Object Oriented Programing 1

Lecture 1: Course Overview and Introduction to Object Oriented programing

By

Elubu Joseph

MSci.IS

Email: josebulinda@gmail.com

or

jose@kumiuniversity.ac.ug

Agenda

- 1. Course Overview
- 2. Introduction to Object Oriented programing

Course Overview

- 1. Course Description
- 2. Course objectives
- 3. Learning outcomes
- 4. Course Calendar/Content
- 5. Mode of Delivery
- 6. Course Requirements
- 7. Assessments Criteria and
- 8. Text Books and References

Course Description

Objected Oriented programming is a term loosely applied to mean any kind of programming that uses a programming language with some object oriented constructs or programming in an environment where some object oriented principles are followed.

Compared to procedural programming, a superficial examination of codes written in both styles would reveal that object oriented codes tends to be broken down into vast numbers of small pieces, with the hope that each piece will be trivially verifiable.

Course Description +

Objected Oriented programming 1 course is therefore, designed to build up a learner with a capability of developing systems based on object oriented concepts/constructs.

Course Objectives

To develop an understanding of the principles of the object-oriented paradigm.

To discuss approaches to object-oriented analysis and design

To introduce a learner to syntax, class hierarchy

Learning Outcomes

On successful completion of this unit, learners should be able to:

• Understand the principles of the object-oriented paradigm and the advantages of this approach.

- Explain and apply an object-oriented analysis and design process to create industry standard object modeling notation.
- Use an object-oriented programming language to build simple applications.

Course Calendar/Content

The main content of this course is organized in weeks as seen in the <u>course syllabus</u>.

Mode of delivery

Teaching will be by lectures through Lecture textmaterial and video tutorials

Requirements

This course is designed to build up from Programing Methodology using C programing language. However, its not a must for one to have C programing language to be able to carry on with this course, although having that knowledge would be an added advantage for quicker absorption of the OOP concepts.

Mode of Assessment

- 1. Tests and / Assignments 30%
- 2. Final examination 70%
- 3. Final Total 100%

Text Books

- Beginning Programming with Java for Dummies (3th Edition), Burd, B. (2012), John Wiley & Sons.
- Beginning Java Programming: The Object-Oriented Approach, Deepak, V. Et al (2015), John Wiley & Sons.
- Object Oriented Computer Systems Engineering, Derrick, M. et al(2012), Springer Science & Business Media.
- 4. Java Programming (8th Edition), Joyce ,F.1 (2015), Cengage Learning, ISBN:

Introduction to Object Oriented programing

Introduction to Object Oriented programing-OOP

OOP is a programming approach which uses real world approach to solve real life problems based on the help of real world algorithms. Object oriented technique offers better and easy way to write program than procedural programming techniques offered by languages such as ALGOL, PASCAL and C etc.

OOP is a programing style which is associated with the concept of Class, Objects and various other concepts like Inheritance, polymorphism, Abstraction and Encapsulation.

Major Features/Concepts of OOP

- 1. Object
- 2. Class
- 3. Inheritance
- 4. Polymorphism
- 5. Encapsulation
- 6. Abstraction

Understanding OOP

To understand this concepts lets take analogy of a human being, when God created human beings He created them Male and female



with common and unique features in both. For Example, All humans have Hands, Legs, eyes, Head etc.



But there are unique features/functions for male or female only, name them?

Body functions

There are some functions shared between male and female e.g. Eating, Drinking, Breathing etc.

But there are some functions that females perform that males don't. **E.g.** pregnancy, giving birth, Ministration Periods etc. This analogy should make us realize that a class can be a Human being, Male, or Female

Note.

Human class should have features that both Male and Female classes share in common, otherwise Male and Female classes will take care of their unique features/functions but borrow common features form Human class.

How to access Anything from God

To access anything from God, One has to ask in the name of Jesus, John 16:23

"On that day, you won't ask me for anything. Truly, I tell all of you emphatically, whatever you ask the Father for in my name, he will give it to you."

Or

To go to God, one has to believe in Jesus Christ first, John 14:5-6

5 Thomas asked him, "Lord, we don't know where you are going, so how can we know the way?" 6 Jesus told him, "I am the way, the truth, and the life. No one comes to the Father except through me.

We see that Jesus is object of God Class(es).

Object

An object is a real world entity. An entity has properties and tasks.

For Example a Human being is an object with properties such as Name, Colour, height, weight etc. And tasks such as Read, write, dance etc.

or

Object is an instance of a class which has physical existence. **E.g.** A man called Joseph exists physically and is an instance of Male class.

An object represents the class and consists of **properties** and **behavior** While a class is a logical definition.

So what is a class

A class is a blueprint of an entity, which defines the properties/features and functions of that given entity.

Like we said above male class is a blueprint of male being which defines the body parts and functions of the male.

Its important to note that a class is a design where everything is defined. Or we can also say it is a type or classification of real world entities. Human beings could be a type of Gods creation, Male Humans could be a subtype of Human beings.

Inheritance

Considering a Human class that holds all body parts that is shared between Males and Females,

Human class

Body parts such as: -

Hands, legs, Eyes, Hears etc.

Common body functions such as:

Walk, talk, see, sleep, etc

Female Class

Will take care of only unique body parts and functions that dose not belong to Male class but inherit every other part and function from Human Class

Male Class

Will take care of only unique body parts and functions that dose not belong to Female class but inherit every other part and function from Human Class

Note.

Human class should have features that both Male and Female classes share in common otherwise Male and Female classes will take care of their unique features/functions but borrow common features form Human class.

Abstraction

Abstraction means hiding the details from the outside world, while showing only necessary information.

Abstraction is a technique used in OOPs to hide the details of a given class from the outside world.

E.g. the mechanism used by the mouth to be able to eat, Nose to be able to breath and smell is hidden away from us. All we know is that the mouth is used for eating but the code behind it is not fully clear to Human beings.

Polymorphism

Is a concept used to define various ways though which tasks can be performed.

This is the concept that allow us to redefine how something is done either by using a different process for it or by using a different part to do it.

E.g. Human class has a body part called the nose which is specifically used to smell and breath but sometimes it can be used for feeding a sick person using a tube most especially when the mouth can no longer process food.

Polymorphism +

To accomplish its capabilities, polymorphism uses key concepts that is: -

- 1. Overriding
- 2. Overloading

Overriding

Considering Human Class which has Legs, used for working forward, when we force the legs to work backward we would have overridden the functions of the legs



Overloading



Still considering Human class, which has a body part called the nose which is specifically used to smell and breath, but sometimes it can be used for feeding a sick person using a tube most especially when the mouth can no longer process food. This is overloading.

Or consider using the hands to walk instead of the legs, in this case the Hand would have been overloaded.



Encapsulation

Is the concept used to bind Class properties to functions. E.g in the Human Class we have body parts that helps us to do different functions as seen below.



We Shall see this features of OOPs in details with clear examples when we start coding.

Advantages of OOPs

These days, most major software development is performed using OOP, due to the benefits that comes with this software development approach.

The benefits that this approach houses are : -

- 1. Modularity for easier troubleshooting
- 2. Reuse of code through inheritance
- 3. Flexibility through polymorphism
- 4. Effective problem solving

Modularity for easier troubleshooting

1. Modularity makes one know exactly where to look when something goes wrong. "Oh, the car object broke down?

The problem must be in the Car class!" You don't have to go line-by-line through all your code. That's the beauty of encapsulation.

2. Also, this modularity allows an IT team to work on multiple objects simultaneously while minimizing the chance that one person might duplicate someone else's functionality.

Reuse of code through inheritance

OOP allows reuse of code through the use of inheritance.

E.g. Suppose that in addition to your Car object, one colleague needs a **RaceCar object**, and another needs a **Limousine object**. Everyone builds their objects separately but discover commonalities between them.

Note, each object above is a *different kind of Cars*. This is where the inheritance technique saves time: Create one generic class (Car), and then define the subclasses (RaceCar and Limousine) that are to inherit the generic class's traits.

Of course, Limousine and RaceCar still have their unique attributes and functions.

Flexibility through polymorphism

Basing on the fact that polymorphism allow us overload and override functions/methods in different classes, you now need just a few functions, like "driveCar," driveRaceCar" and "DriveLimousine." RaceCar functions share some traits with Limousine functions.

This is where polymorphism comes into play. Because a single function can shape-shift to adapt to whichever class it's in, you could create one function in the parent **Car class** called **"drive"** — not "driveCar" or "driveRaceCar," but just "drive." This one function would work with the RaceCar function, Limousine function and so on.

Effective problem solving

OOP is ultimately about taking a huge problem and breaking it down to solvable chunks. For each mini-problem, you write a class that does what you require. And then — best of all — you can reuse those classes, which makes it even quicker to solve the next problem.

This isn't to say that OOP is the only way to write software. But there's a reason that languages like C++, C#, python and Java are the go-to options for serious software development.

Top Six Object oriented Programing languages

We will go through the six object-oriented programming languages commonly used today, including:-





is one of the oldest, most popular, and well-known object-oriented languages. It was designed to be platform-independent and easy to learn, and it has become the primary language used for Android app development.

Java also enjoys a strong following in the business world, thanks to its robustness and security features. This is a great programming language for beginners because it is simple to learn, but it can also be used for large projects where reliability is key.



Since its release in 1995, Java has gained immense popularity.

- 1. It is now part of the CPU on Mac OS X and every Android device,
- It powers most websites that use content management systems like WordPress or Drupal, and
- 3. It's used to code everything from iPhone apps to video game consoles.



C# is a multi-paradigm language developed by Microsoft. It is an OOP language that also supports functional, imperative, and generic programming paradigms. C# was developed in the early 2000s. It's very similar to the first Java, but it has been designed with the .NET framework, making it a little easier to code for Microsoft platforms like ASP.

C# has been gaining importance in the OOP community due to its interoperability with .NET Framework. Furthermore, it can be used to develop Windows desktop applications and Android mobile apps.

This language is a good choice for beginners because it's easy to learn and has strong support from the industry. Due to its popularity, many online resources makes learning this language easy.



is a high-level object-oriented programming language that is easy to learn and enables developers to work faster and integrate systems more effectively. It's a dynamic, interpreted language that emphasizes code readability and supports multiple programming paradigms, including functional, object-oriented, procedural, and imperative.

Python is a multi-paradigm programming language **built by Guido van Rossum**. Python is a good choice for beginners because it is easy to learn and read. It also has a strong community that produces a lot of great resources. It's a versatile language that can be used for scripting, web development, scientific computing, and more.



is a dynamic, open-source programming language focusing on productivity and simplicity. Developed in the 1990s by Yukihiro Matsumoto, Ruby has an elegant syntax that is natural to read and easy to write. Ruby has an extensive and comprehensive standard library, and its simple design makes it very reliable and portable.

One of the best things about this OOP language is its community. The Ruby community is very active, and there are a lot of great resources available online. Safe, object-oriented, and expressive, Ruby is a powerful scripting language that feels natural to program in.



is one of the object-oriented programming languages most commonly used to create dynamic web pages and is also a popular programming language in general. Developed by Rasmus Lerdorf in 1995, PHP was designed for web development and is commonly used in conjunction with HTML and CSS.

PHP is a good choice for beginners because it's also easy to learn and has a strong community that produces great resources.

TypeScript



TypeScript is one of the OOP languages. It is a superset of JavaScript that provides optional static typing, interfaces, and classes. The syntactical additions to JavaScript can be beneficial when writing large-scale applications that are expected to function correctly. Moreover, TypeScript offers type inference, which helps one write code faster and with fewer errors.

It also provides an alternate syntax for constructors and destructors. This syntax makes it easier to identify objects by name since they are always written first in the literal object. In addition, its open-source nature allows it to be quickly adopted into existing JavaScript projects or developed on its own.

Language of Choice for this course



Is our Language of choice in this course

Object Oriented Programing 1

Summary

- Course Overview (Course Description, Course objectives, Learning outcomes, Course Calendar/Content, Mode of Delivery, Course Requirements, Assessments Criteria and Reference Books)
- Introduction to Object Oriented programing:- We looked at the Definition, Major Features/Concepts of OOP(Class, Object, Inheritance, Polymorphism, Encapsulation and Abstraction), Advantages of OOP and finally looked at the Top six OOP languages.





Reference

Half, R. (2021, December 4). *4 advantages of object-oriented programming*. 4 Advantages of OOP | Robert Half. Retrieved March 15, 2022, from https://www.roberthalf.com/blog/salaries-and-skills/4-advantages-of-object-oriented-programming

Tran, T. (2021, December 17). *The list of object-oriented programming languages*. The List of Objectoriented Programming Languages. Retrieved March 15, 2022, from https://www.orientsoftware.com/blog/listof-object-oriented-programming-languages/

What is oops. Studytonight.com. (n.d.). Retrieved March 7, 2022, from https://www.studytonight.com/java/object-and-classes.php