Not just **DRY**: Write Reusable Code

Samuelson Tijesunimi ATIBA

@**PyCon** 2018

Samuelson Tijesunimi **ATIBA**

(Intricate thinking – Solutions)

- Electrical and Electronics Engineering Student @ LAUTECH
- Consultant Developer @ YDSN
- ex-CTO @ Chopnow

- I am a problem analyzer and solutions “profferer”.
- Python Enthusiast

What is **DRY**?

- **DRY** is an acronym for **D**on't **R**epeat **Y**ourself
- It is a **Principle*** of Software Development

  (i.e. it is not a convention or a paradigm)

- It states that,

  "Every piece of knowledge must have a single, unambiguous, authoritative representation within a system"**

* Glossary Term ** See References

Reusable Code - OOP Concepts

- Object Oriented programming is an approach to programming focused on representations of real life objects and data with code.**
- The major aim is to focus on the data and objects the program is meant to represent and manipulate. Thus, a single representation of the object is codified into a class
- OOP introduces **abstraction*** – limiting the scope of the problem – and **encapsulation*** – hiding the granular implementation. These two form the building blocks of reusable code.

* Glossary Term ** See References

Abstraction

- Representation of the Object: define abilities (methods) of objects and the descriptors (properties) of the object
- Ignoring those properties/methods of an object that is outside the scope of the problem to be solved
- It is a representation of the limitedness of the scope of an object posed by a problem
- Answers "What do I need to represent my object"

* Glossary Term   ** See References

Encapsulation

- Hiding complexity, provide interface, bundle data with objects
- Answers how – hides how and present unified interface for interacting and manipulation of objects

* Glossary Term ** See References

Putting it all **together**

- Understand the problem to be solved
- Identify Objects to be created
- Create Object Abstractions
- Implement Object methods
- Selectively expose Object methods
- Write Code with Simplicity > Consistency > Completeness

  This will encourage simple extension of current codebase to reach completion.

* Glossary Term ** See References

Thanks for listening

Samuelson Tijesunimi **ATIBA**

samuelsontijesunimi@gmail.com

Linkedin.com/in/Samuelson_Atiba

Twitter/the_samuelson

* Glossary Term ** See References