CT60A2411
Object oriented Programming: Week 2

Ashok Kumar Veerasamy, PhD

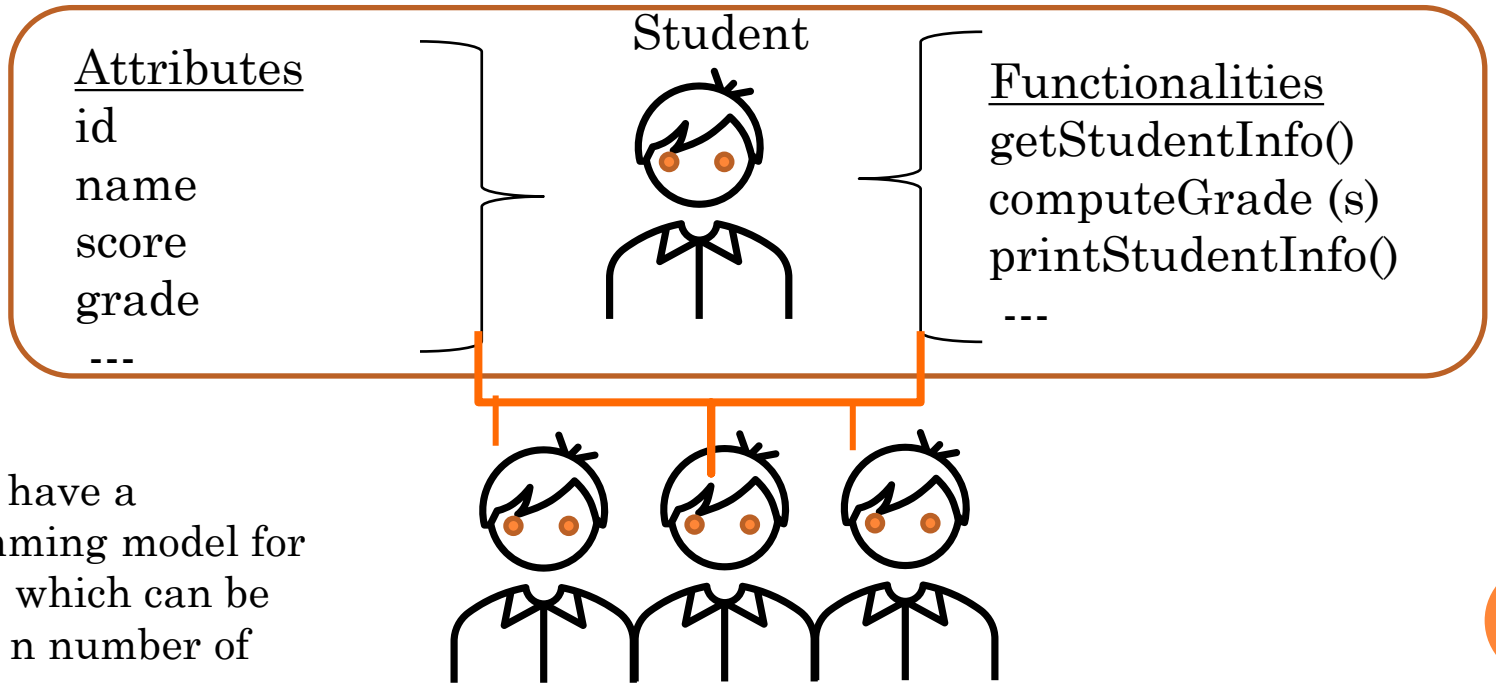
Learning objectives: This week

- Object oriented programming
- Objects and classes

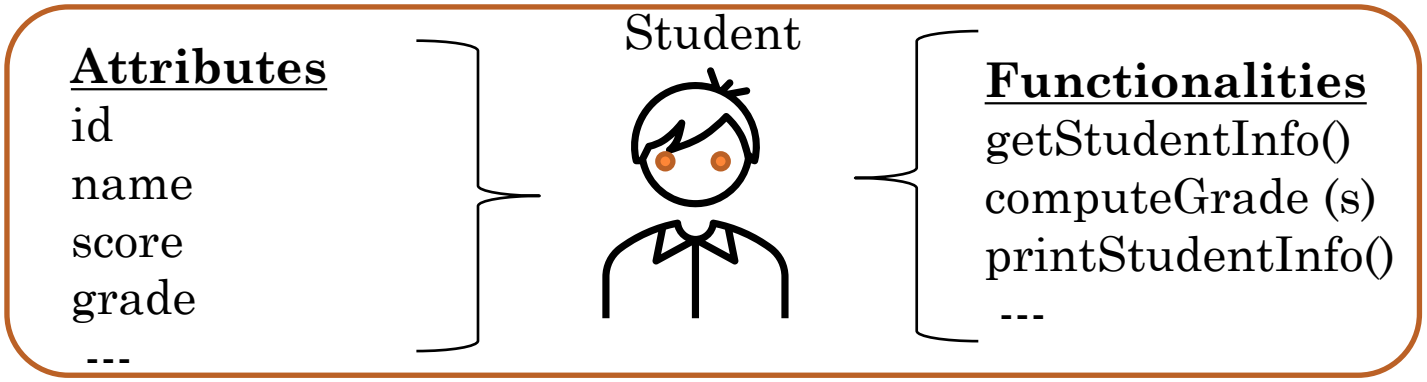
At the conclusion of this lecutre, students will be able to know the basics of object-oriented programming (OOP) concepts and how OOP organizes software design around data instead of functions and logic.

- **What is Object oriented programming (OOP)?**
  - Developing a programming model that can be reused / developing a reusable software
- **Example:**
- Let us create a programming model for student
- student is an object/entity or a real-world object.

- What **attributes** are associated or connected with student?
- What **functionalities** are required to compute grade and more?



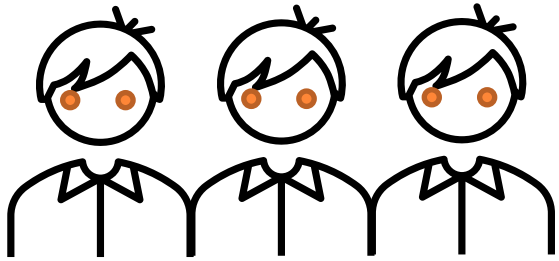
Now we have a programming model for student, which can be used for n number of students.



In OOP, these real-world objects are represented as **Class** .
The instance of those classes are called **object**.

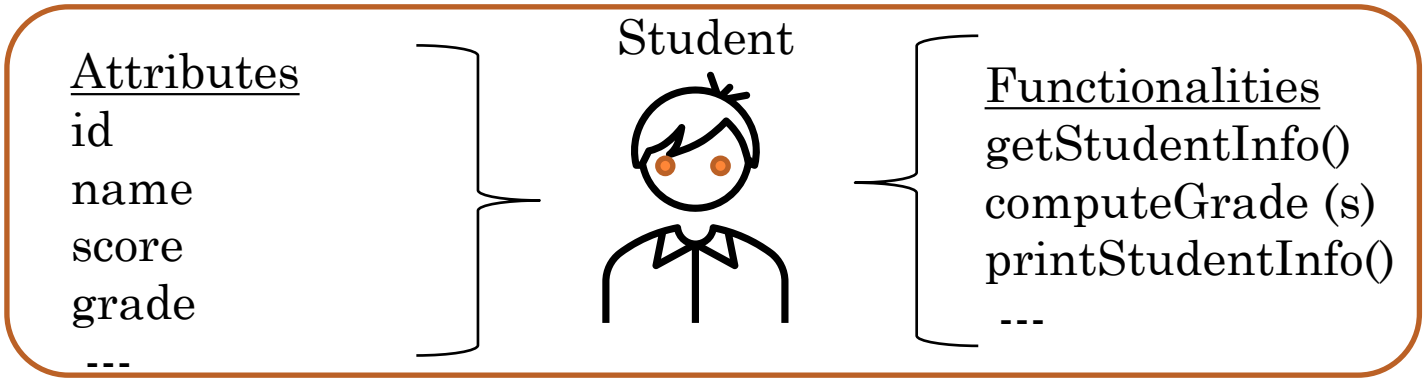
The attributes of the class called **state/data** **that** defined as ***variables*****.**
The functionalities of the class called **behavior/action** that defined as ***methods***.

Here student Id, name… are state of the class
getstudentInfo(), computeGrade()…. are behaviour of the class.



student 1 student 2 student3 ….

Student1, student2,….. are instances of class or object. Each object(student entity) has its own instance of data (reusability).



A unified modelling language(UML) which is a graphical notation that used to depict how a real-world object is represented as class in OOP paradigm. The UML diagram for student class is: