Disadvantages of File-Based System

We will highlight the issues that arise when we use a file-based system and how databases are the solution.

Keeping in mind the university management system discussed in the previous lesson, we will shine a light on the problems that occur when we use a file-based system, which in turn are the reasons we shifted to database systems.

1. Data redundancy

Often, within an organization, files and applications are created by different programmers from various departments over long periods. This can lead to data redundancy, a situation that occurs when the same data is present in many places (files).

For example, if a student wants to change his/her phone number, he/she has to get it updated in various places (files). Similarly, old records must be deleted from all sections representing that student.

2. Data inconsistency

Data is said to be inconsistent if **multiple copies of the same data do not match with each other**, which wastes storage space and duplicates effort. Consider the case where a student's phone number is different in the accounts department and academics department, it will be inconsistent. Inconsistency may be because of typing errors or not updating all copies of the same data.

may be because of typing errors or not updating all copies of the same data.

3. Difficult data access

Another issue is that the user needs to know the exact location of the file in order to access data, which can be a very cumbersome and tedious process. Let's say the user wants to know the hostel allotment number of a specific student from 10,000 unsorted students' records, it will prove to be quite monotonous.

4. Security problems

Using a file-based system may lead to **unauthorized access** of data. If a student gains access to the file that contains his marks, he can change it without authorization.

5. Difficult concurrent access

Concurrency is the ability of the database to allow multiple users access to the same information at the same time. Typically, in a file-based system, when an application opens a file, that file is locked. This means that **no one else can access the file at that moment in time**.

For example, if one of the departments in the university accesses the data on a specific student, the other departments will have to wait until the first department is done before they can access it. Thus, concurrency is not maintained in a file-based system.

The difficulties that arise from using the file-based system have prompted the development of a new approach to managing large amounts of organizational information called the *database approach*.

In the next chapter, we will dive deep into the fundamentals of database systems.