

- Q1.** जावा में *पॉलीमॉर्फिज़्म* को सबसे अच्छी तरह से कौन सा कथन समझाता है?
- (A) एक क्लास में कई कंस्ट्रक्टर रखने की क्षमता
  - (B) एक मेथड में अलग-अलग क्लास में अलग-अलग नाम रखने की क्षमता
  - (C) एक ऑब्जेक्ट में कई रूप धारण करने की क्षमता, जैसे कि एक ही इंटरफ़ेस मेथड के लिए कई इम्प्लीमेंटेशन रखना
  - (D) उपयोगकर्ता से इम्प्लीमेंटेशन विवरण छिपाने की क्षमता

**Which statement best explains *polymorphism* in Java?**

- (A) Ability of a class to have multiple constructors
- (B) Ability of a method to have different names in different classes
- (C) Ability of an object to take many forms, such as having multiple implementations for the same interface method
- (D) Ability to hide implementation details from the user

- Q2.** निम्नलिखित कोड स्निपेट पर विचार करें, और सही उत्तर चुनें:

```
try {  
    throw new IOException("IO Error");  
} catch (Exception e) {  
    System.out.println("Caught Exception");  
} finally {  
    System.out.println("Finally executed");  
}
```

- (A) Compilation error
- (B) केवल "Caught Exception"
- (C) केवल "Finally executed"
- (D) "Caught Exception" के बाद "Finally executed"

**Consider the following code snippet, and choose the correct answer:**

```
try {  
    throw new IOException("IO Error");  
} catch (Exception e) {  
    System.out.println("Caught Exception");  
} finally {  
    System.out.println("Finally executed");  
}
```

- (A) Compilation error
- (B) "Caught Exception" only
- (C) "Finally executed" only
- (D) "Caught Exception" followed by "Finally executed"

**Q3.** मान लीजिए कि आपके पास लिस्ट <Employee> है, जहां Employee की फ़ील्ड्स id और salary हैं। जावा स्ट्रीम्स का उपयोग करते हुए, निम्नलिखित लाइन्स ऑफ़ स्टेटमेंट में से कौन सी एक उन कर्मचारियों के वेतन का योग प्राप्त करने के लिए सही है जिनका वेतन 80,000 से अधिक है?

**Consider you have a List<Employee> where Employee has fields id and salary. Which one of the following lines of statement using Java Streams is correct to get the sum of salaries for employees whose salary is above 80,000?**

- (A) `double total = employees.stream().filter(e → e.getSalary() > 80000).mapToDouble(Employee::getSalary).sum();`
- (B) `double total = employees.stream().filter(e → e.getSalary() < 80000).mapToDouble(Employee::getSalary).sum();`
- (C) `double total = employees.streams().filter(e → e.getSalary() > 80000).mapToDouble(Employee::getSalary).sum();`
- (D) कोई भी विकल्प नहीं /None of the options

**Q4.** जावा में वोलेटाइल के बारे में निम्नलिखित में से कौन सा कथन सही है?

- (A) यह ऑपरेशनों की एटामिसिटी की गारंटी देता है
- (B) यह सिंक्रोनाइज़्ड कीवर्ड के उपयोग के समतुल्य है
- (C) यह निर्देशों के पुनर्व्यवस्थापन को रोकता है और विधियों के समकालिकीकरण को सुनिश्चित करता है
- (D) यह थ्रेड्स में परिवर्तनों की दृश्यता की गारंटी देता है

**Which of the following statements about volatile in Java is true?**

- (A) It guarantees atomicity of operations
- (B) It is equivalent to using synchronized keyword
- (C) It prevents instruction reordering and ensures method synchronization
- (D) It guarantees visibility of changes across threads

Q5. निम्नलिखित कोड स्निपेट पर विचार करें:

```
AtomicInteger count = new AtomicInteger(0);
ExecutorService executor = Executors.newFixedThreadPool(2);
for (int i = 0; i < 1000; i++) {
    executor.submit(() -> count.incrementAndGet());
}
executor.shutdown();
System.out.println(count.get());
```

सबसे अधिक संभावित आउटपुट क्या है?

- (A) 1000 (B) 0 और 1000 के बीच  
(C) 0 (D) कंपाइलेशन एरर

Consider the following code snippet:

```
AtomicInteger count = new AtomicInteger(0);
ExecutorService executor = Executors.newFixedThreadPool(2);
for (int i = 0; i < 1000; i++) {
    executor.submit(() -> count.incrementAndGet());
}
executor.shutdown();
System.out.println(count.get());
```

What is the most likely output?

- (A) 1000 (B) Between 0 and 1000  
(C) 0 (D) Compilation Error

Q6. जावा में कौन सा मेमोरी एरिया मेथड-लोकल वैरिएबल्स को स्टोर करता है?

- (A) हीप (B) मेथड एरिया  
(C) स्टैक (D) नेटिव मेथड स्टैक

Which memory area stores method-local variables in Java?

- (A) Heap (B) Method Area  
(C) Stack (D) Native Method Stack

Q7. नॉन-ब्लॉकिंग आई/ओ ऑपरेशन के लिए किस जावा नियो क्लास का उपयोग किया जाता है?

- (A) बफर्डरीडर (B) फ़ाइलइनपुटस्ट्रीम  
(C) सॉकेटचैनल (D) फ़ाइल

Which Java NIO class is used for non-blocking I/O operations?

- (A) BufferedReader (B) FileInputStream  
(C) SocketChannel (D) File

Q8. निम्नलिखित कोड स्निपेट का आउटपुट क्या होगा?

What will be the output of the following code snippet?

```
Stream.of(1, 2, 3, 4).filter(n -> n % 2 == 0).map(n -> n * n)
    .forEach(System.out::print);
```

- (A) 24 (B) 44  
(C) 4164 (D) 416

Q9. पैरलल स्ट्रीम के बारे में कौन सा कथन सही है?

- (A) वे हमेशा अनुक्रमिक स्ट्रीम की तुलना में तेज़ परिणाम देते हैं
- (B) वे डिफ़ॉल्ट रूप से एलिमेंटों के एंकाउंटर क्रम को बनाए रखते हैं
- (C) यदि म्यूटेबल डेटा स्ट्रुक्चर के साथ उपयोग किया जाए तो वे थ्रेड-सुरक्षा समस्याएँ पैदा कर सकते हैं
- (D) वे प्रत्येक एलिमेंट ऑपरेशन को एक ही थ्रेड में निष्पादित करते हैं

**Which statement about parallel streams is correct?**

- (A) They always produce faster results than sequential streams
- (B) They maintain the encounter order of elements by default
- (C) They can cause thread-safety issues if used with mutable data structures
- (D) They execute each element's operation in a single thread

Q10. यदि वैल्यू एब्सेंट है तो ऑप्शनल का कौन सा मेथड *NoSuchElementException* थ्रो करता है?

**Which method of Optional throws *NoSuchElementException* if the value is absent?**

- (A) get()
- (B) orElse()
- (C) orElseGet()
- (D) ifPresent()

Q11. सर्वलेट्स में कौन सा ऑब्जेक्ट रिक्वेस्ट-रिस्पॉन्स साइकल में क्लाइंट रिक्वेस्ट का निरूपण करता है?

**Which object in Servlets represents the client request in the Request-Response cycle?**

- (A) ServletContext
- (B) ServletConfig
- (C) HttpServletResponse
- (D) HttpServletRequest

Q12. जेएसपी में, निम्नलिखित में से कौन सा एक स्क्रिप्टिंग एलिमेंट का सही ढंग से निरूपण करता है जो हर बार पेज के रिक्वेस्ट किए जाने पर जावा कोड निष्पादित करता है?

**In JSP, which of the following correctly represents a scripting element that executes Java code every time the page is requested?**

- (A) `<%! %>`
- (B) `<jsp:useBean>`
- (C) `<%= %>`
- (D) `<% %>`

Q13. जेपीए में, कौन सा एनोटेशन यह सुनिश्चित करता है कि एंटीटी फ्रील्ड डेटाबेस में बनी नहीं रहती है लेकिन एंटीटी ऑब्जेक्ट में अभी भी उपलब्ध है?

**In JPA, which annotation ensures that an entity field is not persisted in the database but still available in the entity object?**

- (A) @Id
- (B) @Column(nullable=false)
- (C) @Transient
- (D) @Embedded

**Q14.** स्प्रिंग में कस्टम बीन पोस्ट-प्रोसेसिंग के लिए कौन सा इंटरफ़ेस लागू किया गया है?

- (A) बीनफ़ैक्टरी (B) बीनपोस्टप्रोसेसर  
(C) एप्लिकेशनकॉन्टेक्स्टअवेयर (D) इनिशियलाइज़िंगबीन

**Which interface is implemented for custom bean post-processing in Spring?**

- (A) BeanFactory (B) BeanPostProcessor  
(C) ApplicationContextAware (D) InitializingBean

**Q15.** कौन सा एनोटेशन किसी विशिष्ट यूआरआई के लिए एचटीटीपी जीईटी रिक्वेस्ट को हैंडल करने के लिए स्प्रिंग मेथड को चिह्नित करता है?

**Which annotation marks a Spring method to handle HTTP GET requests for a specific URI?**

- (A) @RequestParam  
(B) @PathVariable  
(C) @Controller  
(D) @GetMapping

**Q16.** स्प्रिंग बूट में डिफ़ॉल्ट एम्बेडेड सर्वलेट कंटेनर क्या है?

- (A) जेटी  
(B) अंडरटो  
(C) टॉमकैट  
(D) ग्लासफ़िश

**What is the default embedded servlet container in Spring Boot?**

- (A) Jetty  
(B) Undertow  
(C) Tomcat  
(D) GlassFish

**Q17.** कौन सा मेथड सिग्रेचर जेपीए रिपोजिटरी मेथड पर पेजिनेशन सक्षम करता है?

**Which method signature enables pagination on a JPA repository method?**

- (A) List<User> findAll();  
(B) Page<User> findAll(Pageable pageable);  
(C) User getById(Long id);  
(D) Optional<User> findByName(String name);

**Q18.** कौन सा एनोटेशन स्प्रिंग एप्लिकेशन में मेथड-स्तरीय सुरक्षा को सक्षम करता है?

**Which annotation enables method-level security in a Spring application?**

- (A) @EnableWebSecurity  
(B) @Secured  
(C) @EnableGlobalMethodSecurity  
(D) @PreAuthorize

Q19. स्प्रिंग क्लाउड माइक्रोसर्विसेस एनवायरनमेंट में नेटफ्लिक्स यूरेका की क्या भूमिका है?

- (A) एपीआई गेटवे (B) सर्विस डिस्कवरी  
(C) सर्किट ब्रेकर (D) वितरित ट्रेसिंग

**What is the role of Netflix Eureka in a Spring Cloud microservices environment?**

- (A) API Gateway (B) Service Discovery  
(C) Circuit Breaker (D) Distributed Tracing

Q20. कौन सा एचटीटीपी स्टेटस कोड इंगित करता है कि RESTful रिसोर्स सफलतापूर्वक बनाया गया है?

**Which HTTP status code indicates that a RESTful resource has been successfully created?**

- (A) 200 (B) 201  
(C) 206 (D) 204

Q21. विभिन्न ट्रांजेक्शन मैनेजर में से, सिस्टम में किसी ट्रांजेक्शन की ड्यूरैबिलिटी प्रॉपर्टी की गारंटी के लिए कौन ज़िम्मेदार है?

- (A) ट्रांजेक्शन मैनेजर  
(B) कंकरेंसी कॉन्ट्रोल मैनेजर  
(C) रिकवरी मैनेजर  
(D) एप्लीकेशन प्रोग्रामिंग मैनेजर

**Among the different transaction managers, which one is responsible for guaranteeing the durability property of a transaction in the system?**

- (A) Transaction Manager  
(B) Concurrency Control Manager  
(C) Recovery Manager  
(D) Application Programming Manager

Q22. एक डेटाबेस टेबल T पर विचार करें जिसमें दो कॉलम X और Y हैं, जिनमें से प्रत्येक पूर्णांक प्रकार का है। टेबल क्रिएट करने के बाद, टेबल में एक रिकॉर्ड (X=1, Y=1) इन्सर्ट किया जाता है। मान लीजिए MX और MY किसी भी समय टेबल में सभी रिकॉर्डों में से X और Y के क्रमशः अधिकतम मान दर्शाते हैं। MX और MY का उपयोग करते हुए, टेबल में 128 बार नए रिकॉर्ड इन्सर्ट किए जाते हैं, जिनमें X और Y के मान क्रमशः MX+1, 2 × MY+1 होते हैं। यह ध्यान देने योग्य है कि हर बार इन्सर्ट करने के बाद, MX और MY के मान बदल जाते हैं। ऊपर बताए गए चरणों को पूरा करने के बाद, निम्नलिखित एसक्यूएल क्वेरी का आउटपुट क्या होगा?

`SELECT Y FROM T WHERE X=7;`

**Consider a database table T containing two columns X and Y each of type Integer. After the creation of the table, one record (X=1, Y=1) is inserted in the table. Let MX and MY denote the respective maximum values of X and Y among all records in the table at any point in time. Using MX and MY, new records are inserted in the table 128 times with X and Y values being MX+1, 2 × MY+1 respectively. It may be noted that each time after the insertion, values of MX and MY change. What will be the output of the following SQL query after the steps mentioned above are carried out?**

`SELECT Y FROM T WHERE X=7;`

- (A) 127 (B) 126  
(C) 256 (D) 255

- Q23.** एसक्यूएल में किस प्रकार का इंडेक्स किसी टेबल में असमान रूप से वितरित डेटा की खोज को तेज़ करता है?
- (A) बिटमैप इंडेक्स
  - (B) हैश इंडेक्स
  - (C) बी-ट्री इंडेक्स
  - (D) फुल-टेक्स्ट इंडेक्स

**Which type of index in SQL speeds up searches for data that is distributed non-uniformly across a table?**

- (A) Bitmap Index
- (B) Hash Index
- (C) B-Tree Index
- (D) Full-text Index

- Q24.** एसक्यूएल ऑप्टिमाइज़ेशन में निष्पादन योजना का विश्लेषण क्वेरी प्रदर्शन को बेहतर बनाने में कैसे मदद कर सकता है?
- (A) यह डेटाबेस में रो की संख्या दिखाता है
  - (B) यह क्वेरी रिज़ल्ट को अपडेट करता है
  - (C) यह डुप्लिकेट डेटा डिलीट करता है
  - (D) यह बॉटलनेक्स, मिसिंग इंडेक्स और अनावश्यक टेबल स्कैन की पहचान करता है

**How can analyzing an execution plan in SQL optimization help improve query performance?**

- (A) It shows the number of rows in a database
- (B) It updates query results
- (C) It deletes duplicate data
- (D) It identifies bottlenecks, missing indexes, and unnecessary table scans

- Q25.** रिलेशनल डेटाबेस में फ़ॉरेन की के लिए कौन सा कथन सही है?
- (A) यह अपने टेबल में रिकॉर्ड्स की विशिष्ट पहचान करता है
  - (B) यह केवल स्वयं को रिफरेंस कर सकता है
  - (C) यह केवल एक विशिष्ट कॉन्स्ट्रेंट के रूप में कार्य करता है
  - (D) यह किसी अन्य टेबल में प्राइमरी की को संदर्भित कर सकता है

**Which statement is true for a foreign key in a relational database?**

- (A) It uniquely identifies records in its own table
- (B) It can only reference itself
- (C) It acts only as a unique constraint
- (D) It may reference a primary key in another table

**Q26. डेटाबेस ट्रिगर किस परिदृश्य में सबसे उपयुक्त है?**

- (A) जब आप किसी विशिष्ट ईवेंट पर कई टेबलों में बिजनेस लॉजिक को स्वचालित रूप से लागू करना चाहते हैं
- (B) क्लाइंट-साइड वैलिडेशन राइट करते समय
- (C) नया इंडेक्स डिज़ाइन करते समय
- (D) यूजर इंटरफ़ेस परिवर्तनों के लिए

**In which scenario is a database trigger most appropriate?**

- (A) When you want to enforce business logic across multiple tables automatically on a specific event
- (B) When writing client-side validation
- (C) When designing a new index
- (D) For user interface changes

**Q27. एसक्यूएल में संग्रहीत कार्यविधियों का उपयोग करने का मुख्य लाभ क्या है?**

- (A) यूजर इंटरफ़ेस रेंडरिंग
- (B) केरी ऑप्टिमाइजेशन
- (C) अनुक्रमणिका निर्माण
- (D) डेटाबेस इंजन में पुनः प्रयोज्य, सुरक्षित और अनुकूलित लॉजिक एक्जेक्यूशन

**What is the primary advantage of using stored procedures in SQL?**

- (A) User interface rendering
- (B) Query optimization
- (C) Index creation
- (D) Reusable, secure, and optimized logic execution in the database engine

**Q28. निम्नलिखित में से कौन सा डॉक्यूमेंट-ओरिएंटेड एनओएसक्यूएल डेटाबेस है?**

- (A) कैसैंड्रा
- (B) मोंगोडीबी
- (C) एमवाईएसक्यूएल
- (D) रेडिस

**Which of the following is a document-oriented NoSQL database?**

- (A) Cassandra
- (B) MongoDB
- (C) MySQL
- (D) Redis

**Q29. रिलेशनल डेटाबेस में रेफरेंशियल इंटीग्रिटी लागू करने का मुख्य उद्देश्य क्या है?**

- (A) सुनिश्चित करें कि टेबलों के बीच संबंध सुसंगत रहें
- (B) यूआई में सुधार
- (C) केरी की गति बढ़ाना
- (D) स्टोरेज उपयोग को सीमित करना

**What is the main purpose of enforcing referential integrity in a relational database?**

- (A) Ensure relationships between tables remain consistent
- (B) Improve UI
- (C) Increase query speed
- (D) Limit storage usage

**Q30.** एसक्यूएल में कौन सा आइसोलेशन स्तर केवल कमिटेड डेटा को रीड करने की अनुमति देता है, जिससे डर्टी रीड को रोका जा सकता है?

- (A) रीड अनकमिटेड (B) रीड कमिटेड  
(C) रिपीटेबल रीड (D) सीरियलाइज़ेबल

**Which isolation level in SQL allows reads of committed data only, preventing dirty reads?**

- (A) Read Uncommitted (B) Read Committed  
(C) Repeatable Read (D) Serializable

**Q31.** कौन सा एचटीएमएल5 एलिमेंट डेवलपर्स को वेब पेज के भीतर आकार, चित्र और एनिमेशन जैसे ग्राफिक्स को प्रोग्रामेटिक रूप से प्रस्तुत करने की अनुमति देता है?

**Which HTML5 element allows developers to programmatically render graphics such as shapes, images, and animations within a web page?**

- (A) <canvas>  
(B) <svg>  
(C) <img>  
(D) <figure>

**Q32.** उपलब्ध स्पेस (निश्चित साइजिंग के बाद, यदि कोई हो) तीन कॉलमों में कैसे वितरित किया जाएगा?

- (A) प्रत्येक कॉलम की चौड़ाई बिल्कुल समान होगी  
(B) पहले कॉलम को तीसरे कॉलम से दोगुना स्पेस मिलेगा, और दूसरे कॉलम को तीसरे कॉलम से तीन गुना स्पेस मिलेगा  
(C) पहले कॉलम को आधा स्पेस मिलेगा, दूसरे कॉलम को एक चौथाई स्पेस मिलेगा, और तीसरे कॉलम को एक चौथाई स्पेस मिलेगा  
(D) कॉलमों के बीच स्थान यादृच्छिक रूप से वितरित किया जाता है

**How will the available space (after fixed sizing, if any) be distributed across the three columns?**

- (A) Each column will be exactly the same width  
(B) The first column gets twice the space of the third column, and the second column gets three times the space of the third column  
(C) The first column gets half the space, the second column gets a quarter, and the third column gets a quarter  
(D) Space is distributed randomly among the columns

Q33. निम्नलिखित कोड स्निपेट पर विचार करें:

```
let value = 'outer';  
if (true) {  
  let value = 'inner';  
  console.log(value);  
}  
console.log(value);
```

उपरोक्त कोड स्निपेट का आउटपुट क्या होगा?

**Consider the following code snippet:**

```
let value = 'outer';  
if (true) {  
  let value = 'inner';  
  console.log(value);  
}  
console.log(value);
```

**What will be the output of the above code snippet?**

- (A) inner and inner
- (B) outer and outer
- (C) inner and outer
- (D) Error

Q34. जावास्क्रिप्ट में प्रॉमिस के बिहेवियर के बारे में निम्नलिखित में से कौन सा कथन सही है?

- (A) .then() मेथड को एसिंक्रोनस परिणामों को संभालने के लिए चेन किया जा सकता है
- (B) componentDidMount, componentDidUpdate, componentWillUnmount
- (C) जावास्क्रिप्ट में प्रॉमिस डिफॉल्ट रूप से कोड को सिंक्रोनस रूप से चलाते हैं
- (D) प्रॉमिस का उपयोग केवल एजेएक्स रिक्वेस्ट के साथ किया जा सकता है

**Which of the following is true about the behavior of Promises in JavaScript?**

- (A) The .then() method can be chained to handle asynchronous results
- (B) componentDidMount, componentDidUpdate, componentWillUnmount
- (C) Promises in JavaScript run code synchronously by default
- (D) Promises can only be used with AJAX requests

Q35. आप रिएक्ट फ़ंक्शनल कंपोनेंट में **useEffect Hook** का उपयोग निम्नलिखित में से किस परिदृश्य में करेंगे?

- (A) लोकल स्टेट वैरिएबल डिक्लेयर करना
- (B) मेमोइज़ेशन के साथ रेंडरिंग प्रदर्शन को अनुकूलित करना
- (C) डेटा प्राप्त करने या इवेंट श्रोताओं को सेट अप करने जैसे साइड-इफेक्ट्स निष्पादित करना
- (D) कस्टम हुक बनाना

**Which of the following scenarios would you use the useEffect Hook for in a React functional component?**

- (A) Declaring local state variables
- (B) Optimizing rendering performance with memoization
- (C) Performing side-effects such as fetching data or setting up event listeners
- (D) Creating custom hooks

**Q36. Fetch API के बारे में निम्नलिखित में से कौन सा कथन सही है?**

- (A) डेटा को सिंक्रोनस रूप से रिटर्न करता है
- (B) XMLHttpRequest का उपयोग करता है
- (C) एक प्रॉमिस रिटर्न करता है जो एक रिस्पांस ऑब्जेक्ट में बदल जाता है
- (D) केवल GET अनुरोधों के लिए उपयोग किया जा सकता है

**Which of the following statements about the Fetch API is correct?**

- (A) Returns data synchronously
- (B) Uses XMLHttpRequest under the hood
- (C) Returns a Promise that resolves to a Response object
- (D) Can only be used for GET requests

**Q37. RESTful APIs के संदर्भ में, एचटीटीपी स्टेटस कोड 404 आमतौर पर क्या दर्शाता है?**

- (A) रिक्वेस्टेड रिसोर्स सर्वर पर नहीं मिला
- (B) क्लाइंट रिसोर्स तक पहुँचने के लिए अधिकृत नहीं है
- (C) रिक्वेस्ट सफल रहा और डेटा रिटर्न हो जाता है
- (D) रिक्वेस्ट संसाधित करते समय सर्वर में एक आंतरिक त्रुटि आई

**In the context of RESTful APIs, what does the HTTP status code 404 typically indicate?**

- (A) The requested resource could not be found on the server
- (B) The client is not authorized to access the resource
- (C) The request was successful and data is returned
- (D) The server encountered an internal error while processing the request

**Q38. WebSockets का आधार कौन सा प्रोटोकॉल है?**

- (A) एचटीटीपी
- (B) एफटीपी
- (C) टीसीपी
- (D) यूडीपी

**Which protocol is the foundation of WebSockets?**

- (A) HTTP
- (B) FTP
- (C) TCP
- (D) UDP

**Q39. कौन सा SASS सिंटैक्स गोलाकार कोनों के लिए मिक्सिन को सही ढंग से परिभाषित और उपयोग करता है?**

**Which SASS syntax correctly defines and uses a mixin for rounded corners?**

- (A) @rounded(10px);
- (B) function rounded(\$radius) { border-radius: \$radius; }
- (C) mixin rounded(\$radius) { border-radius: \$radius; } then rounded(10px);
- (D) @mixin rounded(\$radius) { border-radius: \$radius; } then @include rounded(10px);

**Q40. निम्नलिखित में से कौन से DevTools पैनल नेटवर्क अनुरोधों को डीबग करने में मदद करते हैं?**

- (A) परफॉर्मेंस
- (B) एलिमेंट्स
- (C) नेटवर्क
- (D) कंसोल

**Which of the following DevTools panel helps debug network requests?**

- (A) Performance
- (B) Elements
- (C) Network
- (D) Console

**Q41. मोनोलिथिक आर्किटेक्चर की तुलना में माइक्रोसर्विस आर्किटेक्चर का मुख्य लाभ क्या है?**

- (A) एकल इकाई के रूप में सरल परिणियोजन
- (B) संचार के लिए कम संसाधनों की आवश्यकता होती है
- (C) अतिरिक्त उपकरणों के बिना एंड-टू-एंड परीक्षण करना आसान है
- (D) प्रत्येक सर्विस को स्वतंत्र रूप से विकसित और परिणियोजित किया जा सकता है

**What is a primary advantage of Microservices architecture over Monolithic architecture?**

- (A) Simpler deployment as a single unit
- (B) Requires fewer resources for communication
- (C) Easier to test end-to-end without additional tools
- (D) Each service can be developed, deployed independently

**Q42. इवेंट-ड्रिवेन आर्किटेक्चर में, घटकों के बीच संचार का प्राथमिक तंत्र क्या है?**

- (A) प्रत्यक्ष सिंक्रोनस मेथड कॉल
- (B) शेयर्ड मेमोरी
- (C) एसींक्रोनस रिएक्शन को ट्रिगर करने वाली घटनाएँ
- (D) रिमोट प्रोसीजर कॉल (आरपीसी)

**In an Event-Driven architecture, what is the primary mechanism by which components communicate?**

- (A) Direct synchronous method calls
- (B) Shared memory
- (C) Events that trigger asynchronous reactions
- (D) Remote Procedure Calls (RPC)

**Q43. एक स्ट्रिक्ट लेयर्ड आर्किटेक्चर में, यदि ऊपरी लेयर को किसी गैर-आसन्न निचली लेयर के साथ सीधे संचार करने की अनुमति दी जाए, तो क्या होगा?**

- (A) प्रदर्शन में सुधार होता है, लेकिन मॉड्यूलरिटी कम हो जाती है
- (B) मॉड्यूलरिटी और प्रदर्शन दोनों में सुधार होता है
- (C) मॉड्यूलरिटी में सुधार होता है, लेकिन प्रदर्शन कम हो जाता है
- (D) न तो मॉड्यूलरिटी और न ही प्रदर्शन प्रभावित होता है

**In a strict Layered Architecture, what happens if an upper layer is allowed to directly communicate with a non-adjacent lower layer?**

- (A) Performance improves but modularity is reduced
- (B) Both modularity and performance improve
- (C) Modularity improves but performance reduces
- (D) Neither modularity nor performance is affected

**Q44. कौन सा डिज़ाइन पैटर्न यह सुनिश्चित करता है कि किसी क्लास में केवल एक ही इंस्टेंस हो, जबकि एक ग्लोबल पहुँच बिंदु प्रदान करता है?**

- (A) एडाप्टर
- (B) सिंगलटन
- (C) ऑब्ज़र्वर
- (D) फैक्टरी

**Which design pattern ensures a class has only one instance while providing a global point of access?**

- (A) Adapter
- (B) Singleton
- (C) Observer
- (D) Factory

**Q45.** कॉम्प्लेक्स ऑब्जेक्ट निर्माण प्रक्रिया को चरण-दर-चरण निर्माण से बदलने के लिए कौन सा डिज़ाइन पैटर्न सबसे उपयुक्त होगा?

- (A) प्रोटोटाइप (B) बिल्डर  
(C) स्ट्रेटजी (D) डेकोरेटर

**Which design pattern would be best to replace a complex object construction process with step-by-step creation?**

- (A) Prototype (B) Builder  
(C) Strategy (D) Decorator

**Q46.** कौन सा सॉलिड सिद्धांत कहता है कि "सॉफ्टवेयर इकाइयाँ विस्तार के लिए खुली होनी चाहिए, लेकिन संशोधन के लिए बंद होनी चाहिए"?

- (A) सिंगल रेस्पॉन्सिबिलिटी (B) ओपेन/क्लोज्ड  
(C) लिस्कोव पसब्टीट्यूशन (D) डिपेंडेंसी इंवर्जन

**Which SOLID principle states that "software entities should be open for extension, but closed for modification"?**

- (A) Single Responsibility (B) Open/Closed  
(C) Liskov Substitution (D) Dependency Inversion

**Q47.** लिस्कोव प्रतिस्थापन सिद्धांत मुख्यतः क्या सुनिश्चित करता है?

- (A) सुपरक्लास के ऑब्जेक्ट्स को बस के बिना सबक्लास ऑब्जेक्ट्स द्वारा प्रतिस्थापित किया जा सकता है  
(B) सबक्लास में मेथड्स के सिग्नेचर सुपरक्लास से भिन्न होने चाहिए  
(C) क्लासेस को एब्सट्रैक्शन के बजाय सॉलिड क्लासेस पर निर्भर होना चाहिए  
(D) क्लाइट्स को उन इंटरफेस पर निर्भर होने के लिए बाध्य नहीं किया जाना चाहिए जिनका वे उपयोग नहीं करते हैं

**What does the Liskov Substitution Principle primarily ensure?**

- (A) Objects of a superclass can be replaced by subclass objects without bugs  
(B) Methods in subclass should have different signatures than superclass  
(C) Classes should depend on concrete classes rather than abstractions  
(D) Clients should not be forced to depend on interfaces they do not use

**Q48.** कौन सा सिद्धांत इंटरफेस को तोड़ने पर ज़ोर देता है ताकि क्लाइट को केवल उनके द्वारा उपयोग की जाने वाली विधियों के बारे में ही पता हो?

- (A) इंटरफेस सेग्रीगेशन (B) डिपेंडेंसी इंवर्जन  
(C) सिंगल रेस्पॉन्सिबिलिटी (D) ओपेन/क्लोज्ड

**Which principle emphasizes breaking down interfaces so clients only need to know about the methods they use?**

- (A) Interface Segregation (B) Dependency Inversion  
(C) Single Responsibility (D) Open/Closed

**Q49.** कौन सा प्रोटोकॉल मुख्य रूप से उच्च-प्रदर्शन, कम-विलंबता संचार के लिए डिज़ाइन किया गया है जिसका उपयोग आमतौर पर माइक्रोसर्विसेस में किया जाता है?

- (A) REST (B) gRPC  
(C) आरपीसी ओवर एचटीटीपी (D) मैसेज ब्रोकर्स

**Which protocol is designed primarily for high-performance, low-latency communication commonly used in Microservices?**

- (A) REST (B) gRPC  
(C) RPC over HTTP (D) Message Brokers

**Q50.** कौन सा माइक्रोसर्विस संचार तंत्र आमतौर पर उत्पादकों और उपभोक्ताओं को अलग करने के लिए टॉपिक्स और क्यू का उपयोग करता है?

- (A) REST  
(B) आरपीसी  
(C) मैसेज ब्रोकर्स (काफ़ा, RabbitMQ)  
(D) gRPC

**Which microservices communication mechanism typically uses topics and queues to decouple producers and consumers?**

- (A) REST  
(B) RPC  
(C) Message Brokers (Kafka, RabbitMQ)  
(D) gRPC

**Q51.** जीआईटी में "feature" नामक एक नई ब्रांच बनाने के लिए कौन सा कमांड इस्तेमाल किया जाता है?

- (A) git ब्रांच फीचर (B) git चेकआउट फीचर  
(C) git मर्ज फीचर (D) git कमिट फीचर

**What command creates a new branch named "feature" in Git?**

- (A) git branch feature (B) git checkout feature  
(C) git merge feature (D) git commit feature

**Q52.** निम्नलिखित में से कौन जीआईटी में मर्ज और रीबेस के बीच अंतर को सबसे अच्छी तरह से वर्णित करता है?

- (A) मर्ज टॉप पर कमिट्स को रिप्ले करता है, रीबेस ब्रांचों को जोड़ता है  
(B) रीबेस कॉन्फ्लिक्ट को इग्नोर करता है, मर्ज नहीं  
(C) दोनों बिल्कुल एक ही ऑपरेशन करते हैं  
(D) मर्ज हिस्ट्री को जोड़ता है, रीबेस कमिट्स को रैखिक रूप से पुनः लागू करता है

**Which of the following best describes the difference between merge and rebase in Git?**

- (A) Merge replays commits on top, rebase combines branches  
(B) Rebase ignores conflicts, merge does not  
(C) Both perform the exact same operation  
(D) Merge combines histories, rebase reapplies commits linearly

**Q53. सीआई/सीडी पाइपलाइन में, "build" स्टेज का प्राथमिक उद्देश्य क्या है?**

- (A) सोर्स कोड को निष्पादन योग्य बाइनरी में कंपाइल करना
- (B) एप्लिकेशन को प्रोडक्शन में परिनियोजित करना
- (C) यूनिट टेस्ट चलाना
- (D) सिस्टम स्वास्थ्य की निगरानी करना

**In a CI/CD pipeline, what is the primary purpose of the "build" stage?**

- (A) Compile source code into executable binaries
- (B) Deploy application to production
- (C) Run unit tests
- (D) Monitor system health

**Q54. निरंतर डेप्लायमेंट के बारे में कौन सा कथन सही है?**

- (A) इसके लिए प्रत्येक डेप्लायमेंट से पहले मैनुअल अनुमोदन की आवश्यकता होती है
- (B) यह प्रत्येक सफल कोड परिवर्तन को स्वचालित रूप से उत्पादन में परिनियोजित करता है
- (C) यह केवल डेटाबेस परिवर्तनों पर लागू होता है
- (D) यह निरंतर इंटीग्रेशन के समान है

**Which statement about Continuous Deployment is true?**

- (A) It requires manual approval before every deployment
- (B) It automatically deploys every successful code change to production
- (C) It only applies to database changes
- (D) It is the same as Continuous Integration

**Q55. मल्टी-स्टेज डॉकर बिल्ड का उपयोग करने का मुख्य लाभ क्या है?**

- (A) कंटेनर का आकार बढ़ जाता है
- (B) कंटेनर का स्टार्टअप टाइम तेज़ हो जाता है
- (C) बिल्ड और रनटाइम चरणों को अलग करके छोटी अंतिम इमेज बनाई जाती है
- (D) एक ही इमेज में कई कंटेनर चलाने की सुविधा मिलती है

**What is the main advantage of using multi-stage Docker builds?**

- (A) Increased container size
- (B) Faster container startup time
- (C) Smaller final image by separating build and runtime stages
- (D) Allows running multiple containers in a single image

**Q56. कुबरनेट्स में, हेल्म चार्ट मुख्य रूप से क्या प्रबंधित करता है?**

- (A) कंटेनर नेटवर्किंग
- (B) लॉगिंग और मॉनिटरिंग
- (C) नोड प्रोविजनिंग
- (D) टेम्प्लेटेड संसाधन परिभाषाओं के माध्यम से एप्लिकेशन लाइफसाइकल

**In Kubernetes, what does a Helm chart primarily manage?**

- (A) Container networking
- (B) Logging and monitoring
- (C) Node provisioning
- (D) Application lifecycle through templated resource definitions

Q57. ईएलके स्टैक का कौन सा कंपोनेंट डेटा विज़ुअलाइज़ेशन के लिए ज़िम्मेदार है?

- (A) इलास्टिक्स सर्च (B) लॉगस्टैश  
(C) किबाना (D) फ्लुएंटी

**Which component of the ELK stack is responsible for data visualization?**

- (A) Elasticsearch (B) Logstash  
(C) Kibana (D) Fluentd

Q58. निगरानी में प्रोमैथियस की प्राथमिक भूमिका क्या है?

- (A) क्लेरी क्षमताओं के साथ डेटा स्टोरेज और कलेक्शन  
(B) विज़ुअलाइज़ेशन डैशबोर्ड  
(C) लॉग एग्रीगेशन  
(D) कंटेनर ऑर्किस्ट्रेशन

**What is the primary role of Prometheus in monitoring?**

- (A) Data collection and storage with querying capabilities  
(B) Visualization dashboards  
(C) Log aggregation  
(D) Container orchestration

Q59. कौन सा टेस्ट टाइप यह सत्यापित करता है कि व्यक्तिगत फ़ंक्शन या विधियाँ अलग-अलग सही ढंग से काम करती हैं?

- (A) एंड-टू-एंड टेस्ट (B) इंटीग्रेशन टेस्ट  
(C) यूनिट टेस्ट (D) सिस्टम टेस्ट

**Which test type verifies that individual functions or methods work correctly in isolation?**

- (A) End-to-End tests (B) Integration tests  
(C) Unit tests (D) System tests

Q60. स्वचालित टेस्टिंग में कोड कवरेज क्या मापता है?

- (A) टेस्ट रन के दौरान निष्पादित कोड का प्रतिशत  
(B) पास हुए टेस्ट मामलों की संख्या  
(C) टेस्ट का निष्पादन समय  
(D) डिटेक्ट किए गए बगों की संख्या

**What does code coverage measure in automated testing?**

- (A) Percentage of code executed while tests run  
(B) Number of test cases passed  
(C) Execution time of tests  
(D) Number of bugs detected

- Q61. निम्नलिखित में से कौन वाटरफॉल मॉडल की प्रमुख विशेषता है?**
- |                        |                             |
|------------------------|-----------------------------|
| (A) आइटेरेटिव साइकल    | (B) स्टीक्ट सीक्वेंशियल फेज |
| (C) डेली स्क्रम मीटिंग | (D) निरंतर डिलिवरी          |

**Which of the following is a key characteristic of the Waterfall model?**

- |                          |                              |
|--------------------------|------------------------------|
| (A) Iterative cycles     | (B) Strict sequential phases |
| (C) Daily Scrum meetings | (D) Continuous delivery      |

- Q62. लीन सॉफ्टवेयर डेवलपमेंट का मुख्य फोकस क्या है?**
- |   |
|---|
| (A) प्रदान की जाने वाली सुविधाओं की संख्या को अधिकतम करना |
| (B) अग्रिम योजना  |
| (C) दैनिक स्टैंडअप  |
| (D) गुणवत्ता से समझौता किए बिना अपव्यय को कम करना         |

**What is the main focus of Lean Software Development?**

- |  |
|--|
| (A) Maximizing the number of features delivered  |
| (B) Upfront Planning                             |
| (C) Daily Standups                               |
| (D) Minimizing waste without sacrificing quality |

- Q63. स्क्रम में स्प्रींट रेट्रोस्पेक्टिव का प्राथमिक लक्ष्य क्या है?**
- |  |
|--|
| (A) अगले स्प्रींट के कार्यों की योजना बनाना            |
| (B) हितधारकों को कार्य का प्रदर्शन करना                |
| (C) टीम की प्रक्रिया में सुधार के लिए उस पर विचार करना |
| (D) टीम के सदस्यों को भूमिकाएँ सौंपना                  |

**What is the primary goal of the Sprint Retrospective in Scrum?**

- |  |
|--|
| (A) Planning the next sprint's tasks               |
| (B) Demonstrating work to stakeholders             |
| (C) Reflecting on the team's process to improve it |
| (D) Assigning roles to team members                |

- Q64. विकास टीम के कार्य से उत्पन्न उत्पाद के मूल्य को अधिकतम करने के लिए कौन सी स्क्रम भूमिका जिम्मेदार है?**

- |                   |                      |
|-------------------|----------------------|
| (A) स्क्रम मास्टर | (B) प्रॉडक्ट ओनर     |
| (C) स्टेकहोल्डर   | (D) परियोजना प्रबंधक |

**Which Scrum role is responsible for maximizing the value of the product resulting from the work of the Development Team?**

- |                  |                     |
|------------------|---------------------|
| (A) Scrum Master | (B) Product Owner   |
| (C) Stakeholder  | (D) Project Manager |

- Q65. सूचीबद्ध अन्य सुविधाओं की तुलना में ट्रेलो में कौन सी विशेषता विशिष्ट है?**
- (A) कार्ड-आधारित कानबन बोर्ड (B) टाइमलाइन (Gantt) व्यू  
(C) बिल्ट इन गिट इंटीग्रेशन (D) बर्नडाउन चार्ट

**Which feature is unique to Trello compared to the others listed?**

- (A) Card-based Kanban board (B) Timeline (Gantt) view  
(C) Built-in Git integration (D) Burndown charts

- Q66. समस्याओं और माइलस्टोन को ट्रैक करने के लिए जीआईटी रिपॉजिटरी के साथ आमतौर पर किस टूल का उपयोग किया जाता है?**

- (A) जिरा (B) एसाना  
(C) ट्रेलो (D) गिटहब इसूज

**Which tool is commonly used with Git repositories for tracking issues and milestones?**

- (A) Jira (B) Asana  
(C) Trello (D) Github Issues

- Q67. परियोजना प्रबंधन में रिस्क लॉग क्या है?**

- (A) डेवलपर्स को कार्य सौंपने का एक उपकरण  
(B) परियोजना जोखिमों और उनकी स्थिति को सूचीबद्ध करने वाला एक डॉक्युमेंट  
(C) पूर्ण सुविधाओं की एक रिपॉजिटरी  
(D) हितधारकों के लिए एक संचार योजना

**What is a risk log in project management?**

- (A) A tool to assign tasks to developers  
(B) A document listing project risks and their status  
(C) A repository for completed features  
(D) A communication plan for stakeholders

- Q68. जोखिमों के लिए "शमन रणनीति" का सबसे अच्छा वर्णन क्या है?**

- (A) जोखिम होने पर रिकॉर्डिंग करना  
(B) जोखिम भरे कार्य के लिए परियोजना प्राथमिकता बढ़ाना  
(C) जोखिम की संभावना या प्रभाव को कम करने के लिए योजनाबद्ध कार्रवाई करना  
(D) कम संभावना वाले जोखिमों की अनदेखी करना

**What is the best description of a "mitigation strategy" for risks?**

- (A) Recording when a risk occurs  
(B) Increasing project priority for a risky task  
(C) Taking planned action to reduce probability or impact of a risk  
(D) Ignoring low-probability risks

**Q69. सॉफ्टवेयर परियोजनाओं में नियमित हितधारक रिपोर्टिंग क्यों महत्वपूर्ण है?**

- (A) यह परियोजना की प्रगति को धीमा कर देता है
- (B) यह सुनिश्चित करता है कि हितधारक सूचित और शामिल रहें
- (C) यह दस्तावेज़ीकरण को अनावश्यक बनाता है
- (D) यह केवल परियोजना की शुरुआत में ही उपयोगी होता है

**Why is regular stakeholder reporting important in software projects?**

- (A) It slows down project progress
- (B) It ensures stakeholders remain informed and involved
- (C) It makes documentation unnecessary
- (D) It is only useful at the project start

**Q70. निम्नलिखित में से कौन सा आमतौर पर परियोजना संचार योजना का हिस्सा नहीं होता है?**

- (A) सोर्स कोड रिपोजिटरी
- (B) हितधारकों की सूची
- (C) अपडेट की फ्रिक्वेंसी
- (D) पसंदीदा संचार चैनल

**Which of the following is NOT typically part of a project communication plan?**

- (A) Source code repository
- (B) List of stakeholders
- (C) Frequency of updates
- (D) Preferred communication channels

**Q71. सॉफ्टवेयर परीक्षण में संक्षिप्त शब्द टीडीडी का पूर्ण रूप क्या है?**

- (A) टेस्ट-ड्रिवेन डिबगिंग
- (B) टूल-ड्रिवेन डेवलपमेंट
- (C) टेस्ट-ड्रिवेन डेवलपमेंट
- (D) टेम्पलेट-डिज़ाइन डॉक्युमेंटेशन

**What does the acronym TDD stand for in software testing?**

- (A) Test-Driven Debugging
- (B) Tool-Driven Development
- (C) Test-Driven Development
- (D) Template-Design Documentation

**Q72. यूनिट टेस्टिंग में मॉकीटो की भूमिका का सबसे अच्छा वर्णन कौन सा कथन करता है?**

- (A) यह एक डेटाबेस प्रबंधन टूल है
- (B) यह परीक्षणों में निर्भरताओं का अनुकरण करने के लिए एक मॉकिंग फ्रेमवर्क है
- (C) यह एक प्रदर्शन परीक्षण उपकरण है
- (D) यह परीक्षण कवरेज रिपोर्ट तैयार करता है

**Which statement best describes the role of Mockito in unit testing?**

- (A) It is a database management tool
- (B) It is a mocking framework for simulating dependencies in tests
- (C) It is a performance testing tool
- (D) It generates test coverage reports

- Q73. माइक्रोसर्विस आर्किटेक्चर के लिए इंटीग्रेशन टेस्टिंग क्यों महत्वपूर्ण है?**
- (A) यह अलग-अलग विधियों को अलग-अलग मान्य करता है
  - (B) यह पुष्टि करता है कि मॉड्यूल नेटवर्क सीमाओं के पार सही ढंग से इंटरैक्ट करते हैं
  - (C) यह केवल फ्रंट-एंड कार्यक्षमता को टेस्ट करता है
  - (D) यह सिस्टम प्रदर्शन का बेंचमार्क करता है

**Why is integration testing important for microservices architectures?**

- (A) It validates individual methods in isolation
- (B) It confirms modules interact correctly across network boundaries
- (C) It only tests front-end functionality
- (D) It benchmarks system performance

- Q74. एपीआई इंटीग्रेशन टेस्टिंग को ऑटोमेट करने के लिए आमतौर पर किस टूल का उपयोग किया जाता है?**
- (A) सेलेनियम
  - (B) पोस्टमैन
  - (C) जेमीटर
  - (D) जेनकिंस

**Which tool is commonly used to automate API integration testing?**

- (A) Selenium
- (B) Postman
- (C) JMeter
- (D) Jenkins

- Q75. ब्राउज़र ऑटोमेशन और यूआई टेस्टिंग के लिए कौन सा टेस्टिंग टूल सबसे प्रसिद्ध है?**
- (A) सेलेनियम
  - (B) टेस्टएनजी
  - (C) जेमीटर
  - (D) सोनारक्यूब

**Which testing tool is best known for browser automation and UI testing?**

- (A) Selenium
- (B) TestNG
- (C) JMeter
- (D) SonarQube

- Q76. कौन सा परीक्षण स्वचालन उपकरण जावास्क्रिप्ट का उपयोग करता है और आधुनिक वेब ऐप फ्रेमवर्क और समानांतर निष्पादन के लिए मूल समर्थन प्रदान करता है?**
- (A) जेयूनिट
  - (B) टेस्टएनजी
  - (C) साइप्रेस
  - (D) जेमीटर

**Which test automation tool uses JavaScript and offers native support for modern web app frameworks and parallel execution?**

- (A) JUnit
- (B) TestNG
- (C) Cypress
- (D) JMeter

- Q77. जेमीटर वेब एप्लिकेशन में मुख्य रूप से किसका विश्लेषण करता है?**
- (A) सोर्स कोड की गुणवत्ता
  - (B) नेटवर्क प्रदर्शन
  - (C) सिस्टम लोड और रिस्पॉंस टाइम
  - (D) यूआई एक्सेसिबिलिटी

**What does JMeter primarily analyze in a web application?**

- (A) Source code quality
- (B) Network performance
- (C) System load and response times
- (D) UI Accessibility

**Q78.** सेक्योरिटी टेस्टिंग में ओडब्ल्यूएसपी टेस्टिंग गाइड किस सिद्धांत पर ज़ोर देती है?

- (A) सिस्टम की कमज़ोरियों की पहचान की जानी चाहिए और ज्ञात थ्रेट मॉडल के सापेक्ष उनका सत्यापन किया जाना चाहिए
- (B) प्रत्येक फीचर में प्रदर्शन मीट्रिक होना चाहिए
- (C) यूआई टेस्टिंग पहले आनी चाहिए
- (D) केवल मैनुअल टेस्ट ही विश्वसनीय होते हैं

**Which principle is emphasized by the OWASP Testing Guide in security testing?**

- (A) System vulnerabilities should be identified and validated against known threat models
- (B) Every feature must have performance metrics
- (C) UI testing should come first
- (D) Only manual tests are reliable

**Q79.** कौन सा स्टेटमेंट सीआई/सीडी पाइपलाइनों में निरंतर टेस्टिंग को सबसे अच्छी तरह परिभाषित करता है?

- (A) परीक्षण केवल परिनियोजन के बाद किया जाता है
- (B) सॉफ्टवेयर वितरण के प्रत्येक चरण में स्वचालित टेस्ट चलाए जाते हैं
- (C) टेस्ट केवल डेवलपर मशीनों पर चलाए जाते हैं
- (D) केवल प्रदर्शन टेस्ट शामिल हैं

**Which statement best defines continuous testing in CI/CD pipelines?**

- (A) Testing is performed only after deployment
- (B) Automated tests run at every stage of software delivery
- (C) Tests are run solely on developer machines
- (D) Only performance tests are included

**Q80.** सोनारक्यूब क्यूए प्रक्रिया में क्या प्रदान करता है?

- (A) रीयल-टाइम यूआई टेस्ट निष्पादन
- (B) स्वचालित कोड कवरेज, कोड गुणवत्ता और स्टैटिक एनालिसिस
- (C) प्रत्यक्ष एपीआई संचार परीक्षण
- (D) प्रदर्शन बेंचमार्किंग रिपोर्ट

**What does SonarQube provide in the QA process?**

- (A) Real-time UI test execution
- (B) Automated code coverage, code quality, and static analysis
- (C) Direct API communication tests
- (D) Performance benchmarking reports

**Q81.** किस प्रकार के हमले में अन्य उपयोगकर्ताओं द्वारा देखे गए वेब पेज में दुर्भावनापूर्ण स्क्रिप्ट इंजेक्ट करना शामिल है?

- (A) एसक्युएल इंजेक्शन (B) क्रॉस-साइट स्क्रिप्टिंग (एक्सएसएस)  
(C) सीएसआरएफ (D) डायरेक्टरी ट्रैवर्सल

**Which type of attack involves injecting malicious scripts into a web page viewed by other users?**

- (A) SQL Injection (B) Cross-Site Scripting (XSS)  
(C) CSRF (D) Directory Traversal

**Q82.** किसी एप्लिकेशन में एसक्युएल इंजेक्शन को रोकने का सबसे प्रभावी तरीका क्या है?

- (A) संदिग्ध कीवर्ड को ब्लैकलिस्ट करना  
(B) डेटाबेस को एन्क्रिप्ट करना  
(C) डेटाबेस एक्सेस को डिसेबल करना  
(D) पैरामीटराइज़्ड क्वेरीज़ या तैयार स्टेटमेंट्स का उपयोग करना

**What is the most effective way to prevent SQL Injection in an application?**

- (A) Blacklisting suspicious keywords  
(B) Encrypting the database  
(C) Disabling database access  
(D) Using parameterized queries or prepared statements

**Q83.** किस मेथड में उपयोगकर्ता को एक्सेस प्रदान करने से पहले दो प्रकार की पहचान प्रदान करने की आवश्यकता होती है?

- (A) एकल साइन-ऑन (एसएसओ) (B) OAuth2  
(C) बेसिक औथेंटिकेशन (D) मल्टी-फैक्टर औथेंटिकेशन (एमएफए)

**Which method requires a user to provide two forms of identification before granting access?**

- (A) Single Sign-On (SSO) (B) OAuth2  
(C) Basic Authentication (D) Multi-Factor Authentication (MFA)

**Q84.** OAuth2 के संबंध में कौन सा कथन सत्य है?

- (A) OAuth2 का उपयोग केवल यूजर लॉगिन के लिए किया जाता है  
(B) OAuth2 थर्ड-पार्टी ऐप्स को क्रेडेंशियल साझा किए बिना यूजर डेटा तक पहुँचने की अनुमति देता है  
(C) OAuth2 केवल स्टैटिक एपीआई कुंजियाँ उत्पन्न करता है  
(D) OAuth2 के लिए बायोमेट्रिक प्रमाणीकरण आवश्यक है

**Which statement is true regarding OAuth2?**

- (A) OAuth2 is used only for user login  
(B) OAuth2 allows third-party apps to access user data without sharing credentials  
(C) OAuth2 generates static API keys only  
(D) OAuth2 requires biometric authentication

Q85. **सीमेट्रिक और एसीमेट्रिक एन्क्रिप्शन में क्या अंतर है?**

- (A) सीमेट्रिक एन्क्रिप्शन और डिक्लिप्शन के लिए एक ही कुंजी का उपयोग करता है; एसीमेट्रिक कुंजियों के एक युग्म का उपयोग करता है
- (B) एसीमेट्रिक केवल डेटाबेस सुरक्षा के लिए है
- (C) सीमेट्रिक केवल वेबसाइटों पर काम करता है
- (D) कोई अंतर नहीं है

**What distinguishes symmetric from asymmetric encryption?**

- (A) Symmetric uses the same key for encryption and decryption; asymmetric uses a pair of keys
- (B) Asymmetric is only for database protection
- (C) Symmetric works only on websites
- (D) There's no difference

Q86. **एसएचए-256 और एमडी5 किस प्रकार के सुरक्षा तंत्र के उदाहरण हैं?**

- (A) ब्लॉक सिफर
- (B) हैशिंग एल्गोरिदम
- (C) एसीमेट्रिक एन्क्रिप्शन
- (D) फ़ायरवॉल नियम

**SHA-256 and MD5 are examples of what kind of security mechanism?**

- (A) Block cipher
- (B) Hashing algorithms
- (C) Symmetric encryption
- (D) Firewall rules

Q87. **संयुक्त राज्य अमेरिका में स्वास्थ्य सूचना गोपनीयता और सुरक्षा को मुख्य रूप से कौन सा अनुपालन मानक संबोधित करता है?**

- (A) जीडीपीआर
- (B) एचआईपीए
- (C) पीसीआई-डीएसएस
- (D) आईएसओ/आईईसी 27001

**Which compliance standard primarily addresses health information privacy and security in the United States?**

- (A) GDPR
- (B) HIPAA
- (C) PCI-DSS
- (D) ISO/IEC 27001

Q88. **सुरक्षा में "थ्रेट मॉडलिंग" का सबसे अच्छा वर्णन कौन सी गतिविधि करती है?**

- (A) पुराने पैकेजों के लिए ओपन-सोर्स लाइब्रेरीज़ को स्कैन करना
- (B) किसी एप्लिकेशन के संभावित खतरों की पहचान, मूल्यांकन और समाधान के लिए आरेख डिज़ाइन करना
- (C) घटना प्रतिक्रिया योजनाओं का परीक्षण करना
- (D) फ़ायरवॉल नीतियाँ राइट करना

**Which activity best describes "threat modeling" in security?**

- (A) Scanning open-source libraries for outdated packages
- (B) Designing diagrams to identify, evaluate, and address potential threats to an application
- (C) Testing incident response plans
- (D) Writing firewall policies

Q89. "रेड टीम/ब्लू टीम" अभ्यासों के बारे में निम्नलिखित में से कौन सा कथन सही है?

- (A) ब्लू टीम केवल आक्रमण करती है, रेड टीम केवल अनुपालन का परीक्षण करती है
- (B) दोनों टीमों डॉक्यूमेंटेशन लिखती हैं
- (C) दोनों टीमों कोड विकसित करती हैं
- (D) रेड टीम आक्रमणकारियों का अनुकरण करती है, ब्लू टीम बचाव करती है

Which of the following statements is true about "Red Team/Blue Team" exercises?

- (A) Blue Team only attacks, Red Team only tests for compliance
- (B) Both Teams write documentation
- (C) Both Teams develop code
- (D) Red Team simulates attackers, Blue Team defends

Q90. एसडीएलसी में सुरक्षा को इंटीग्रेट करने का मुख्य लक्ष्य क्या है?

- (A) सुरक्षा का परीक्षण केवल रिलीज़ चरण में किया जाता है
- (B) सुरक्षा को एक बाद की बात मानना
- (C) यह सुनिश्चित करना कि विकास के शुरुआती चरण में ही कमज़ोरियों को रोका और पहचाना जा सके
- (D) सुरक्षा ज़िम्मेदारी को एक ही डेवलपर तक सीमित रखना

What is a core goal of integrating security into the SDLC?

- (A) Security is tested only at the release phase
- (B) To treat security as an afterthought
- (C) To ensure vulnerabilities are prevented and detected early in development
- (D) To limit security responsibility to a single developer

Q91. माइक्रोसर्विसेज़ के लिए डोमेन-ड्रिवेन डिज़ाइन (डीडीडी) में सेवा सीमाओं का सबसे अच्छा वर्णन कौन सा सिद्धांत करता है?

- (A) बाउंडेड कॉन्टेक्ट
- (B) टू-फ़ेज़ कमिट
- (C) डेटा शार्डिंग
- (D) सिंगलटन पैटर्न

Which principle best describes service boundaries in Domain-Driven Design (DDD) for microservices?

- (A) Bounded Context
- (B) Two-Phase Commit
- (C) Data Sharding
- (D) Singleton Pattern

Q92. माइक्रोसर्विस आर्किटेक्चर में सर्विस डिस्कवरी के लिए कॉन्सल या यूरेका का उपयोग करने का मुख्य लाभ क्या है?

- (A) बिल्ड ऑटोमेशन प्रदान करना
- (B) सर्विस इंस्टेंस का गतिशील पंजीकरण और डिस्कवरी सक्षम करना
- (C) डेटाबेस कनेक्शन एन्क्रिप्ट करना
- (D) लोड टेस्टिंग करना

What is the main benefit of using Consul or Eureka for service discovery in a microservices architecture?

- (A) Providing build automation
- (B) Enabling dynamic registration and discovery of service instances
- (C) Encrypting database connections
- (D) Performing load testing

**Q93.** निम्नलिखित में से कौन सा एपीआई गेटवे स्तर पर सर्किट ब्रेकर पैटर्न को सबसे अच्छी तरह समझाता है?

- (A) ओवरलोड को रोकने के लिए विफल डाउनस्ट्रीम सेवा के रिक्वेस्ट को अस्थायी रूप से ब्लॉक करता है
- (B) सभी सेवाओं के लॉग एग्रीगेट करता है
- (C) एपीआई ट्रैफ़िक को एन्क्रिप्ट करता है
- (D) औथेंटिकेशन मिडलवेयर प्रदान करता है

**Which of the following best explains the Circuit Breaker pattern at the API gateway level?**

- (A) Temporarily blocks requests to a failing downstream service to prevent overload
- (B) Aggregates logs from all services
- (C) Encrypts API traffic
- (D) Provides authentication middleware

**Q94.** एसिंक्रोनस मैसेजिंग के लिए माइक्रोसर्विसेज़ में आमतौर पर किस संचार शैली का उपयोग किया जाता है?

- (A) काफ़का या रैबिटएमक्यू जैसे मैसेज ब्रोकर
- (B) REST एपीआई
- (C) gRPC
- (D) एसओएपी

**Which communication style is typically used in microservices for asynchronous messaging?**

- (A) Message Brokers like Kafka or RabbitMQ
- (B) REST APIs
- (C) gRPC
- (D) SOAP

**Q95.** माइक्रोसर्विसेज़ में बल्कहेड आइसोलेशन लागू करने का प्राथमिक लक्ष्य क्या है?

- (A) डेटाबेस का विस्तार करना
- (B) यूआई प्रदर्शन में सुधार करना
- (C) डायरेक्ट फ़ाइल शेयरिंग सक्षम करना
- (D) विफलताओं को अलग करना ताकि वे पूरे सिस्टम को प्रभावित न करें

**What is the primary goal of applying Bulkhead isolation in microservices?**

- (A) Scaling up the database
- (B) Improving UI performance
- (C) Enabling direct file sharing
- (D) Isolating failures to prevent them from affecting the entire system

**Q96. माइक्रोसर्विसेज़ में सागा पैटर्न का उपयोग मुख्यतः \_\_\_\_\_ के लिए किया जाता है।**

- (A) ग्लोबल लॉक के साथ वितरित लेनदेन चलाना
- (B) क्षतिपूर्ति क्रियाओं के साथ स्थानीय लेनदेन के अनुक्रम के माध्यम से लंबे समय तक चलने वाले लेनदेन का प्रबंधन करना
- (C) सख्ती से एसीआईडी-अनुपालक बहु-डेटाबेस लेनदेन सुनिश्चित करना
- (D) डेटा को विभाजित करके क्षैतिज रूप से स्केल करना

**The Saga pattern in microservices is primarily used to:**

- (A) Run distributed transactions with global locks
- (B) Manage long-running transactions via a sequence of local transactions with compensating actions
- (C) Ensure strictly ACID-compliant multi-database transactions
- (D) Scale horizontally by sharding data

**Q97. इक्रोसर्विसेज़ में टू-फ़ेज़ कमिट प्रोटोकॉल का उपयोग कम ही क्यों किया जाता है?**

- (A) यह सुरक्षित नहीं है
- (B) यह वेब एप्लिकेशन के लिए बहुत सरल है
- (C) यह टाइट कपलिंग और ब्लॉक्स का निर्माण करता है, जिससे स्केलेबिलिटी और विश्वसनीयता को नुकसान पहुँचता है
- (D) यह नॉन-रिलेशनल डेटाबेस को सपोर्ट नहीं करता

**Why is the Two-Phase Commit protocol rarely used in microservices?**

- (A) It's not secure
- (B) It's too simple for web applications
- (C) It introduces tight coupling and blocks, harming scalability and reliability
- (D) It does not support non-relational databases

**Q98. माइक्रोसर्विसेज़ में सुरक्षित और स्टेटलेस प्रमाणीकरण सुनिश्चित करने के लिए आमतौर पर किस विधि का उपयोग किया जाता है?**

- (A) पासिंग सेशन कुकीज़ ओन्ली
- (B) सभी सेवाओं को सार्वजनिक बनाना
- (C) कोड में एपीआई कुंजियों को हार्डकोड करना
- (D) जेएसओएन वेब टोकन (जेडब्ल्यूटी) का उपयोग करना

**Which method is commonly used to ensure secure and stateless authentication across microservices?**

- (A) Passing session cookies only
- (B) Making all services public
- (C) Hardcoding API keys in code
- (D) Using JSON Web Tokens (JWT)

**Q99. माइक्रोसर्विसेज़ में डिस्ट्रीब्यूटेड ट्रेसिंग क्यों महत्वपूर्ण है?**

- (A) यह डेटाबेस के प्रदर्शन को बेहतर बनाता है
- (B) यह अनुरोधों को ट्रैक करता है क्योंकि वे कई सेवाओं में प्रसारित होते हैं, जिससे अड़चनों का निदान करने में मदद मिलती है
- (C) यह एक प्रकार का यूनिट परीक्षण है
- (D) यह केवल मोनोलिथ के लिए आवश्यक है

**Why is distributed tracing critical in microservices?**

- (A) It improves database performance
- (B) It tracks requests as they propagate across multiple services, helping diagnose bottlenecks
- (C) It is a type of unit test
- (D) It is only needed for monoliths

**Q100. क्लाउड-नेटिव माइक्रोसर्विसेज़ में मेट्रिक्स संग्रह और निगरानी के लिए किस टूल का व्यापक रूप से उपयोग किया जाता है?**

- (A) जिपकिन
- (B) हेल्म
- (C) युरेका
- (D) प्रोमैथियस

**Which tool is widely used for metrics collection and monitoring in cloud-native microservices?**

- (A) Zipkin
- (B) Helm
- (C) Eureka
- (D) Prometheus

