What is your favourite design pattern, and why?

Abstract Factory is my recommend design pattern for coding, because it is easily to maintain, highly re-usability, use to readable. For example, I have different User type such as Employee, Children, Parent User, Admin, etc. I can easy to add a common fields or functions (email and updateEmail()) in User object or employee only fields (payRate) in employee object. I can also copy the User object pattern into the similar object type such as Finance object, under this object can have invoice, credit note, overpayment, etc.

For your favourite programming language, tell me about a new (or upcoming) language feature that has you excited. Why is it exciting for you?

My favourite programming language is C#. At the early stage mobile developing, you need to learn different and use different tool to develop mobile app for android and ios, it is very painfully. When I know I can use C# to develop the most of the android and ios app using Xamarin, I so happy about it and download the Xamarin Studio to play around. As Xamarin is growing, from Xamarin.Andriod and Xamarin.Ios to Xamarin.form, it is can easy and easy to develop mobile app for everyone.

What do you NOT like to see when you're reviewing your own or another colleague's code?

No comment of the code, not under the same tab space, no long line of code (try to use several lines for a long line of code), not to check the data/object is null ort not when get form database,

Tell me about a time you fixed a performance issue.

At the time developing the invoice manager that show all invoice and booking details, it uses the client paging that fetch all data at the same time, it works fine at the first year. When the invoice and booking number is getting large, the loading time is getting slower and slower, then I design to use server paging that only fetch only 10 - 20 records each time. The loading time have much improvement.

I also try to use toList() common to fetch a list of data especial a larger amount of data at the end of each function to reduce the loading and server resource.