

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Lecture 3: Design Patterns I

Today's Plan

- 1st hour:
 - Quiz 3 (assessed, correctness)
 - Abstraction açade, Factory
- Break <https://eduassistpro.github.io/>
- 2nd hour: Add WeChat edu_assist_pro
 - PI 3 (assessed, participation)
 - Player-Role, General Hierarchy, Delegation, Adaptation,
- Long break
- 13:00-14:00 OR 14:00-15:00 in BO1028: Lab 3 Design Patterns Part 1

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Quiz 3: Design Patterns

QUIZ

- Go to YACRS
 - <https://eduassistpro.github.io/>
- Join session : **1171**
 - prepare for quiz.
 - 120 sec per question (I'll call out 60 sec, 30 sec, 5 sec)

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Q3.1 Design Patterns

Design patterns have a standard description format composed of 7 key elements. Which of the following options IS NOT one of these elements?

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

- A: Solution
- B: Anti-patterns
- C: Context
- D: Discussion

Q3.1 Design Patterns I (solution)

TODO **Assignment Project Exam Help**

Design patterns have 7 key elements. Which of these elements? <https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

- A: Solution ← The recommended way to solve the problem
- B: Anti-patterns ← Incorrect applications of the pattern
- C: Context ← The general situation where a pattern should be used.
- D: Discussion

Q3.2: Design Patterns II

Assignment Project Exam Help

Patterns should normally be illustrated using a simple diagram. The

<https://eduassistpro.github.io/>

- A: Follow a formal syntax [†] used in all pattern diagrams **Add WeChat edu_assist_pro**
- B: Have an informal syntax with elements from UML
- C: Do not have any standard syntax
- D: Patterns are not represented with diagrams

Q2.2: Design Patterns II (solution)

Assignment Project Exam Help

Patterns should normally be illustrated using a simple diagram. The

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

- B: Have an informal syntax nts from UML
- For example, see Figure 6.1 and 6.2 in your book.

Q3.3: Abstraction Occurrence

Assignment Project Exam Help

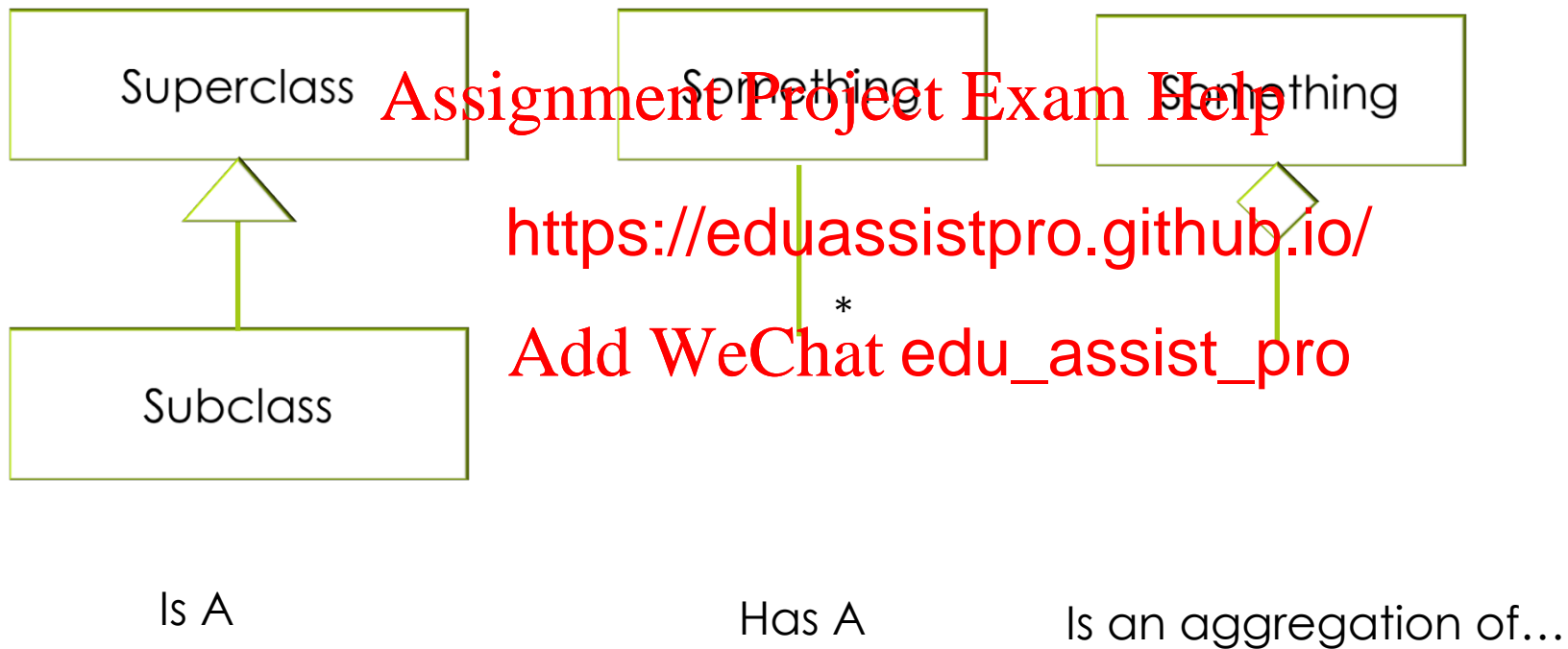
The abstraction occurrence pattern is an example of a:

<https://eduassistpro.github.io/>

- A: “is a” relationship
- B: “Many-to-many” relationship
- C: “One-to-one” relationship
- D: “has a” relationship

Add WeChat edu_assist_pro

Q3.3: Abstraction-Occurance



“Is a” and “has a” relationships

Design patterns (in this course)

This week:

- Abstraction-Occurrence

- Façade

- Singleton

- The Factory

- Player-Role

- Delegation

- Adapter

- General Hierarchy

Next week:

- Observer

mutable

Factory-Only Interface

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

For Each Pattern

Assignment Project Exam Help

- You must understand its context, problem, forces, solution, a representation.

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

- You should have a good understanding such that you could write a basic implementation.

Design Patterns Gameplan

- Design Patterns are important, we'll spend two weeks on this chapter
- Now:
 - Abstraction, Occurrence, Strategy, Factory, Façade

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Abstraction Occurrence

- Context: A set of related object that we will call occurrences, the members of such a set share common information but differ in other important ways.
 - Episodes in a television series
 - Flights offered
 - Series of lectures
- Solution: Create an abstraction that contains the data that is common to all the members of a set of occurrences. Then create an Occurrence class representing the occurrence of this abstraction.
- Anti-patterns: Using a single class with inheritance.

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Abstraction Occurrence

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Abstraction Occurrence Antipattern

- Why is the following use of abstraction occurrence incorrect?

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro



Singleton

- Context: Use in scenarios where you only want one instance to exist.

- Company

- MainWindow

- Solution: A private class variable stores the single instance. A static method returns the instance. The constructor is private.

- Note: Pattern should not be overused.

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

```
<<Singleton>>
```

```
theInstance
```

```
getInstance()
```

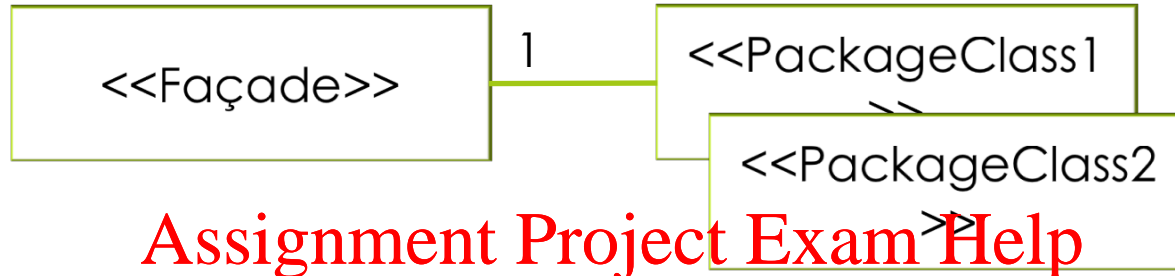
Singleton

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Façade



Assignment Project Exam Help

<https://eduassistpro.github.io/>

- Context: On complex packages, a program with such packages has to manage many different classes.
 - Airline uses a large number of classes which could be simplified
- Solution: Create a façade class, which provided a set of public methods that simplifies the usage of the package.

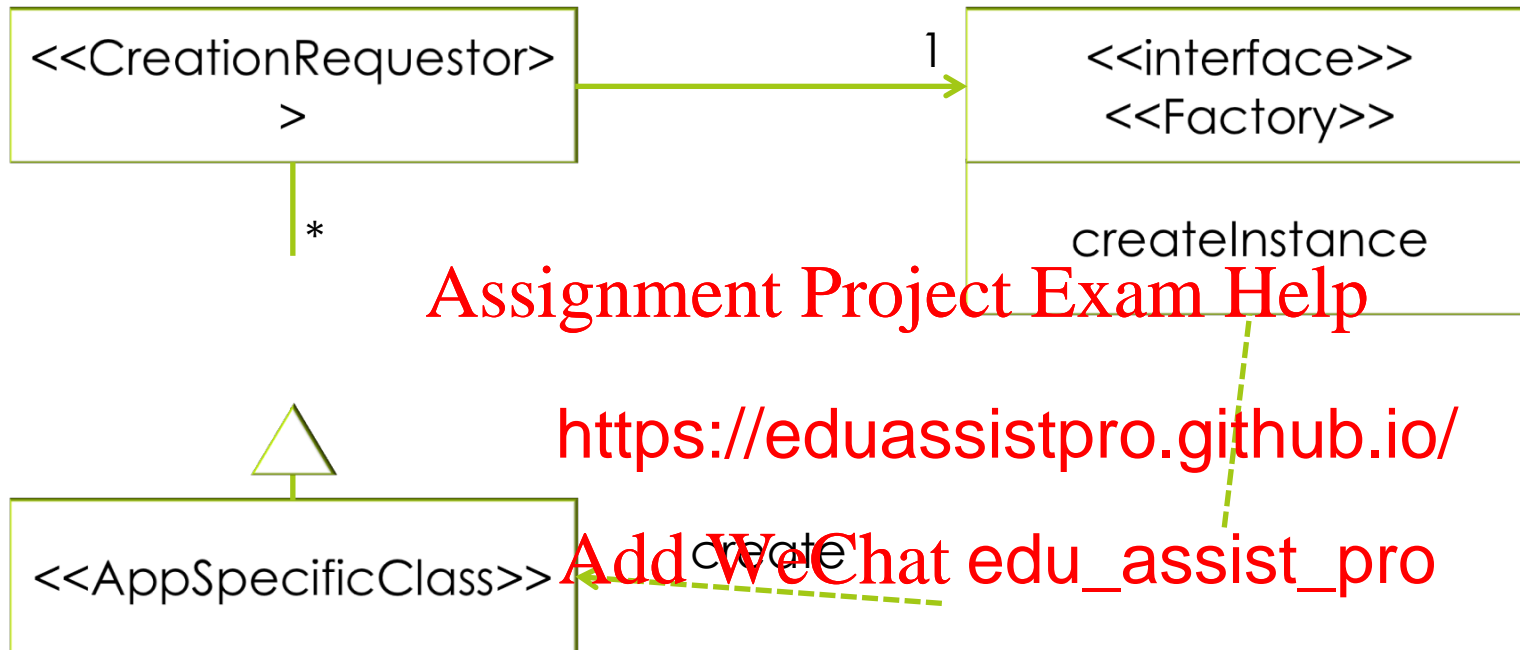
Façade

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Factory



- Context: You have a **framework** that needs to create objects as part of its work. However, the class of the created objects will depend on the application.
- MediaFileManager framework which needs to make different kinds of media files

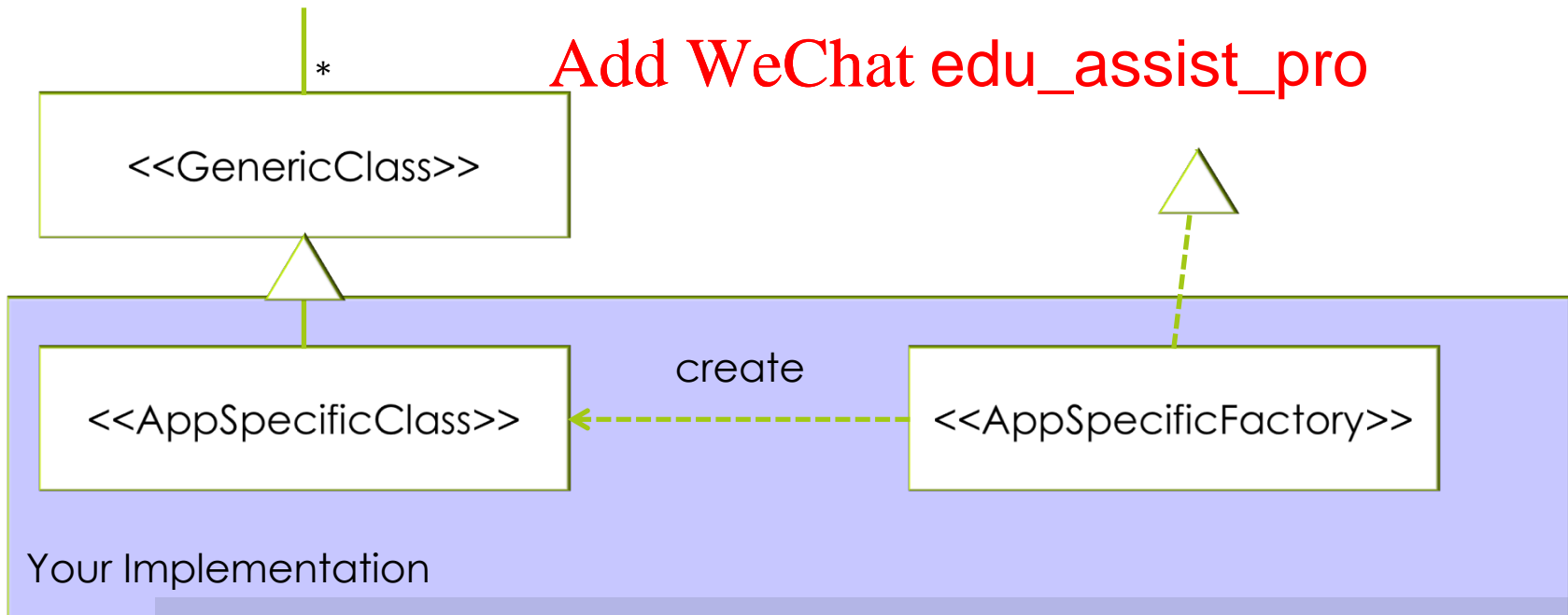
Factory Pattern

Framework

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro



Factory Pattern

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Factory

- Solution: The framework delegates creation of **specific classes** to the factory implement <https://eduassistpro.github.io/>. The **specific class** extends a general class defined in the framework. Add WeChat edu_assist_pro

PI 3: Questions

- Join YACRS Now! **Assignment Project Exam Help**
- Join Sessio **<https://eduassistpro.github.io/>**
Add WeChat edu_assist_pro

PI 3.1: Singleton Pattern

Assignment Project Exam Help

The Singleton pattern ensures other classes can't create multiple instances of a class by:

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

- ❑ A: Not using a constructor
- ❑ B: Using a private constructor
- ❑ C: Using a protected constructor
- ❑ D: Using a public constructor

PI 3.1: Singleton Pattern (solution)

- The Singleton pattern ensures other classes can create multiple instances of the Singleton object by:

<https://eduassistpro.github.io/>

- A: Not using a constructor
- B: Using a private constructor <– Only a private constructor ensure other classes can't create multiple instantiations of the Singleton object
- C: Using a protected constructor
- D: Using a public constructor

PI3.2: Singleton Pattern

Assignment Project Exam Help
A system for managing a list of company contacts keeps each company in the database using the Singleton pattern. This way, a company's record is available throughout the system.

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

A: This is a good use of the Singleton Pattern.

B: This is a good use of the Singleton Pattern, but it introduces unnecessary complexity.

C: This is a bad use of the Singleton Pattern, and it introduces unnecessary complexity.

D: This is a bad use of the Singleton Pattern, but it will functionally work as the database of companies expands.

PI3.2: Singleton Pattern (solution)

A system for managing a list of company contacts keeps each company in the database using the Singleton pattern. This way, a company's records are not duplicated throughout the system.

A: This is a good use of the Singleton Pattern

Assignment Project Exam Help
Are Singletons conceptually what we need?

B: This is a good use of complexity.

<https://eduassistpro.github.io/>

There is unnecessary coupling because of this pattern.

Add WeChat edu_assist_pro

C: This is a bad use of the Singleton Pattern, and it introduces unnecessary complexity.

Correct.

D: This is a bad use of the Singleton Pattern, but it will functionally work as the database of companies expands.

Imagine how maintaining this class will work. How will you add new companies? You would need to implement a new Class every time you wanted to add a company. If you feel like you have to fight the pattern to make it work, it might not be the right pattern.

PI3.3: Façade Pattern

How many of the following statements about the Façade pattern are true:

- The Façade pattern simplifies a complex group of packages.
- Once the Façade changes to how to make small changes to how to
- A Façade class is useful because it reduces coupling.

A: 0

B: 1

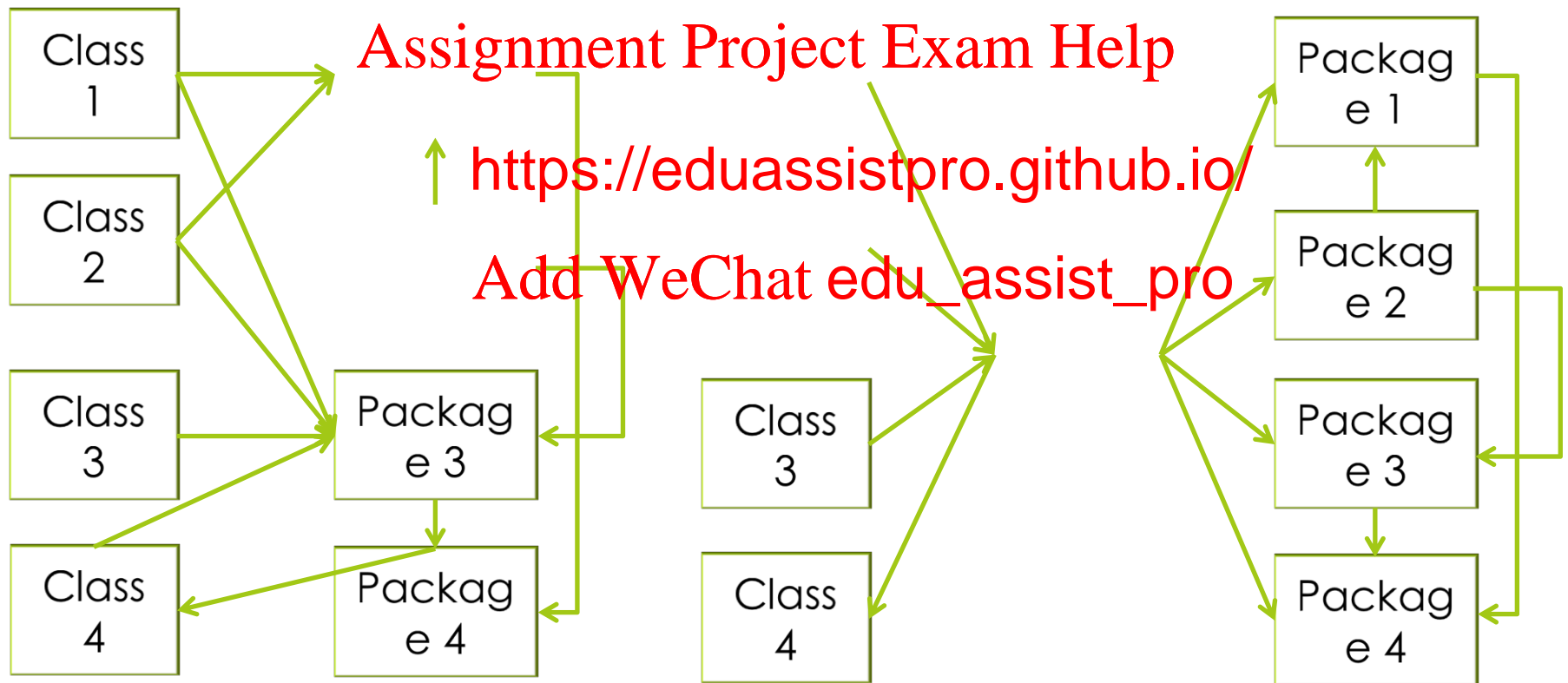
C: 2

D: 3

PI3.3: Façade Pattern (solution)

The Façade pattern simplifies a complex group of packages.

Once the Façade is defined, it can be easier to make small changes to how to packages work together.



Design Patterns (cont.)

■ Now: **Assignment Project Exam Help**

■ Player Ro

eral Hierarchy

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Player-Role

■ **Context:**

- A role is a particular set of properties associated with an object in a particular context.

- An object may have multiple contexts.

- **Problem:** How do you best model players so that a player can change roles or possess multiple roles?

■ **Forces:**

- It is desirable to improve encapsulation by capturing the information associated with each separate role in a class.
- You want to avoid multiple inheritance.
- You cannot allow an instance to change class

Player-Role

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Player-Role

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

The Delegation Pattern

- **Context:**
 - You are
 - You reali <https://eduassistpro.github.io/> method which provides the required service
 - Inheritance is not appropri **Add WeChat edu_assist_pro**
- **Problem:**
 - How can you most effectively make use of a method that already exists in the other class?

The Delegation Pattern

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

The Delegation Pattern

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

The Adaptor Pattern

- **Context:**
 - You are incorporating a new technology and want to reuse an existing class.
 - The reused class is also often part of its own inheritance hierarchy.
- **Problem:**
 - How to obtain the power of polymorphism when reusing a class whose methods have the same function but not the same signature

The Adaptor Pattern

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

The Adaptor Pattern

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

General Hierarchy

■ Context: Assignment Project Exam Help

- Objects i or more objects above them (superordinates). Some objects have any subordinates (subordinates). Some objects have any subordinates
- How do you represent a hierarchy of objects, in which some objects cannot have subordinates?

- Problem: How do you represent a hierarchy of objects, in which some objects cannot have subordinates?

General Hierarchy

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

General Hierarchy

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

PI Questions

■ Join YACKS session: 1172 **Assignment Project Exam Help**

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

PI 3.4: Player Role

Assignment Project Exam Help

The Player-Role pattern is related to the abstraction occurrence.

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

A: They both use inheritance.

B: They both use abstract classes.

C: They both relate abstractions to real world entities.

D: They both use interfaces.

PI 3.4: Player Role (solution)

Assignment Project Exam Help

The Player-Role pattern is related to the abstraction occurrence.

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

A: They both use inheritance.

B: They both use abstract classes.

C: They both relate abstractions to real world entities.

D: They both use interfaces.

PI 3.5: General Hierarchy

Assignment Project Exam Help

The general hierarchy organizes classes into a hierarchy structure. This

<https://eduassistpro.github.io/>

A: Many to Many relationship

Add WeChat edu_assist_pro

B: Inheritance Relationship

C: Superior/Subordinate Nodes

D: Abstract Relationship

PI 3.5: General Hierarchy (solution)

The general hierarchy organises classes into a hierarchy structure. This represents an **Assignment Project Exam Help**

A: Many to Many relationship **<https://eduassistpro.github.io/>**

B: Inheritance Relationship ← This is tern **Add WeChat edu_assist_pro**

C: Superior/Subordinate Nodes

D: Abstract Relationship

PI 3.6: Adaptor Pattern

Assignment Project Exam Help

The adaptor pattern allows you to incorporate an existing class into your code. It is useful because:

<https://eduassistpro.github.io/>

A: It encourages you to write abstract classes.

Add WeChat edu_assist_pro

B: It simplifies code using polymorphism.

C: It prevents incorrect usage of inheritance.

D: It uses interface polymorphism.

PI 3.6: Adaptor Pattern (solution)

The adaptor pattern allows you to incorporate an existing class into your inheritance hierarchy. This is useful because:

A: It encourages y <https://eduassistpro.github.io/>

B: It reuses code using polymorphis
Add WeChat edu_assist_pro

C: It prevents incorrect usage of inheritance. ← You can still use inheritance incorrectly with this pattern.

D: It uses interface polymorphism.

Lab 3

Assignment Project Exam Help

- We'll implement some design patterns in Java.

- See you in <https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Next Week (i.e. week 4)

Assignment Project Exam Help

- Remaining 4 patterns (OOSE Chapter 6).
- Review all <https://eduassistpro.github.io/>

Add WeChat edu_assist_pro