For OOP in Flutter :

1- **Class & Object**: Classes define templates; objects are instances.
2- **Encapsulation**: Data and methods are bundled, hiding implementation.
3- **Inheritance**: Subclasses inherit behavior from superclasses.
4- **Polymorphism**: Objects of different classes behave like superclass objects.
5- **Abstract Classes**: Serve as templates for other classes, cannot be instantiated.
6- **Interfaces**: Define required methods for classes to implement.
7- **Constructors**: Initialize object properties, with default and named types.
8- **Final/Const**: final can't be reassigned; const is a compile-time constant.
9- **Mixins**: Reuse code across classes without inheritance.
10- **Method Overriding**: Subclasses can customize inherited methods.

For Classes :

1- **Dart Classes**: Dart is object-oriented, using classes to create objects with data and methods.
2- **Declaring a Class**: Use the class keyword followed by the class name and its body.
3- **Constructor**: Initializes objects; can be parameterized or default.
4- **Named Constructors**: Define multiple constructors for a class.
5- **Getters and Setters**: Manage class attributes via the get and set keywords.
6- **Inheritance**: A class can inherit from another using extends, inheriting properties and methods.
7- **Method Overriding**: Child classes can override parent class methods using @override.
8- **Static Members**: Static fields and methods retain values throughout program execution.
9- **This Keyword**: Refers to the current instance to resolve name ambiguity.
10- **Super Keyword**: Refers to parent class methods or properties.