



# Python OOP: Glossary



## Key Takeaways

- Important Terminology:

- **OOP:** Object-Oriented Programming. A programming paradigm or approach used to analyze and solve problems that is based on the representation of real-world objects in the system.
- **“Pythonic”:** adjective commonly used to describe an approach to programming that follows the standards, rules, and best practice of the Python programming language.
- **Class:** one of the building-blocks of Object-Oriented Programming that acts as a “blueprint” where the data and the actions of the objects are defined.
- **Instance:** a concrete object that is created from the class “blueprint”.
- **Method:** an “action” defined in the class that the instances of the class can perform. It is very similar to a function, but closely related to instances such that instances can call methods and methods can act on the individual data of the instances.



# Glossary



## Key Takeaways

- Important Terminology:

- **Object:** a concrete instance of a class stored in memory.
- **Docstrings:** (also Documentation Strings) they are strings located at the top of functions, classes, methods, or modules that describe their purpose, arguments, attributes, and relevant characteristics. They are “linked” to the element they describe by the `__doc__` attribute.
- **Mutation:** the result of changing the original object in memory instead of making a separate copy of the object.
- **Aliasing:** when two or more names (variables) point to the same memory address.
- **Cloning:** process of creating a separate copy of an object. This copy is initially equal to the original object, but it could change and the original object would not be affected.

