* **Fundamentals**

C# is an object-oriented programming language. The four basic principles of object-oriented programming are:

* Abstraction Modeling the relevant attributes and interactions of entities as classes to define an abstract representation of a system.
* Encapsulation Hiding the internal state and functionality of an object and only allowing access through a public set of functions.
* Inheritance Ability to create new abstractions based on existing abstractions.
* Polymorphism Ability to implement inherited properties or methods in different ways across multiple abstractions.

*Revature Training notes*

* OOP = Object Oriented Programming
* Everything is based on the concept/idea of “objects”
* Every object is based on a class
* Every class has a state/properties/fields
* Every class has an action/function/method
* Based on having objects interact with each other

**Abstraction**

* + It “Hints” at what it is or does
  + Only shows the essential or necessary features, and hides (abstracts) the irrelevant stuff.
  + Examples – making coffee, starting and driving a car

**Encapsulation**

* Wrapping/combining/mating data/state with functionality/action into a single object (class).
* Gives us the ability to protect the data/members/fields of an object so that they cannot be modified (either by accident or malicious intent).
* Public, Private, Internal (Access Modifiers)
* Access Modifiers – let us limit what portion of our program can change a value.
* Example – gumball machine

**Inheritance**

* Creating two or more classes that share a relationship to each other.
* (Parent/Super/Base Class) – Gives a field or method or property to a child/sub/derived class.
* Examples – a dog is a(n) animal, a Ford is a car

**Polymorphism**

* “Many Forms”
* A derived class can inherit functionality from a base class and define its own behavior in addition to the inherited behaviors.
* Overloading – accepting more than one set of parameters to accomplish a task.

**Access Modifiers**

o Control which region/block of our program can access the member/class/method

o Public – unlimited access – any part of the program can use this member.

o Private- only the containing/current class can access this member.

o Internal- access is limited to the current assembly.

o Protected- access is limited to the containing class and its derived (inherited) classes

* **Beginner**

1. Operators
2. Data Types and Keywords
3. Printing/Reading to/from the console
4. Utilizing looping structures (switch statement)
5. Arrays, Lists
6. Role of a ‘class’
7. Namespaces, modifiers, “using”, methods
8. Constructors and parameters (passing in)
9. Fields, Properties
10. Good Commenting and meaningful variable names
11. Concatenation

* **Intermediate**

1. Getters/Setters
2. Passing in a returned value from a method
3. Casting
4. Call Methods
5. Stack/Heap
6. Queues, Dictionary, etc.
7. Inheritance (and interfaces)
8. String manipulation
9. Parsing (and TryParse)
10. Exception Handling/Error Handling
11. Packages, others libraries

* **Advance**

1. APIs (Controllers)
2. SQL/MYSQL Database connections (.READ)
3. Interface
4. Separation of concerns