: Facebook, TikTok)। এটি মূলত ইউজারকে বেশিক্ষণ অ্যাপে আটকে রাখার জন্য ব্যবহৃত হয়।

৫. ডাটাবেস কুয়েরি উদাহরণ (SQL)

আপনি যদি ডাটাবেস থেকে ২য় পাতার ১০টি ডাটা আনতে চান, তবে কুয়েরি হবে অনেকটা এরকম:

SQL

```
SELECT * FROM products

ORDER BY id

LIMIT 10 OFFSET 10;
```

(এখানে LIMIT 10 মানে ১০টি ডাটা নেবে, আর OFFSET 10 মানে প্রথম ১০টি বাদ দিয়ে পরের ১০টি থেকে শুরু করবে।)

আপনি যদি একজন ডেভেলপার হন, তবে আপনার জানা উচিত যে কোনো প্রজেক্টের ব্যাকএন্ড এপিআই (API) বানানোর সময় প্যাগিনেশন যোগ করা বাধ্যতামূলক।

আপনি কি **Node.js, Python বা PHP** দিয়ে কীভাবে প্যাগিনেশন কোড লিখতে হয় তা দেখতে চান? আমি একটি প্র্যাকটিক্যাল উদাহরণ দিতে পারি।