# CSC 127 Elements of OOP

Elements of OOP

- Objects
- Classes
- Encapsulation
- Data Abstraction
- Inheritance
- Polymorphism
- Dynamic Binding
- Message Passing

Objects

- OOP uses objects as its **fundamental building blocks.**

- Objects are the **basic run-time entities** in an object-oriented system.

- Every object is associated with **data** and **functions** which define meaningful operations on that object.

- Object is a real world **existing entity**.

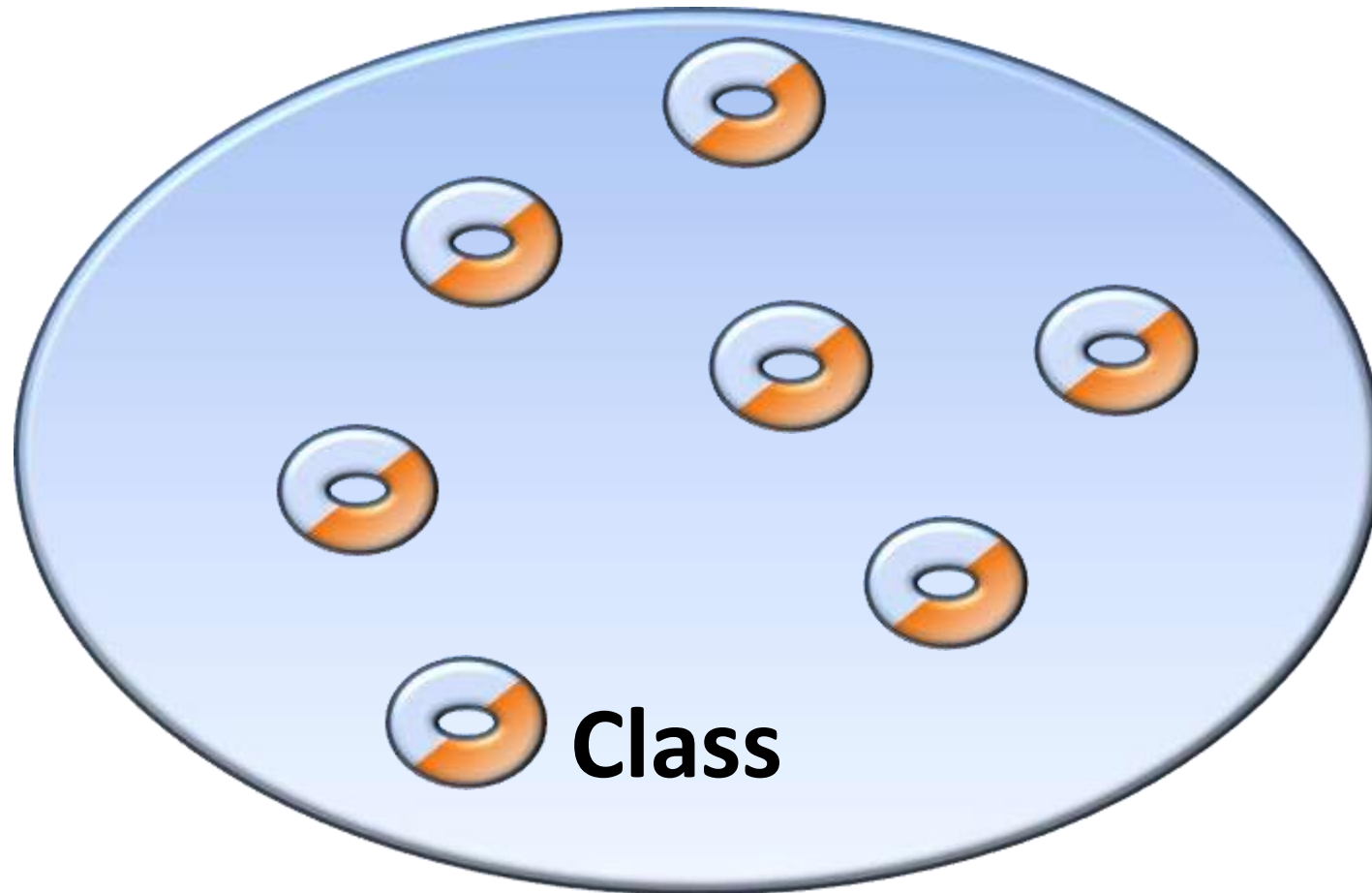- Object is an**Instance** of a particular class.

Object

Operation

Operation

Attributes

Operation

Operation
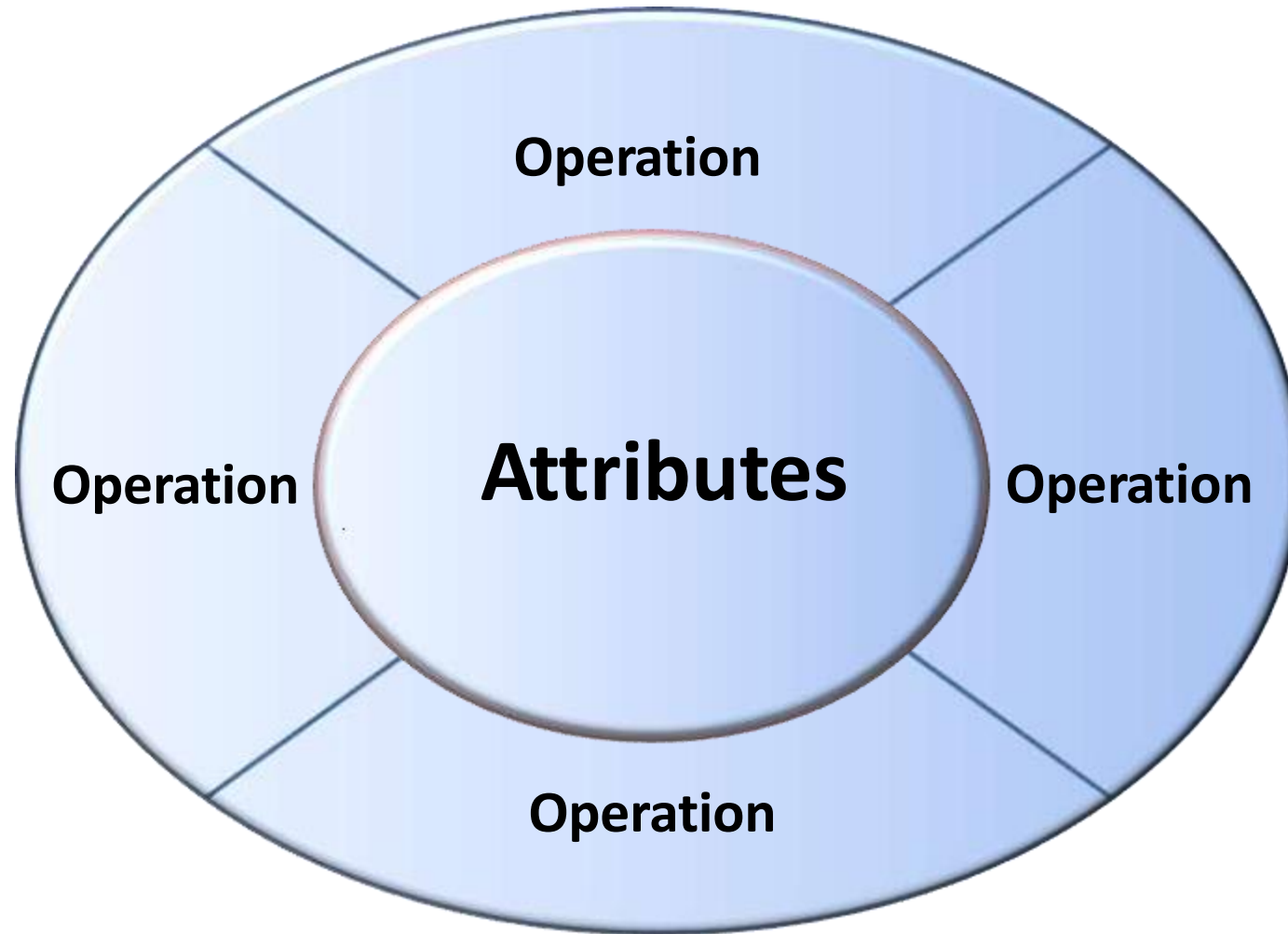
Example: StudentObject
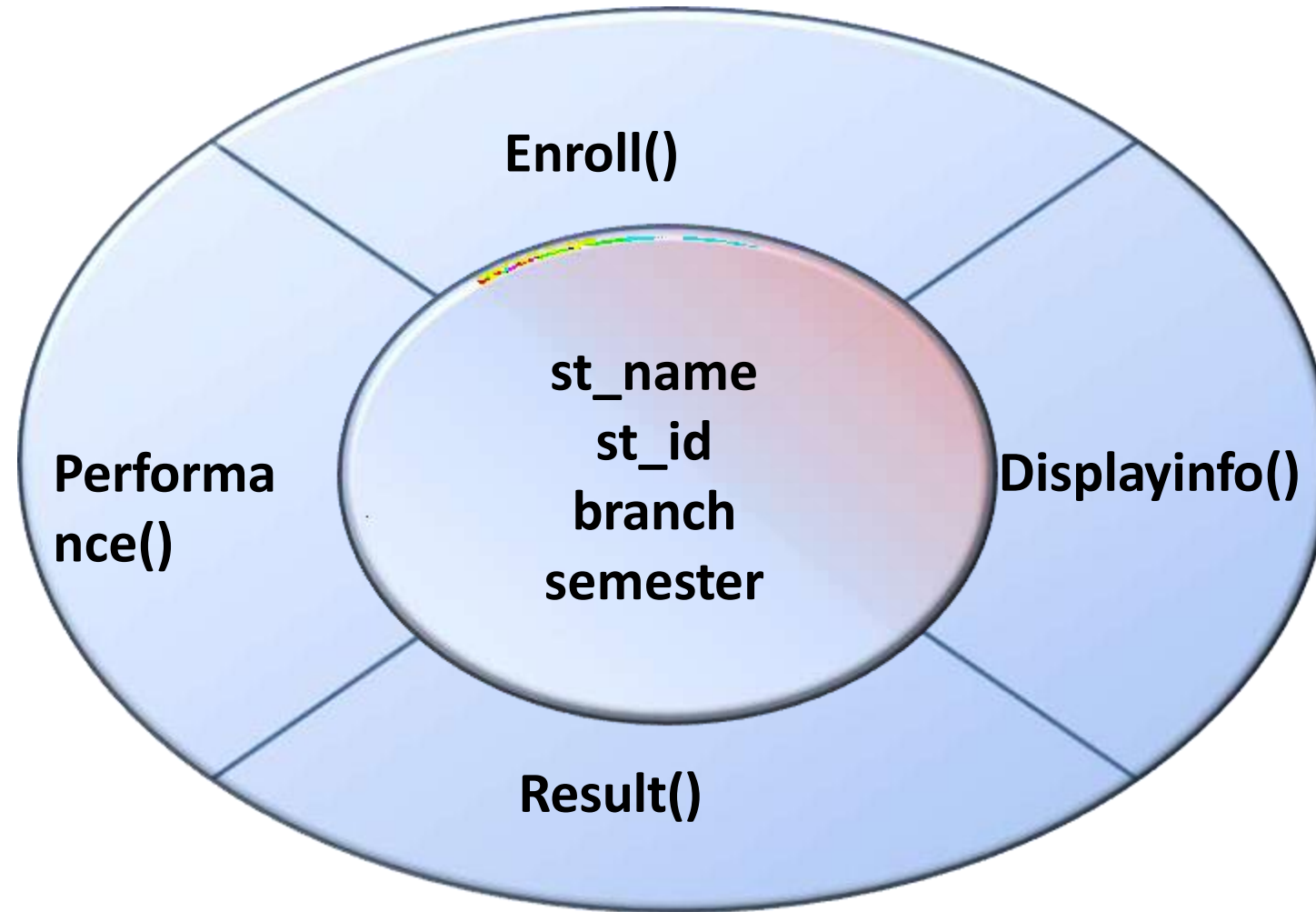
Class

- Class is a collection of **similar objects**.

Class

Encapsulation

"Mechanism that associates the **code** and the **data** it manipulates into a single unit and keeps them safe from external interference and misuse."

Encapsulation

Class: student

Attributes: st_name, st_id, branch, semester

Functions: Enroll()
Displayinfo()
Result()
Performance()

Data Abstraction

"A data abstraction is a **simplified view** of an object that includes only features one is **interested** in while **hides** away the **unnecessary** details."

"Data abstraction becomes an **abstract data type** (ADT)or a user-defined type."

C++ Implementation

```cpp
class class_name
 {
Attributes;//Properties
Operations;//Behaviours
};
```

C++ Implementation

```cpp
class student
{
char st_name[30];
char st_id[10];
char branch[10];
char semester[10];
Void Enroll( );

Void Displayinfo( );

Voide Result( );

Void Performance( );
};
```

```cpp
class stack
 {
int stck[SIZE];
int tos;
void init();
void push(int i);
int pop();
};
```

Inheritance

- "Inheritance is the mechanism to provides the power of **reusability** and **extendibility**."
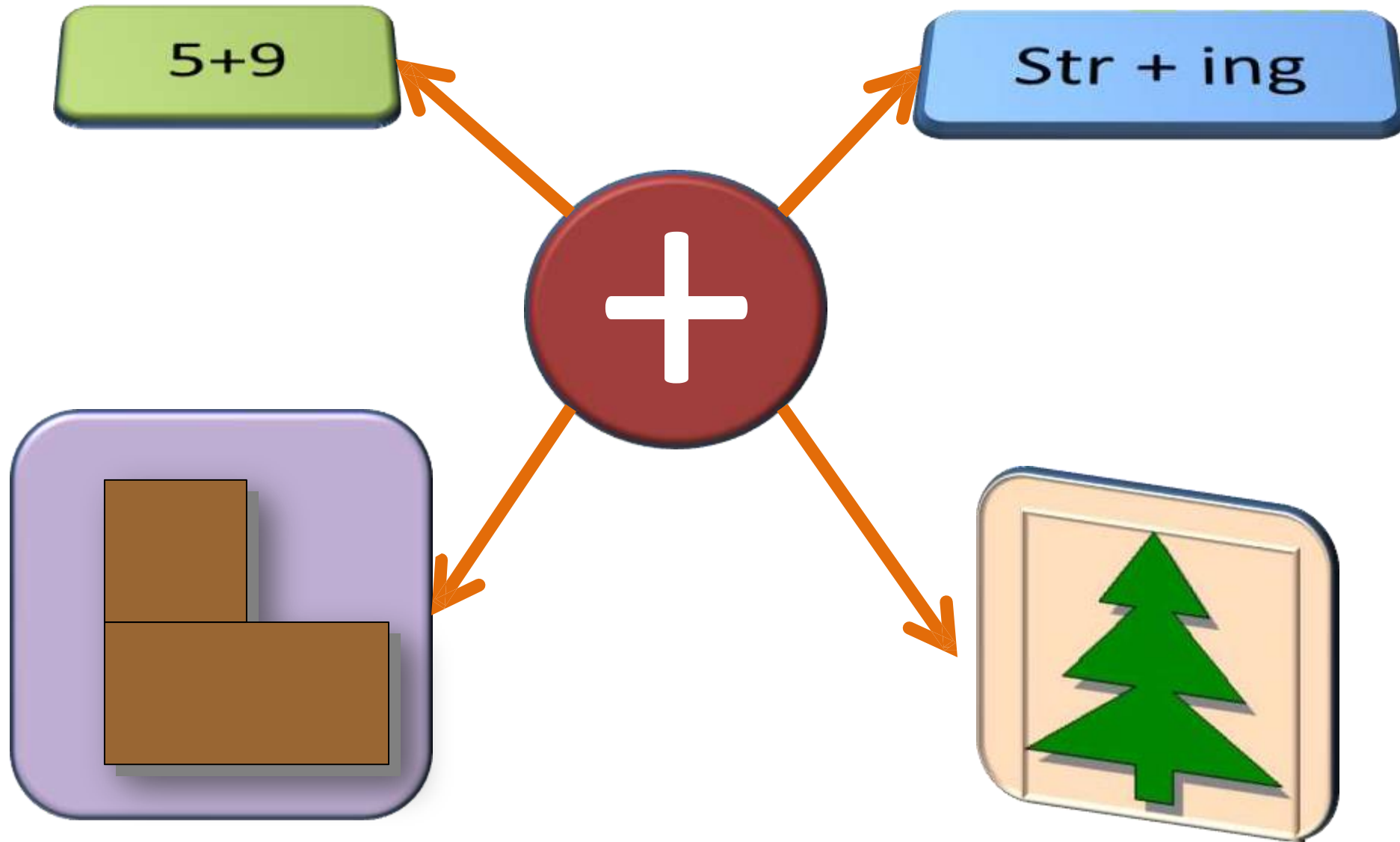- "Inheritance is the process by which one **object can acquire the properties of another object**."

Inheritance
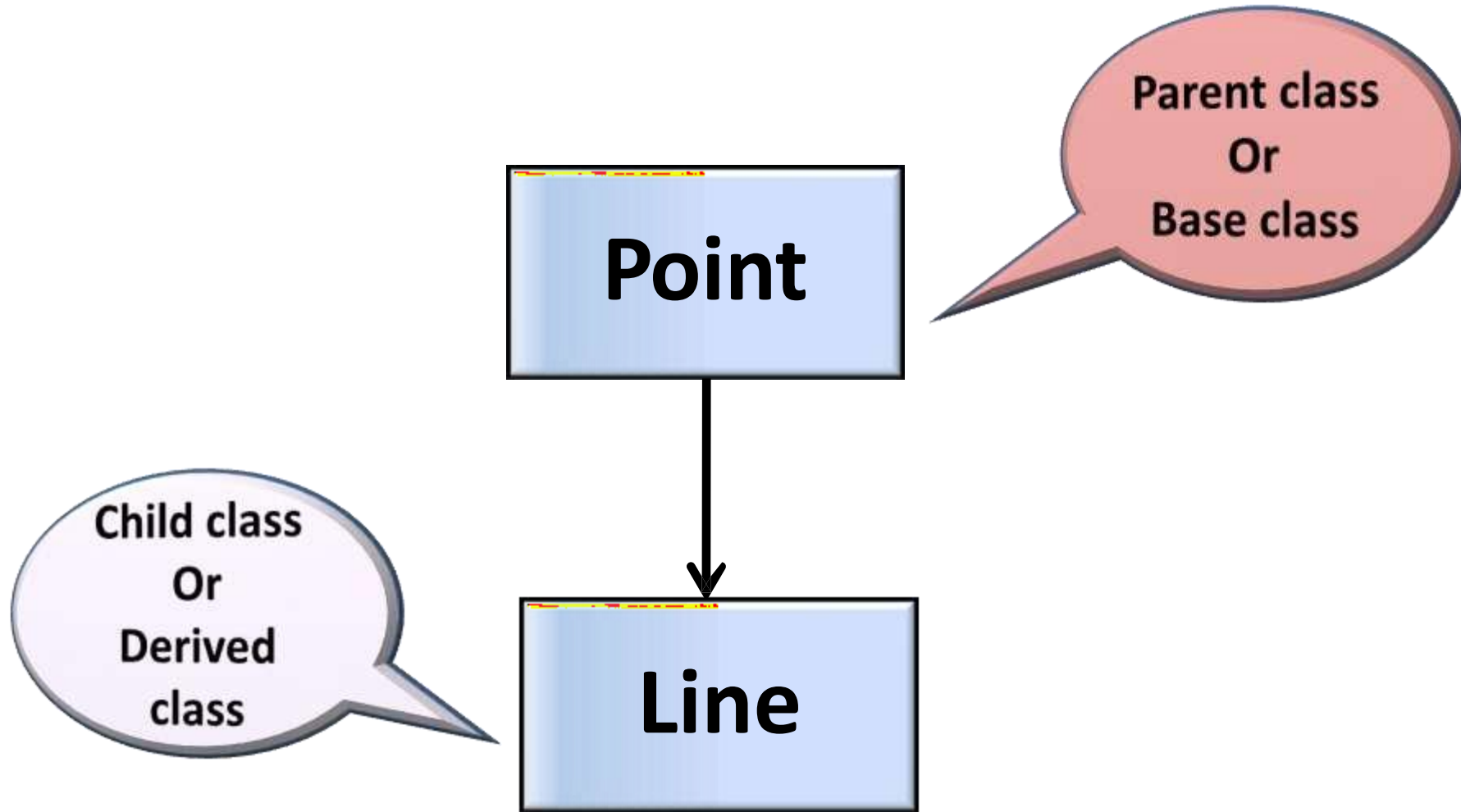
Polymorphism

- Polymorphism means that the **same thing** can exist in **two forms**.

- "Polymorphism is in short the ability to call **different functions** by just using **one** type of **function call**."

Polymorphism

5+9

Str + ing

Dynamic Binding

" Dynamic Binding is the process of **linking** of the **code** associated with a **procedure call** at the **run-time**".

Message Passing

- "The process of invoking an operation on an object. In response to a message the corresponding method is executed in the object".

Message Passing

StudentObject

FacultyObject

Performance

Result

MgmtObject

Performance