*Solve three cases. Write feedback and recommendations for each of them.*

**Case 1**

Eva is working on the **XYZ project** as a senior developer. Her colleague Philip has just finished the producer/consumer module, the main feature responsible for managing orders on a large scale. Philip decided to implement this feature using some design patterns.

During the code review, Eva sees that the existing solution implements the functional requirements correctly but lacks unit tests, and it looks like the design patterns are used incorrectly.

While conducting the review, she also realizes that from time to time, her team members neglect to write unit tests, just like Philip. They say that they forget or just don't have time. Often, the team members also argue about what design patterns or code styling they need to use.

How should Eva respond to this review request?

**Feedback:**

Philip, great job on completing the producer/consumer module for the XYZ project! Your efforts in delivering the functional requirements are commendable, and I appreciate the dedication you've shown in implementing this crucial feature.

**Recommendations:**

1. Since there are lack of unit testing in project, Work with the team to define and document coding standards and guidelines. This includes clear expectations for unit testing, design patterns, and coding style. Having a well-documented set of standards provides a reference for all team members and reduces ambiguity.

## **Reference in the course:** Code review fundamentals from Module 1, Lesson 1

1. Introduce pair programming sessions, where team members work together on writing code. This fosters knowledge transfer, allows immediate feedback, and can improve the overall quality of the code. It also provides an opportunity for less experienced team members to learn from their more experienced peers.

**Reference in the course:** Pair programming practice from Module 3, Lesson 1

1. Since this is not the first-time team members have implemented the pattern incorrectly or haven't implemented it at all, this situation requires additional attention. For that let's improve the coding standards, or this mistake should be covered in a separate checklist.

**Reference in the course**: Module 2, Lesson 1: Code review strategy (about coding standards); Module 3, Lesson 1: Checklists practice

**Case 2**

Alex is working on the **XYZ project** as a junior developer. He has just finished his task, where he did a lot of refactoring of the old authentication module and added a new feature for sorting orders in the shopping cart. He prefers to send his code only to Michael, the lead developer, for review because he respects him and thinks he can learn a lot from his comments.

However, Michael is overloaded with such requests because other team members also think like Alex. For Michael, it's also hard to understand what was implemented in a pull request because not all of his colleagues described the change well. Some of them created a mess in the commit history.

After 30 minutes of investigating Alex's pull request, Michael sees that the review will take too long because it will involve a lot of refactoring and implementation of new features. It would probably take more than 2 hours to do just a cursory check.

What should Michael do, and how should he comment on this review?

**Feedback:**

Firstly, great job on completing the XYZ project task! I appreciate your efforts in refactoring the authentication module and adding the new sorting feature to the shopping cart. I've started reviewing your pull request, and it seems like a substantial piece of work.

**Recommendations:**

1. Alex, I Suggest breaking down future tasks into smaller, more manageable units. This not only makes the review process more straightforward but also allows for better tracking of changes and understanding of their individual impacts. **Reference in the course:** Code Review Quality Control from Module 3, Lesson 3
2. I recommend you follow the Use Instant Code Review Approach as you are developing a lot of refactoring of the old authentication module and adding a new feature for sorting orders in the shopping cart. With the help of this approach helps you and team learn, share, advance knowledge, and continuously improve your code base together and whole team have any idea what is going on this part of the task and later go to the higher reviewer so that process will consume less time and will be effective.

**Reference in the course:** Code review best practice Module 3, Lesson 1

**Case 3**

Sandra is working on the **XYZ project** as a senior developer on a distributed team. Her working hours overlap with most of her team members by only 2 hours since they are in the UK. Her junior colleague Oliver (located in Bristol) has just finished an architecturally important task.

Initially, the implementation details were discussed with the whole team, and Oliver volunteered to work on the task. It was a great opportunity for him to learn new coding approaches during a feature implementation. Oliver has spent six days developing the feature.

During the code review, Sandra noticed that most classes and methods were not organized as decided by the team six days ago. There are a lot of other fixes she needs to discuss with Oliver too.

What should Sandra do, and what should she mention in her feedback?

**Feedback:**

I hope this message finds you well. Firstly, I want to express my appreciation for your dedication and hard work on the recent architecturally important task for the XYZ project. Your commitment to the project is evident, and I value the effort you put into it. There are several positive aspects of your work that I'd like to highlight. However, during the code review, I observed some deviations from the coding standards and organizational structure that we discussed as a team six days ago.

**Recommendations:**

1. Oliver, I recommend Setting Communication Guidelines for the entire team. With this, the entire team will be on the same page, as the working time zone is different for each developer, and with communication guidelines, the whole team will be aware of what changes are needed.

**Reference in the course:** Code Review Best Practices from Module 3, Lesson1 Set Communication Guidelines

1. I can see the most classes and methods were not organized so I recommend you follow the coding standards, Coding standards are rules, guidelines, techniques, and best practices for creating cleaner, more readable, and more efficient code with minimal errors. Coding standards encourage developers to follow rules based on project and company requirements rather than on their experience.

**Reference in the course:** Setting up the code review process from Module 3, L esson 1 coding standard.

1. As you are working on some important architectural changes, please be sure to put less than 200 lines of code changes in per review and follow some future tasks into smaller, more manageable units in each review. If you won't follow this approach, the ability to find bugs may increase.

## **Reference in the course:** Code Review Best Practices from Module 3, Lesson 1