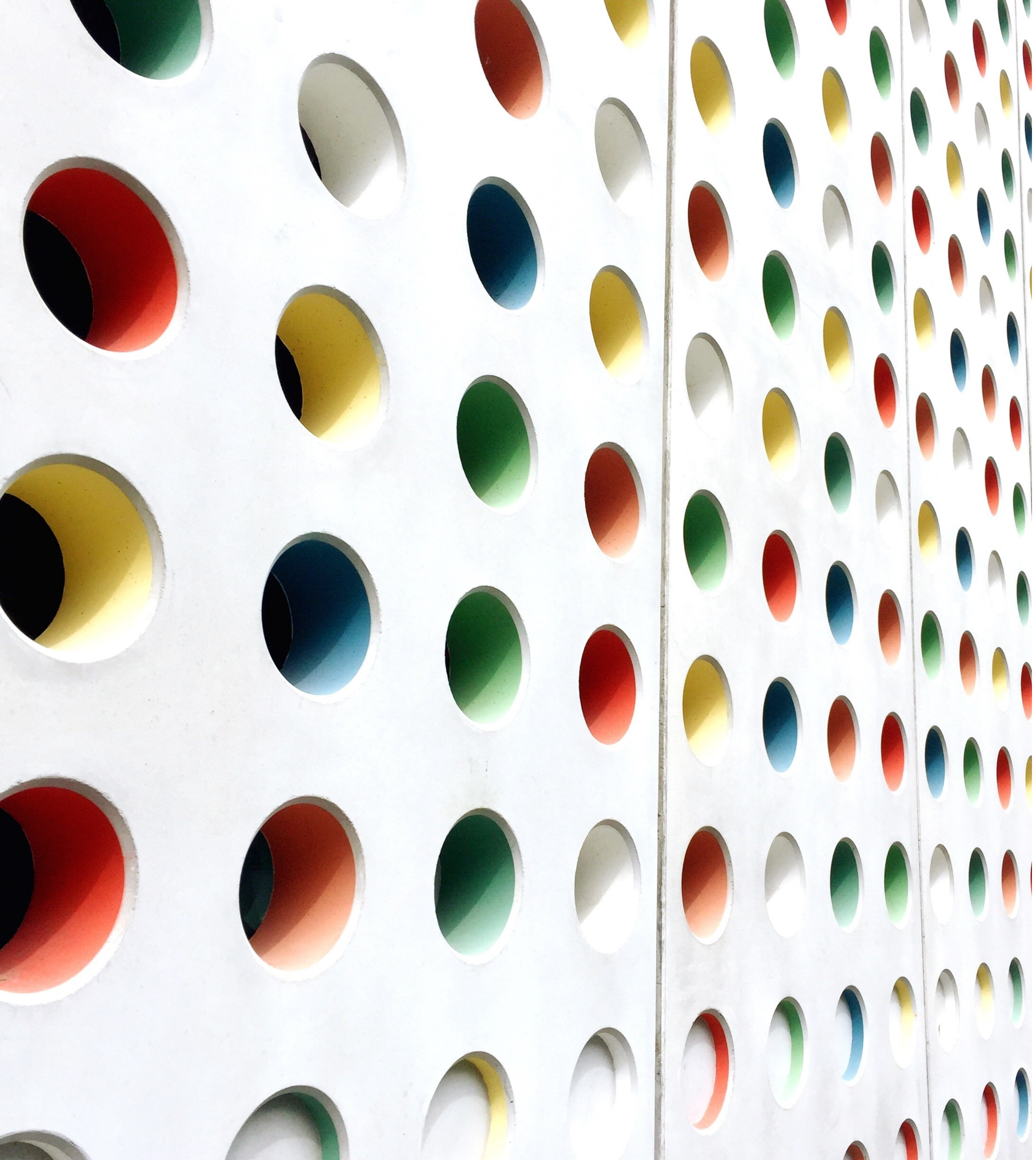


JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY

jiit

विद्या तत्व ज्योतिसम:

Topic: Introduction to object oriented programming

COURSE TITLE:

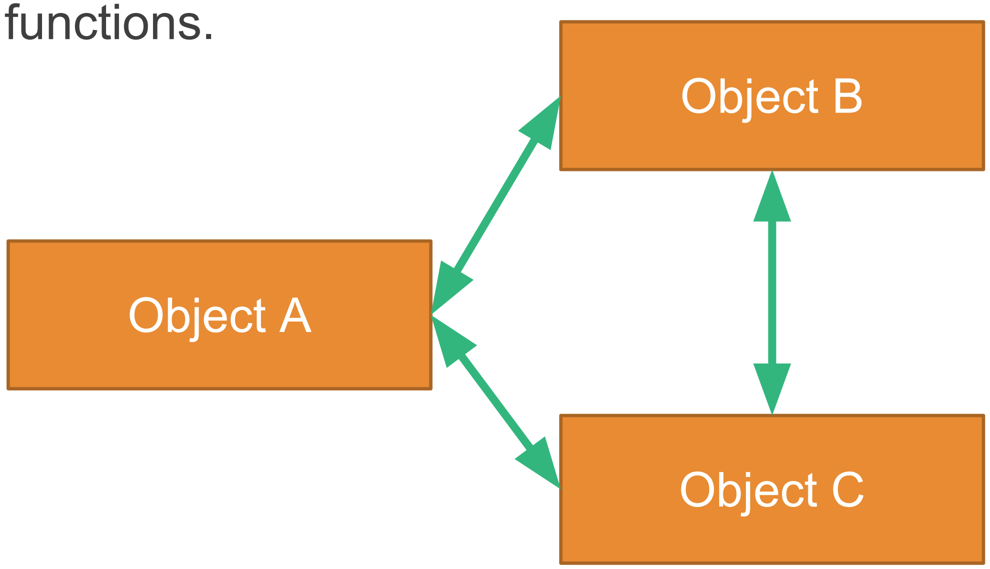
SOFTWARE DEVELOPMENT
FUNDAMENTALS-2

Procedural vs Object oriented

Procedural programming	Object oriented programming
Divided into primitive entities called as functions.	Divided into primitive entities called as objects.
It gives preference to functions over data.	It gives preference to data over functions.
It doesn’t support data hiding, polymorphism and exception handling.	It supports data hiding, polymorphism and exception handling.
It doesn’t have access specifiers.	It has access specifiers such as private, public and protected.
Inclusion of new function and data is cumbersome.	Inclusion of new function and data is easier.
Example: C	Example: C++, JAVA, Python

Introduction to object oriented programming

- Object oriented programming emphasizes more on data than procedures.
- Each program is represented using objects.
- An object communicates with another object through functions.
- Data is hidden and protected from external functions.
- It has following characteristics,
  - Data Encapsulation
  - Data hiding
  - Data abstraction
  - Inheritance
  - Polymorphism



Introduction to object oriented programming

- **Data Encapsulation** binds data with its corresponding functions.
- Only those functions which are encapsulated together can access the data.
- These bounded functions acts like an interface that allows access to the data for any program.
- By restricting the access to data, it provides **data hiding** from external program.

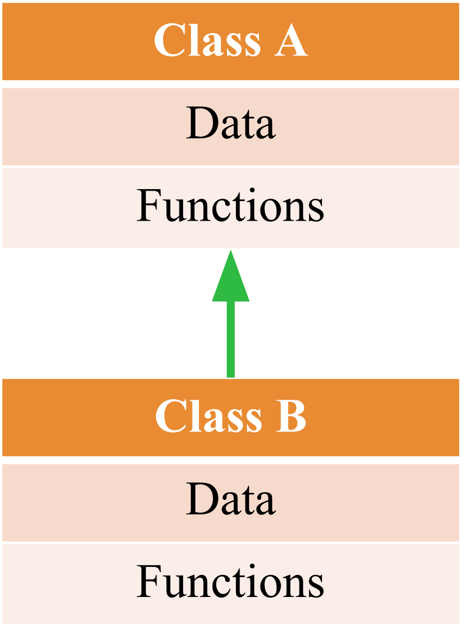
Class A
Data
Functions

Introduction to object oriented programming

- **Data Abstraction** shows only relevant information about the objects instead of the entire background details.
- It separates the interface from its implementation.
- Classes are used to implement abstraction and they are called as abstract data types (ADT).
- For example, the function **cout** is used as a interface to display on the console and the internal implementation is not revealed to the outside world.

Introduction to object oriented programming

- **Inheritance** is a way to include functionalities of a parent class to a child class.
- It doesn't allow inheritance of private members from parent class.
- It enhances code reusability



Class A

Data

Functions

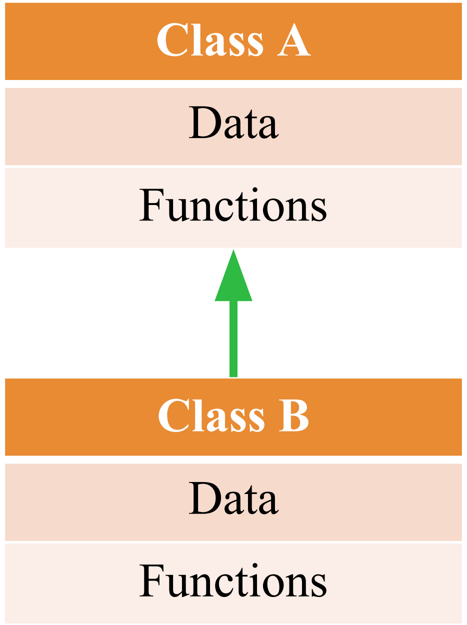
Class B

Data

Functions

Introduction to object oriented programming

- **Polymorphism** enables to define many forms of a function and it is invoked based on the object calling it.
- The functionality of a function differs according to the object that calls it.
- Inheritance helps to implement polymorphism.



Benefits of object oriented programming

- Data hiding helps to protect the data and it secures the programs.
- Exception handling can be performed using OOP.
- Program templates can be developed and used in future applications.
- Programs can be easily upgraded and portable.
- Reduces redundant codes and improves code reusability.
- Improves productivity by saving development time.

References

- Herbert Schildt, “C++: The complete reference”, Mc Graw Hill Osborne media, 4th edition, 2017
- Robert Lafore, “Object oriented programming in C++”, SAMS, 4th edition, 2002