

Diferença entre ArrayList & LinkedList

O LinkedList e o ArrayList são duas implementações diferentes da interface List em Java, cada uma com suas próprias características e usos adequados.

1. ArrayList:

- Implementa a interface List usando uma matriz dinâmica para armazenar os elementos.
- Ele permite acesso rápido e aleatório aos elementos com base no índice.
- É mais eficiente em termos de acesso (leitura e gravação) do que o LinkedList.
- No entanto, adicionar ou remover elementos no meio da lista pode ser menos eficiente, especialmente para grandes conjuntos de dados, pois pode exigir a realocação de elementos na matriz.

2. LinkedList:

- Implementa a interface List usando uma lista duplamente encadeada.
- Fornece um bom desempenho para adição ou remoção de elementos no início ou no meio da lista. Isso ocorre porque, ao contrário do ArrayList, não há necessidade de realocar elementos.
- O acesso aleatório aos elementos é menos eficiente do que o ArrayList, já que é necessário percorrer os nós da lista a partir do início ou do final.
- Consome mais memória do que o ArrayList, pois cada elemento em uma LinkedList precisa armazenar referências para o próximo e o anterior.

Em resumo, se você precisa de acesso rápido aos elementos por índice e não planeja adicionar ou remover elementos com frequência, o ArrayList é geralmente a melhor escolha. Por outro lado, se você precisa frequentemente adicionar ou remover elementos no início ou no meio da lista e não se importa com o acesso mais lento aos elementos por índice, então o LinkedList pode ser mais adequado.