# Questions

## Day 1

1. You have to validate user input for certain criteria like
   1. Correct format (phone number, email address)
   2. Maximum length
   3. Required field
   4. Cross-field validation
   5. Custom validations

The result should be a summary of all failed validations.

What pattern do you recommend? (Chain of Responsibility)

1. You create an application that uses a lot of images. You notice that the application became slow due to the loading of the images. You want to delay the loading until the images are actually needed.

What pattern do you recommend? (Proxy. Lazy loading scenario)

1. You have a system with all kind of products (e.g. Clothing, Electronics, Groceries). For each product there is a different tax regime or a price discount.

What pattern can you use to accommodate all the requirements? (Visitor. Products accept functionality)

1. For a web shop you are assigned the task of presenting customizable products (for example a car). The visitor should be able to configure properties like color, engine type, lightening, pilot assist etc.

What pattern do you recommend? (Prototype; Basic car with extra’s)

1. You’ve created a website based on a huge data set. To avoid overwhelming the user with data (and the browser and network) you decide to apply paginations.

What design pattern might be useful here? (Iterator. The iterator presents a certain amount of data)

1. You’ve assigned the task of keeping track of the entire state of the application (large object). It’s not necessary to persist the state. It should only exist on runtime.

What pattern do you recommend? (Singleton)

1. For your social media website you’re assigned the task of creating a system to notify users if their friends are doing something

What design pattern do you recommend? (Observer)

1. For a multi-user chess application with a huge player base you’re assigned the task to create a matchmaking system to find opponents with your ELO-rating and help to set up the match.

What pattern do you recommend? (Mediator)

## Day 2

1. A system dynamically applies permissions (e.g., read, write, delete) to users.

You’re asked to think about a solution. What pattern comes to mind? (Decorator)

1. An UI application supports multiple themes (e.g., Light, Dark), where each theme has its own styles for buttons, menus, and forms.

What pattern do you recommend here? (Abstract factory. Family of GUI elements)

1. Vehicles (e.g., Car, Truck, Motorcycle) can have different types of engines (e.g., Petrol, Diesel, Electric). You need to find a solution flexible enough to accommodate future engines and vehicles

What pattern do you recommend? (Bridge)

1. Different databases like MySQL, PostgreSQL, or MongoDB may need to be connected using the same interface, but the connection process differs.

What pattern might be useful here? (Factory Method)

1. A menu system in an application has nested menus and menu items. For example, a menu can contain submenus, and submenus can contain menu items or further submenus. At some point you need to display the menu.

What pattern do you recommend? (Composit)

1. Creating web request can be tricky. There are many options like Method (POST, GET etc) and a bunch of headers. All error prone because strings.

Is there a pattern that might help you out? (Builder)

1. You have legacy systems that you want to integrate with your new systems, but their interfaces don't match.

What pattern do you recommend? (Adapter)

## Day 3

1. You must design a model to handle documents from beginning to the moment it can be published. Between init en publish a user can be working on the document. Documents must also be reviewed before being published.

What pattern do you recommend? (State)

1. You’re assigned the task to write software for a programmable keyboard. Each button can be programmed separately to perform certain (predefined) task or execute a macro.

What design pattern might be useful here? (Command Pattern)

1. You must create a program that imports data from an external source. The steps are as follows:
   1. A source must be opened. You don’t know the exact source. The only thing you know that it creates a stream
   2. The stream must be processed. You don’t know the exact format of the data. Can be XML, JSON etc.
   3. The resulting data must be stored. Can be a database, a file or whatever.

What pattern might be useful here? (Template Method)

1. A travel company sells fully arranged trips. This involves multiple services like flight booking, hotel reservation, and car rental

With what pattern can you simplify the order process? (Façade)

1. For an image editor you’re assigned the task to keep a history of modifications on the images, like contrast, brightness, filter and more. At any time you want to be able to roll back to an earlier version.

What pattern do you recommend? (memento)

1. For a webs shop you’re assigned the task to create a payment system that allows certain payment options (CreditCard, PayPal, Ideal en future payment methods)

What design pattern can be useful here? (Strategy)

1. You need to create functionality on a map to present landmarks (e.g., restaurants, hotels, musea), the landmark is presented by an icon, geo location and name.

What pattern might be useful here? (flyweight)