1. What is the purpose of static analysis in software development?

a) To execute the code and identify runtime errors
b) To analyze the code without executing it to find potential issues
c) To test the code’s functionality through user interactions
d) To debug the code during runtime

Answer: b) To analyze the code without executing it to find potential issues

Explanation: Static analysis involves examining the code without executing it to identify potential defects, security vulnerabilities, or adherence to coding standards.

2. Which technique involves a manual examination of code to identify defects and improve quality?

a) Black-box testing
b) White-box testing
c) Code inspections
d) Integration testing

Answer: c) Code inspections

Explanation: Code inspections involve a manual examination of code by peers to identify defects, improve code quality, and ensure adherence to coding standards.

3. Which testing level focuses on verifying the interactions between integrated components or modules?

a) Unit testing
b) System testing
c) Integration testing
d) Acceptance testing

Answer: c) Integration testing

Explanation: Integration testing verifies the interactions and interfaces between integrated components or modules to ensure they function together correctly.

4. What is the purpose of a test oracle?

a) To design test cases
b) To execute test cases
c) To evaluate test results
d) To specify the expected outcome of test cases

Answer: d) To specify the expected outcome of test cases

Explanation: A test oracle specifies the expected outcomes of test cases, helping testers determine whether the system under test behaves correctly.

5. Which test technique examines the functionality of software without considering its internal structure?

a) Black-box testing
b) White-box testing
c) Gray-box testing

d) Regression testing

Answer: a) Black-box testing

Explanation: Black-box testing focuses on examining the functionality of software without considering its internal structure, focusing solely on inputs and outputs.

6. What type of testing evaluates individual units or components of a software application?

a) Integration testing
b) System testing
c) Acceptance testing
d) Unit testing

Answer: d) Unit testing

Explanation: Unit testing evaluates individual units or components of a software application in isolation to ensure they function correctly.

7. Which testing framework is commonly used for unit testing in Java applications?

a) JUnit
b) PyTest
c) NUnit
d) Jasmine

Answer: a) JUnit

Explanation: JUnit is a widely used testing framework for unit testing in Java applications.

8. What is the primary focus of system testing?

a) Testing individual components
b) Testing the entire system as a whole
c) Verifying integration between modules
d) Ensuring code quality

Answer: b) Testing the entire system as a whole

Explanation: System testing focuses on testing the entire system as a whole to ensure that it meets specified requirements and functions correctly in its intended environment.

9. Which specialized testing technique focuses on assessing how a system performs under stress or high load?

a) Usability testing
b) Performance testing
c) Security testing
d) Compatibility testing

Answer: b) Performance testing

Explanation: Performance testing evaluates how a system performs under specific conditions, such as high load or stress, to ensure it meets performance requirements.

10. What is a test plan in software testing?

a) A document specifying the test approach, objectives, and activities
b) A tool used for executing test cases
c) A framework for automated testing
d) A set of metrics used to evaluate test effectiveness

Answer: a) A document specifying the test approach, objectives, and activities

Explanation: A test plan is a document that outlines the test approach, objectives, scope, resources, and schedule for a testing project.

11. Which metric measures the number of defects found per unit of time during testing?

a) Defect density
b) Defect removal efficiency
c) Defect arrival rate
d) Test coverage

Answer: c) Defect arrival rate

Explanation: Defect arrival rate measures the number of defects found per unit of time during testing, providing insights into defect detection efficiency.

12. Which testing tool is commonly used for automated functional testing of web applications?

a) JIRA
b) Selenium
c) Jenkins

d) Apache JMeter

Answer: b) Selenium

Explanation: Selenium is a popular testing tool used for automated functional testing of web applications.

13. In object-oriented analysis and design (OOAD), what is a class?

a) A specific instance of an object
b) A template for creating objects
c) A method for performing a specific action
d) A collection of attributes

Answer: b) A template for creating objects

Explanation: In OOAD, a class serves as a template or blueprint for creating objects with similar properties and behaviors.

14. Which phase of the software development life cycle (SDLC) involves transforming requirements into an object-oriented model?

a) Analysis
b) Design
c) Implementation
d) Testing

Answer: b) Design

Explanation: The design phase of the SDLC involves transforming requirements gathered during the analysis phase into an object-oriented model.

15. How does object-oriented analysis differ from structured software engineering?

a) Object-oriented analysis focuses on data structures, while structured software engineering focuses on control structures.
b) Object-oriented analysis emphasizes encapsulation, while structured software engineering emphasizes modularity.
c) Object-oriented analysis models systems using objects and their interactions, while structured software engineering uses hierarchical structures.
d) Object-oriented analysis is more suitable for large-scale systems, while structured software engineering is preferred for small-scale projects.

Answer: c) Object-oriented analysis models systems using objects and their interactions, while structured software engineering uses hierarchical structures.

Explanation: Object-oriented analysis focuses on modeling systems using objects and their interactions, whereas structured software engineering typically employs hierarchical structures and procedural approaches.

16. What is a unit in unit testing?

a) A single line of code
b) The smallest testable part of an application
c) A test case
d) A function or method

Answer: b) The smallest testable part of an application

Explanation: In unit testing, a unit refers to the smallest testable part of an application, such as a function, method, or class.

17. Which type of testing involves executing the software application with the intent of finding defects?

a) Validation testing
b) Verification testing
c) White-box testing
d) Black-box testing

Answer: d) Black-box testing

Explanation: Black-box testing involves executing the software application with the intent of finding defects by examining its functionality without considering its internal structure.

18. What is the purpose of a testing framework?

a) To execute test cases manually
b) To automate the testing process
c) To analyze test results
d) To document test cases

Answer: b) To automate the testing process

Explanation: A testing framework provides tools and utilities to automate the testing process, including test case execution, reporting, and management.

19. Which testing technique is also known as glass-box testing?

a) White-box testing
b) Black-box testing
c) Gray-box testing
d) Integration testing

Answer: a) White-box testing

Explanation: White-box testing, also known as glass-box testing, involves examining the internal structure and logic of the software application.

20. Which type of testing focuses on verifying the behavior of the entire system?

a) Unit testing
b) Integration testing
c) System testing
d) Acceptance testing

Answer: c) System testing

Explanation: System testing focuses on verifying the behavior of the entire system to ensure it meets specified requirements and functions correctly in its intended environment.

21. What is the primary objective of acceptance testing?

a) To identify defects in the software
b) To verify the system meets specified requirements
c) To validate the software against user expectations
d) To ensure code quality

Answer: c) To validate the software against user expectations

Explanation: Acceptance testing validates the software against user expectations and verifies that it meets specified requirements before it is accepted for deployment.

22. Which testing level evaluates the interactions between integrated components or modules?

a) Unit testing
b) System testing
c) Integration testing
d) Acceptance testing

Answer: c) Integration testing

Explanation: Integration testing evaluates the interactions and interfaces between integrated components or modules to ensure they function together correctly.

23. Which testing technique relies on input values and their corresponding output values?

a) Boundary value analysis
b) Equivalence partitioning
c) Decision table testing

d) Cause-effect graphing

Answer: c) Decision table testing

Explanation: Decision table testing relies on input values and their corresponding output values to derive test cases and verify system behavior.

24. What is the purpose of a test case?

a) To design the software application
b) To execute the software application
c) To verify the behavior of a specific feature or functionality
d) To analyze code coverage

Answer: c) To verify the behavior of a specific feature or functionality

Explanation: A test case is a set of conditions or actions used to verify the behavior of a specific feature or functionality of a software application.

25. Which metric measures the percentage of code exercised by tests?

a) Test coverage
b) Defect density
c) Defect removal efficiency
d) Test effectiveness

Answer: a) Test coverage

Explanation: Test coverage measures the percentage of code exercised by tests, indicating how much of the code has been executed during testing.

26. Which type of testing examines how well a system performs under normal conditions?

a) Stress testing
b) Usability testing
c) Regression testing
d) Functional testing

Answer: d) Functional testing

Explanation: Functional testing examines how well a system performs under normal conditions to ensure it meets specified functional requirements.

27. What is the purpose of a testing tool?

a) To design software applications
b) To execute test cases manually
c) To automate the testing process
d) To analyze code quality

Answer: c) To automate the testing process

Explanation: Testing tools are used to automate various aspects of the testing process, including test case execution, reporting, and management, to improve efficiency and effectiveness.

28. Which phase of the software development life cycle (SDLC) involves gathering and analyzing user requirements?

a) Analysis
b) Design
c) Implementation
d) Testing

Answer: a) Analysis

Explanation: The analysis phase of the SDLC involves gathering and analyzing user requirements to understand the needs and constraints of the software project.

29. Which testing technique focuses on evaluating the user interface of a software application?

a) Usability testing
b) Security testing
c) Performance testing
d) Compatibility testing

Answer: a) Usability testing

Explanation: Usability testing focuses on evaluating the user interface and overall user experience of a software application to ensure it is intuitive and easy to use.

30. What is the purpose of a test plan in software testing?

a) To design test cases

b) To execute test cases

c) To document the test approach and objectives

d) To analyze test results

Answer: c) To document the test approach and objectives

Explanation: A test plan is a document that outlines the test approach, objectives, scope, resources, and schedule for a testing project, providing guidance for the testing process.

Related Posts:

1. The Software Product and Software Process MCQ
2. Software Design MCQ
3. Software Maintenance & Software Project Measurement MCQ
4. Computer Architecture, Design, and Memory Technologies MCQ
5. Introduction to Energy Science MCQ
6. Ecosystems MCQ
7. Biodiversity and its conservation MCQ
8. Environmental Pollution mcq
9. Social Issues and the Environment MCQ
10. Field work mcq
11. Discrete Structure MCQ
12. Set Theory, Relation, and Function MCQ
13. Propositional Logic and Finite State Machines MCQ
14. Graph Theory and Combinatorics MCQ
15. Relational algebra,Functions and graph theory MCQ
16. Data Structure MCQ

17. Stacks MCQ
18. TREE MCQ
19. Graphs MCQ
20. Sorting MCQ
21. Digital Systems MCQ
22. Combinational Logic MCQ
23. Sequential logic MCQ
24. Analog/Digital Conversion, Logic Gates, Multivibrators, and IC 555 MCQ
25. Introduction to Digital Communication MCQ
26. Introduction to Object Oriented Thinking & Object Oriented Programming MCQ
27. Encapsulation and Data Abstraction MCQ
28. MCQ
29. Relationships – Inheritance MCQ
30. Polymorphism MCQ
31. Library Management System MCQ
32. Numerical Methods MCQ
33. Transform Calculus MCQ
34. Concept of Probability MCQ
35. Algorithms, Designing MCQ
36. Study of Greedy strategy MCQ
37. Concept of dynamic programming MCQ
38. Algorithmic Problem MCQ
39. Trees, Graphs, and NP-Completeness MCQ
40. Basic Structure of Computer MCQ
41. Computer Arithmetic MCQ
42. I/O Organization MCQ
43. Memory Organization MCQ

44. Multiprocessors MCQ
45. Introduction to Operating Systems MCQ
46. File Systems MCQ
47. CPU Scheduling MCQ
48. Memory Management MCQ
49. Input / Output MCQ
50. Operating Systems and Concurrency
51. Software Development and Architecture MCQ
52. Software architecture models MCQ
53. Software architecture implementation technologies MCQ
54. Software Architecture analysis and design MCQ
55. Software Architecture documentation MCQ
56. Introduction to Computational Intelligence MCQ
57. Fuzzy Systems MCQ
58. Genetic Algorithms MCQ
59. Rough Set Theory MCQ
60. Introduction to Swarm Intelligence, Swarm Intelligence Techniques MCQ
61. Neural Network History and Architectures MCQ
62. Autoencoder MCQ
63. Deep Learning MCQs
64. RL & Bandit Algorithms MCQs
65. RL Techniques MCQs
66. Review of traditional networks MCQ
67. Study of traditional routing and transport MCQ
68. Wireless LAN MCQ
69. Mobile transport layer MCQ
70. Big Data MCQ

71. Hadoop and Related Concepts MCQ
72. Hive, Pig, and ETL Processing MCQ
73. NoSQL MCQs Concepts, Variations, and MongoDB
74. Mining social Network Graphs MCQ
75. Mathematical Background for Cryptography MCQ
76. Cryptography MCQ
77. Cryptographic MCQs
78. Information Security MCQ
79. Cryptography and Information Security Tools MCQ
80. Data Warehousing MCQ
81. OLAP Systems MCQ
82. Introduction to Data& Data Mining MCQ
83. Supervised Learning MCQ
84. Clustering & Association Rule mining MCQ
85. Fundamentals of Agile Process MCQ
86. Agile Projects MCQs
87. Introduction to Scrum MCQs
88. Introduction to Extreme Programming (XP) MCQs
89. Agile Software Design and Development MCQs
90. Machine Learning Fundamentals MCQs
91. Neural Network MCQs
92. CNNs MCQ
93. Reinforcement Learning and Sequential Models MCQs
94. Machine Learning in ImageNet Competition mcq
95. Computer Network MCQ
96. Data Link Layer MCQ
97. MAC Sub layer MCQ

98. Network Layer MCQ
99. Transport Layer MCQ
100. Raster Scan Displays MCQs
101. 3-D Transformations MCQs
102. Visualization MCQ
103. Multimedia MCQs
104. Introduction to compiling & Lexical Analysis MCQs
105. Syntax Analysis &Syntax Directed Translation MCQs
106. Type Checking & Run Time Environment MCQs
107. Code Generation MCQs
108. Code Optimization MCQs
109. INTRODUCTION Knowledge Management MCQs
110. Organization and Knowledge Management MCQs
111. Telecommunications and Networks in Knowledge Management MCQs
112. Components of a Knowledge Strategy MCQs
113. Advanced topics and case studies in knowledge management MCQs
114. Conventional Software Management MCQs
115. Software Management Process MCQs
116. Software Management Disciplines MCQs
117. Rural Management MCQs
118. Human Resource Management for rural India MCQs
119. Management of Rural Financing MCQs
120. Research Methodology MCQs
121. Research Methodology MCQs
122. IoT MCQs
123. Sensors and Actuators MCQs
124. IoT MCQs: Basics, Components, Protocols, and Applications

125. MCQs on IoT Protocols
126. IoT MCQs
127. INTRODUCTION Block Chain Technologies MCQs
128. Understanding Block chain with Crypto currency MCQs
129. Understanding Block chain for Enterprises MCQs
130. Enterprise application of Block chain MCQs
131. Block chain application development MCQs
132. MCQs on Service Oriented Architecture, Web Services, and Cloud Computing
133. Utility Computing, Elastic Computing, Ajax MCQs
134. Data in the cloud MCQs
135. Cloud Security MCQs
136. Issues in cloud computinG MCQs
137. Introduction to modern processors MCQs
138. Data access optimizations MCQs
139. Parallel Computing MCQs
140. Efficient Open MP Programming MCQs
141. Distributed Memory parallel programming with MPI MCQs
142. Review of Object Oriented Concepts and Principles MCQs.
143. Introduction to RUP MCQs.
144. UML and OO Analysis MCQs
145. Object Oriented Design MCQs
146. Object Oriented Testing MCQs
147. CVIP Basics MCQs
148. Image Representation and Description MCQs
149. Region Analysis MCQs
150. Facet Model Recognition MCQs
151. Knowledge Based Vision MCQs

152. Game Design and Semiotics MCQs
153. Systems and Interactivity Understanding Choices and Dynamics MCQs
154. Game Rules Overview Concepts and Case Studies MCQs
155. IoT Essentials MCQs
156. Sensor and Actuator MCQs
157. IoT Networking & Technologies MCQs
158. MQTT, CoAP, XMPP, AMQP MCQs
159. IoT MCQs: Platforms, Security, and Case Studies
160. MCQs on Innovation and Entrepreneurship
161. Innovation Management MCQs
162. Stage Gate Method & Open Innovation MCQs
163. Innovation in Business: MCQs
164. Automata Theory MCQs
165. Finite Automata MCQs
166. Grammars MCQs
167. Push down Automata MCQs
168. Turing Machine MCQs
169. Database Management System (DBMS) MCQs
170. Relational Data models MCQs
171. Data Base Design MCQs
172. Transaction Processing Concepts MCQs
173. Control Techniques MCQs
174. DBMS Concepts & SQL Essentials MCQs
175. DESCRIPTIVE STATISTICS MCQs
176. INTRODUCTION TO BIG DATA MCQ
177. BIG DATA TECHNOLOGIES MCQs
178. PROCESSING BIG DATA MCQs

179. HADOOP MAPREDUCE MCQs
180. BIG DATA TOOLS AND TECHNIQUES MCQs
181. Pattern Recognition MCQs
182. Classification Algorithms MCQs
183. Pattern Recognition and Clustering MCQs
184. Feature Extraction & Selection Concepts and Algorithms MCQs
185. Pattern Recognition MCQs
186. Understanding Cybercrime Types and Challenges MCQs
187. Cybercrime MCQs
188. Cyber Crime and Criminal justice MCQs
189. Electronic Evidence MCQs
190. Data Science MCQs
191. DBMS Normalization MCQs
192. Advanced Computer Architecture MCQ
193. Introduction to Information Security MCQ
194. Computer Graphics Multimedia PYQ
195. HTML MCQs
196. Basic concepts of OOP MCQS
197. File System MCQs
198. Social Issues and the Environment mcqs
199. FM Modulation & Transmission MCQs
200. Introduction to ICs and Op-Amps MCQs