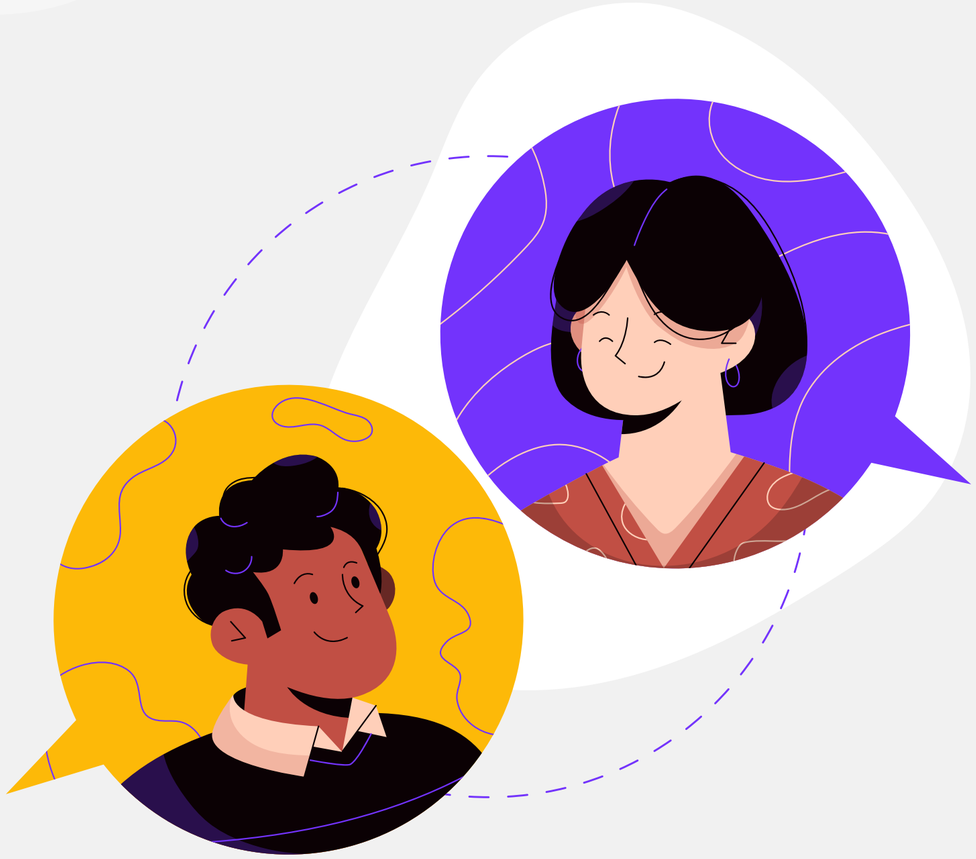
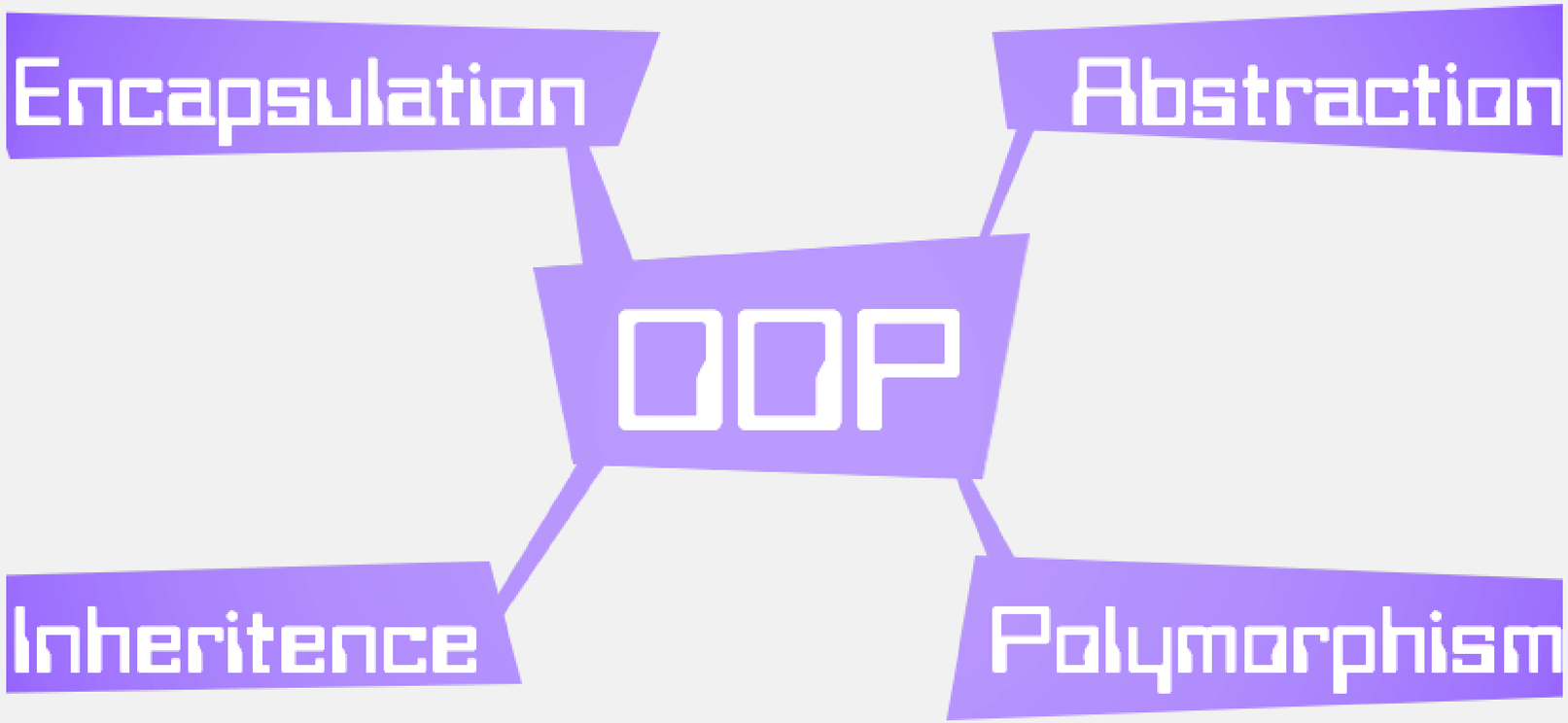
Important OOPS concepts - Abstraction

Let's Learn →



→

Strategy is to thoroughly learn 4 pillars of OOPS



Data Abstraction

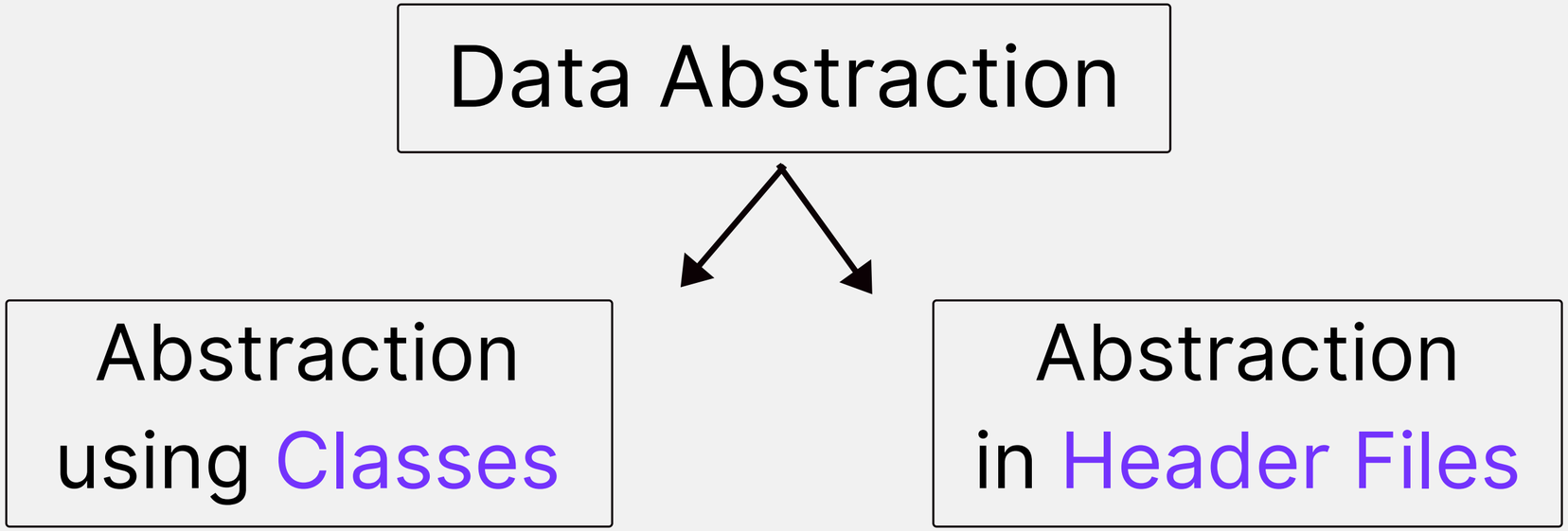
- **Data Abstraction is a process of providing only the essential details to the outside world and hiding the internal details, i.e., representing only the essential details in the program.**

- In technical terms, Data Abstraction is a programming technique that depends on the separation of the interface and implementation details of the program.

- C++ provides a great level of abstraction. For example, pow() function is used to calculate the power of a number without knowing the algorithm the function follows.

Data Abstraction can be done in TWO Ways

1. **Abstraction using classes**
2. **Abstraction in header files.**



Abstraction
using Classes

An abstraction can be achieved using classes. A class is used to group all the data members and member functions into a single unit by using the access specifiers. A class has the responsibility to determine which data member is to be visible outside and which is not.

Public specifier: When the members are declared as public, members can be accessed anywhere from the program.

Private specifier: When the members are declared as private, members can only be accessed only by the member functions of the class.

Abstraction
in Header

An another type of abstraction is header file. For example, pow() function available is used to calculate the power of a number without actually knowing which algorithm function uses to calculate the power.

Thus, we can say that header files hides all the implementation details from the user.

Advantages of Abstraction

- Implementation details of the class are protected from inadvertent user-level errors.
- A programmer does not need to write low-level code.
- Data Abstraction avoids code duplication, i.e., the programmer does not have to undergo the same tasks every time to perform a similar operation.
- The main aim of the data abstraction is to reuse the code and the proper partitioning of the code across the classes.
- Internal implementation can be changed without affecting the user-level code.

What is Data Abstraction? Explain with real world example.

Data abstraction is a very important feature of OOPs that allows displaying only the important information and hiding the implementation details.
For example, All are performing operations on the ATM machine like cash withdrawal, money transfer, retrieve mini-statement, etc. but we can't know internal details about ATM.

How to achieve data abstraction?

Data abstraction can be achieved through:

1) Abstraction using Classes
2) Abstraction in Header

Thank You

Let's learn the concept of Encapsulation along with the most probable interview questions on encapsulation in the next Lecture