* Paging | Non-contiguous Memory Allocation

- To overcome the issue of External fragmen We use Non-contiguous memory allocation; two Ways of doing this is "Paging" and "segmentation".

- **Paging** :- Break the memory and problem into fixed-sized blocks.

Physcial Memory -> (RAM) -> divide into fram
Logical Memory (Process) -> divide into Pages.

"Page size = frame size".

Now pages can stored anywhere in the frames of RAM.

- Example :- Page/Frame size = 1KB.
A process of 4 KB will be divided into 4 pages :- P0, P1, P2, P3.

- Free frames : F3, F4, F10, F12.

- Process can be allocated as

P0 -> F3
P1 -> F4
P2 -> F10
P3 -> F12

Page No.:
Date:

Even though RAM spaces are scattered, the process run fines because OS knows where each page is store via "Page table".

- **Page Table** :-

① It is a type of Data structure maintained by OS.

② It maps Page Number → Frame Number.

- **Address Translation** :-

Each logical address generated by CPU is divided into

① Page number (p) → Index into page table.

② Page offset (d) → Exact location inside the page.

Physical address ⇒ Frame Address + Offset.

- **Issue in Paging**

Each memory access needs to:-

① Read the page table

② Then access the memory.

- **Solution** :-

TLB (Translation Lookaside Buffer), a small fast cache hardware, stores the recent page → frame mappings.

① If CPU finds mapping in TLB (TLB hit), it directly gets the frame number.

② If not (TLB miss), it checks Page Table and updates TLB for next time.