

# Professional Online Academy Vu Topper RM



## CS603-Software Architecture and Design Update MCQ'S Mid Term



Follow Us on YouTube:  
[@vutopperrm](https://www.youtube.com/@vutopperrm)

For More Help Contact What's app Me!!!  
Number's = # 0322-4021365, 0316-4980094

[Rizwan Manzoor](https://www.facebook.com/RizwanManzoor)  
[Vu Topper RM](https://www.facebook.com/VuTopperRM)

وَتَعَزُّ مَنْ تَشَاءُ وَتَذِلُّ مَنْ تَشَاءُ



### Our Services

- Assignments
- Quiz & Gdb's
- LMS Handling
- Online Classes
- Lectures Views
- Project Handling
- Important Notes

**Result**

85% To  
100%

پہلے ہمارے سٹوڈنٹس کے رزلٹ دیکھیں  
پھر ہم سے سروسز لیں

[www.youtube.com/@vutopperrm](https://www.youtube.com/@vutopperrm)



Contact us  
0322 4021365  
0316 4980094

بری صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365

**Question No:1**

**(Marks:1)**

**Vu-Topper RM**

By using strategy design pattern, you can change the behavior of context by \_\_\_\_\_ it with different objects.

A. Encapsulating

**B. Aggregating**

C. Associating

D. Inheriting

**Question No:2**

**(Marks:1)**

**Vu-Topper RM**

You want to avoid multiple inheritance. Which design pattern would you choose?

A. Singleton Pattern

B. General Hierarchy Pattern

C. Abstraction-Occurrence Pattern

**D. Player-Role Pattern**

**Question No:3**

**(Marks:1)**

**Vu-Topper RM**

Reflexive associations are an association to connect a class to \_\_\_\_\_.

**A. Itself**

B. Others

C. One another

D. Each other

**Question No:4**

**(Marks:1)**

**Vu-Topper RM**

In case of a tree data structure, all \_\_\_\_\_ have the same structure and behavior.

A. Classes

B. Objects

C. Variables

**D. Nodes**

بري صحبت سے تنہائی بہتر ہے اور تنہائی سے نيك صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:5**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ deal with decoupling interface and implementation of classes and objects.

- A. Behavioural patterns
- B. Structural patterns**
- C. Architectural patterns
- D. Creational patterns

**Question No:6**

**(Marks:1)**

**Vu-Topper RM**

Facade Design Pattern is a \_\_\_\_\_.

- A. Observer Pattern
- B. Creational Pattern
- C. Structural Pattern**
- D. Behavioral Pattern

**Question No:7**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ should be followed if you want to avoid duplicating data.

- A. General Hierarchy Pattern
- B. Facade Pattern
- C. Abstraction-Occurrence Pattern**
- D. Player Role Pattern

**Question No:8**

**(Marks:1)**

**Vu-Topper RM**

Fowler's book assigns "names" to refactoring in the same way that the \_\_\_\_\_ book assigned names to patterns.

- A. GoF's**
- B. K. N. Thompson
- C. William C. Wake
- D. Steven Metsker

برى صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:9**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ defines a family of algorithms, encapsulates each one, and makes them interchangeable. Strategy lets the algorithm vary independently from clients that use it.

**A. Strategy Pattern**

B. MVC Pattern

C. Design Pattern

D. Singleton Pattern

**Question No:10**

**(Marks:1)**

**Vu-Topper RM**

In the specialization a \_\_\_\_\_ can be thought of an attribute that will have a different value in each subclass.

A. Ticket

B. Tag

**C. Discriminator**

D. Marker

**Question No:11**

**(Marks:1)**

**Vu-Topper RM**

The object which depends on a subject and updates according to its subject's state is called \_\_\_\_\_.

A. Linker

B. Loader

C. State Machine

**D. Observer**

**Question No:12**

**(Marks:1)**

**Vu-Topper RM**

Design pattern \_\_\_\_\_ is the abstract description of design problems and how the pattern solves it.

A. Element

B. Name

**C. Solution**

بري صحبت سے تہائی بہتر ہے اور تہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

D. Problem

**Question No:13**

**(Marks:1)**

**Vu-Topper RM**

The benefit of the state pattern is that state-specific logic is localized in \_\_\_\_\_ that represent that state.

**A. Classes**

B. Functions

C. Structures

D. Arrays

**Question No:14**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ deal with initializing and configuring classes and objects.

**A. Creational Patterns**

B. Structural Patterns

C. Bridge Patterns

D. Behavioural Patterns

**Question No:15**

**(Marks:1)**

**Vu-Topper RM**

In \_\_\_\_\_ the ends of the association are semantically different from each other, even though the associated class is the same.

**A. Asymmetric Reflexive Association**

B. Symmetric Reflexive Association

C. Reflexive Association

D. Symmetric Association

**Question No:16**

**(Marks:1)**

**Vu-Topper RM**

In state design patterns, by \_\_\_\_\_ each state in a class, we localize any changes that need to be made.

برى صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

- A. Inheriting
- B. Associating
- C. Aggregating
- D. Encapsulating**

**Question No:17** (Marks:1) **Vu-Topper RM**

\_\_\_\_\_ deal with dynamic interactions among societies of classes and objects.

Options:

- A. Creational patterns
- B. Architectural patterns
- C. Behavioural patterns**
- D. Structural patterns

**Question No:18** (Marks:1) **Vu-Topper RM**

At the time of refactoring, the \_\_\_\_\_ behavior of the \_\_\_\_\_ does not change.

- A. Internal, Object
- B. Internal, System
- C. External, Object
- D. External, System**

**Question No:19** (Marks:1) **Vu-Topper RM**

How many bad smells fall in Object-orientation abusers?

- A. Three
- B. Five
- C. Six
- D. Four**

برى صحبت سے تھائی بہتر ہے اور تھائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:20**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ is a creational design pattern which solves the problem of creating product objects without specifying their concrete classes.

- A. Strategy pattern
- B. Factory method**
- C. State pattern
- D. Template method

**Question No:21**

**(Marks:1)**

**Vu-Topper RM**

Higher cohesion makes sure that changes made to the design of a system should be \_\_\_\_\_ in effect.

- A. General
- B. Modular
- C. Global
- D. Local**

**Question No:22**

**(Marks:1)**

**Vu-Topper RM**

Classes should depend on abstractions, not on implementations.

- A. Dependency inversion principle**
- B. Open-close principle
- C. Liskov substitution principle
- D. Interface segregation principle

**Question No:23**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ creation is not polymorphic.

- A. Object**
- B. Class
- C. Entity
- D. Array

بري صحبت سے تھائی بہتر ہے اور تھائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:24**

**(Marks:1)**

**Vu-Topper RM**

In \_\_\_\_\_, both the entities are dependent on each other.

- A. Abstraction
- B. Inheritance
- C. Composition**
- D. Aggregation

**Question No:25**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ is one in which a new class is created (derived class) that inherits the features from the already existing class (base class).

- A. Inheritance**
- B. Aggregation
- C. Composition
- D. Polymorphism

**Question No:26**

**(Marks:1)**

**Vu-Topper RM**

According to Open-Closed Principle (OCP) a software artifact should be open for extension but closed for \_\_\_\_\_.

- A. Coupling
- B. Inversion
- C. Creation
- D. Modification**

**Question No:27**

**(Marks:1)**

**Vu-Topper RM**

Subtypes should be faithful to parent types.

- A. Liskov Substitution**
- B. Single responsibility principle
- C. Dependency Inversion
- D. Open-Close principle

بري صحبت سے تہائی بہتر ہے اور تہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:28**

**(Marks:1)**

**Vu-Topper RM**

In programming, the \_\_\_\_\_ is a way to decouple software modules.

A. Both Interface Segregation & Dependency Inversion Principle

**B. Dependency Inversion Principle**

C. Digression Inversion Principle

D. Interface Segregation Principle

**Question No:29**

**(Marks:1)**

**Vu-Topper RM**

In an open/close principle, system should be open to \_\_\_\_\_, but close to \_\_\_\_\_.

**A. Extension, modification**

B. Modification, close

C. Modification, extension

D. Extension, open

**Question No:30**

**(Marks:1)**

**Vu-Topper RM**

In \_\_\_\_\_, parent owns child entity so child entity can't exist without parent entity.

A. Abstraction

**B. Composition**

C. Inheritance

D. Aggregation

**Question No:31**

**(Marks:1)**

**Vu-Topper RM**

The major goal of Liskov Substitution Principle (LSP) is to ensure

\_\_\_\_\_.

A. Usability

**B. Proper Inheritance**

C. Reduce inter-module coupling

بري صحبت سے تنہائی بہتر ہے اور تنہائی سے نيك صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

D. Security

**Question No:32**

**(Marks:1)**

**Vu-Topper RM**

Wherever a polymorphic call is made, compiler inserts code to first look for \_\_\_\_\_.

A. vtable

**B. vptr**

C. vfirst\_look

D. vcode\_firstlook

**Question No:33**

**(Marks:1)**

**Vu-Topper RM**

A lecturer may be assigned a course to teach. That lecturer may be removed from that course at any time. The lecturer is added to the course and not vice-versa is a/ an example of \_\_\_\_\_.

**A. Unidirectional Association**

B. Dependency

C. Bidirectional Association

D. Aggregation

**Question No:34**

**(Marks:1)**

**Vu-Topper RM**

The main advantage of \_\_\_\_\_ over \_\_\_\_\_ is that it allows objects of the derived class to be dynamically bound in the same hierarchy as its base classes.

A. Inheritance, Composition

B. Inheritance, Aggregation

C. Polymorphism, Aggregation

**D. Polymorphism, Composition**

بري صحبت سے تنہائی بہتر ہے اور تنہائی سے نيك صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:35**

**(Marks:1)**

**Vu-Topper RM**

According to "Jacobson's Heuristic" Policy information should not be placed \_\_\_\_\_ involved in the policy decision.

A. Outside the interfaces

**B. inside the domain**

C. Inside of classes

D. Outside of classes

**Question No:36**

**(Marks:1)**

**Vu-Topper RM**

Vptr is a pointer to vtable. It is maintained per \_\_\_\_\_.

A. Method

B. Class

C. Function

**D. Object**

**Question No:37**

**(Marks:1)**

**Vu-Topper RM**

Association can be achieved by using \_\_\_\_\_.

**A. Pointers**

B. Loops

C. Variables

D. Integer

**Question No:38**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ and \_\_\_\_\_ are used in pattern designing.

A. Polymorphism, Aggregation

B. Inheritance, Composition

**C. Inheritance, Polymorphism**

D. Composition, Aggregation

بري صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:39**

**(Marks:1)**

**Vu-Topper RM**

Classes that have \_\_\_\_\_ interfaces are classes whose interfaces are not cohesive.

**A. Fat**

B. Thin

C. Thick

D. Slim

**Question No:40**

**(Marks:1)**

**Vu-Topper RM**

Suppose, you have only one line of code in the system. Identify the option that is not suitable for one line code system.

**A. It will be easy to understand.**

B. It will be easy to fix.

C. It will be easy to amendable to future changes.

D. Bugs will be hard to discover.

**Question No:41**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ is a table of functions pointers.

Options:

**A. vtable**

B. vpointer\_table

C. vptr

D. vfunctions\_pointers

**Question No:42**

**(Marks:1)**

**Vu-Topper RM**

When Coupling is \_\_\_\_\_, cohesion tends to be \_\_\_\_\_ and vice versa.

A. Low, low

B. High, low

**C. Low, high**

D. High, high

بري صحبت سے تنہائی بہتر ہے اور تنہائی سے نيك صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:43**

**(Marks:1)**

**Vu-Topper RM**

Software Design is not a/an \_\_\_\_\_ process.

- A. iterative
- B. incremental
- C. emergent
- D. deterministic**

**Question No:44**

**(Marks:1)**

**Vu-Topper RM**

“Output” of the Software Design Process is a \_\_\_\_\_.

- A. Computer Software
- B. Blueprint for implementation**
- C. System Specification
- D. Tool

**Question No:45**

**(Marks:1)**

**Vu-Topper RM**

When a child object is created, a corresponding \_\_\_\_\_ object/s are also created.

- A. One more Child
- B. Parent**
- C. Both Parent and Child
- D. Two more child

**Question No:46**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ is to make or draw plans for something.

- A. Design**
- B. Engineering
- C. Science
- D. Coding

بري صحبت سے تہائی بہتر ہے اور تہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:47**

**(Marks:1)**

**Vu-Topper RM**

Design techniques do not rely on \_\_\_\_\_.

**A. repeatable processes**

B. rules-of-thumb

C. heuristics and rules-of-thumb

D. heuristics

**Question No:48**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ Modules are more reusable

**A. Low coupled**

B. Highly coupled

C. Highly cohesive

D. Low cohesive

**Question No:49**

**(Marks:1)**

**Vu-Topper RM**

Object oriented design is a/an \_\_\_\_\_.

**A. High level approach**

B. Bottom up approach

C. Action oriented approach

D. Top down approach

**Question No:50**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ coupled modules are less dependent upon each other and hence are easier to change.

A. Moderately

B. Reasonably

**C. Loosely**

D. Highly

برى صحبت سے تنہائی بہتر ہے اور تنہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:51** (Marks:1) **Vu-Topper RM**  
In Hierarchical Organization \_\_\_\_\_ components may be composed of \_\_\_\_\_ components.

- A. Smaller,Larger
- B. Medium,Larger
- C. Larger,Smaller**
- D. Small,Medium

**Question No:52** (Marks:1) **Vu-Topper RM**  
In \_\_\_\_\_ one instance of B is associated with many instances of A.

- A. Inheritance
- B. Composition
- C. Association**
- D. Aggregation

**Question No:53** (Marks:1) **Vu-Topper RM**  
Coupling usually correlates with \_\_\_\_\_.

- A. maintenance
- B. deployment
- C. cohesion**
- D. testing

**Question No:54** (Marks:1) **Vu-Topper RM**  
The \_\_\_\_\_ serves as a technical plan for system implementation.

- A. Requirements Engineering
- B. Software Testing
- C. Software Design process**
- D. System Specification

بري صحبت سے تہائی بہتر ہے اور تہائی سے نیک صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

Question No:55

(Marks:1)

Vu-Topper RM

\_\_\_\_\_ programs are difficult to understand and modify.

- A. Standard
- B. Better designed
- C. Poorly designed**
- D. Basic

Question No:56

(Marks:1)

Vu-Topper RM

The cardinality of Inheritance relationship is \_\_\_\_\_.

- A. 1:1 and 1:m only
- B. 1:m only**
- C. 1:1 only
- D. 1:1, 1:m and m:m

Question No:57

(Marks:1)

Vu-Topper RM

While mapping \_\_\_\_\_, each class of the whole-part hierarchy is mapped to a separate table.

- A. Association
- B. Composition**
- C. Polymorphism
- D. Inheritance

Question No:58

(Marks:1)

Vu-Topper RM

Structured design is a/an \_\_\_\_\_.

- A. Top down approach**
- B. Bottom up approach
- C. Medium level approach
- D. Center level approach

بري صحبت سے تہائی بہتر ہے اور تہائی سے نیک صحبت بہتر ہے

For More Help Contact What's app 03224021365

**Question No:59**

**(Marks:1)**

**Vu-Topper RM**

In \_\_\_\_\_, each class of the inheritance hierarchy is mapped to a separate table.

A. Horizontal Mapping

**B. Vertical Mapping**

C. One-to-one Mapping

D. Filtered Mapping

**Question No:60**

**(Marks:1)**

**Vu-Topper RM**

Complexities that are inherent in the problem are known as \_\_\_\_\_.

**A. Essential complexities**

B. Incidental complexities

C. Total complexities

D. Accidental complexities

**Question No:61**

**(Marks:1)**

**Vu-Topper RM**

In the case of mapping Association, for \_\_\_\_\_ relationship, a separate table is used to maintain the information about the relationship.

A. self-joining

**B. many-to-many**

C. one-to-one

D. one-to-many

**Question No:62**

**(Marks:1)**

**Vu-Topper RM**

Without associations, there is only a set of unconnected \_\_\_\_\_.

A. Classes

**B. Objects**

C. Methods

D. Functions

بري صحبت سے تنہائی بہتر ہے اور تنہائی سے نيك صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Question No:63**

**(Marks:1)**

**Vu-Topper RM**

\_\_\_\_\_ represents a hierarchy of abstraction in which one class inherits from one or more super-classes.

A. Polymorphism

**B. Inheritance**

C. Encapsulation

D. Design

**Question No:64**

**(Marks:1)**

**Vu-Topper RM**

In object-oriented design, \_\_\_\_\_ is a/an central concept.

**A. Object**

B. Method

C. Structured

D. Attribute

**Question No:65**

**(Marks:1)**

**Vu-Topper RM**

An instance of the \_\_\_\_\_ cannot be created.

A. Control Class

B. Sub Class

C. Super Class

**D. Abstract Class**

بري صحبت سے تنہائی بہتر ہے اور تنہائی سے نيك صحبت بہتر ہے

**For More Help Contact What's app 03224021365**

**Visit My YouTube Channel**  
**For Subjective and More**  
**Important Files**  
**Channel Name = #VuTopperRM**

برى صءبء سے ءءهائى بهءر هے اور ءءهائى سے نيك صءبء بهءر هے

For More Help Contact What's app 03224021365