Nguyen Gia Binh - 104219428

Ex 1:

- Polymorphism enables objects of various classes to override differently to the same method. It eliminates the need to pick between classes during compilation and allows you to code for base functionality rather than individual types. Polymorphism encourages generalizing interfaces and writing code that can interact with multiple types in the background.

- In task 1, we created an abstract class call SummaryStategy with it two derived class AverageSummary and MinMaxSummary. Next, we the abstract class we have a abstract method call PrintSummary that can be override in both of it derived class. Furthermore, the DataAnalyser class use the SummaryStrategy class as it type so it can calls PrintSummary of both AverageSummary and MinMaxSummary

Ex 2:

- Abstraction involve hiding unnecessary implementation details and exposing only essential information through abstract interfaces. It distinguishes between the logical architecture of objects and the specifics of how they are implemented in classes. This allows higher-level functionality to operate with objects without knowing how they are made under the hood.

- Example: We are making a game with different type of vehicle. We make a abstract class “Vehicle” with abstract methods the represent the common features that every vehicel has like start, stop, brake, etc. Then we create classes that inherit “Vehicle” like Car class, Train class, etc. Each of these class then can implement their own way of methods of the abstract methods in “Vehicle”. In the end, we can create instances of different vehicle and treat them as Vehicle objects, invoking the common methods defined in the abstract base class.

Ex 3:

In the original design, each of the summary teachnique have its own variable. If we have 50 more summary teachniques we would have to add 50 more variables. As a result, many code would have to be unnecessary duplicated, reducing the efficiency and scalability as each summary teachnique would need its own parameters and ways of handling it.