Java, Selenium and Cucumber framework-

UHG -

Java programs –

1. Armstrong Program
2. How to delete only special characters from the string. write code
3. Write code for the n\*n\*n program and how you will input values.
4. Three inners for loops and if for most inner loop break statement is executed. write code, and it should come out of all three loops
5. Write class having encapsulation
6. Static topic
7. Final topic
8. 5 Interfaces in selenium
9. Write code for re-execution logic in selenium
10. Data table vs. Scenario outline in BDD Framework
11. Write code for drag-and-drop selenium.
12. Write all git commands. How do you check out, check-in, merge, commit, create PR
13. Difference between abstraction and interface
14. Linked list structure
15. What is a linked hashmap
16. When ArrayList can be used and when linked list
17. What are ArrayList and link lists in collections, and what’s the difference?
18. Diff. b/w List, Queue, Set.
19. Why SET<> doesn’t allow duplicate values? Tell me it’s background implementation.
20. In a string “ABC DEF,” reverse each word without any inbuilt function.
21. Tell me your framework project structure.
22. Tell me the procedure for interacting with Excel files.
23. Types of Xpath.
24. Write an XPath using content following siblings.
25. How to perform parallel execution.
26. Explain your project.
27. What is the difference between the star and snowflake schema?
28. What is meant by clustered index?
29. What is the difference between primary and unique keys?
30. What is meant by integration check?
31. Explain smoke and sanity tests.
32. Explain is subqueries.
33. What are the different types of transformations available in Informatica?
34. Explain the dim table with an example.
35. What is the difference between DWH and DW?
36. If two tables have the same columns, how do you get common columns without using joins?
37. In your project, how many dim and fact tables do you have?
38. How to find out 3rd highest salary?
39. Few selenium exceptions.
40. What are your roles and responsibilities?
41. What is STLC?
42. What are the primary key and unique keys?
43. What is referential integrity?
44. Tell me, Unix commands?
45. How can you find out the orphan records?
46. What are metric validations?
47. What are star schema and snowflake schema?
48. What slowly changing dimensional tables?
49. What is a mean junk-dimensional table?
50. Data flow of your projects?
51. Find out the 3rd highest salary.
52. Tell me about your project.
53. Which validation are you performing between ods and staging tables?
54. Are you using aggregate operations without group by clause?
55. Sel sum(sal) from emp group by Deptno where deptno=10
56. Having count(\*)>1, what is the order of execution?
57. How do you get the top five records in Unix?
58. How to find duplicate records?
59. How to delete duplicate records?
60. What is the difference between delete and truncate?
61. Write syntax for instr and substr?
62. How do you show which constraints you have used on a table in oracle DB?
63. What is the difference between a filter and a router?
64. What is meant by active and passive transformations?
65. Diff. b/w thread & wait.
66. Perform drag and drop between two frames.
67. Types of wait statements.
68. Reporting in selenium.
69. How to switch b/w Tabs.
70. How do you execute scripts, Ajax using Java?
71. Have you made webdriver static or not?
72. An example of an Abstract class & Interface class.
73. How can we create a method with a body in the Interface?
74. From which Java version can we create a static method with a body in Java?
75. Selenium version you are working on.
76. Explain method overriding and method overloading
77. Difference between put and post
78. Difference between run and compile polymorphism
79. Bug life cycle
80. Scenario-based test cases
81. SQL outer joins
82. Java program reverse string
83. What is a test runner in a cucumber?
84. WAP to count the number of characters in a string
85. The interviewer will give some code and ask what the error in the code
86. The interviewer will ask for output for some code.
87. WAP to count duplicate characters in a string
88. SQL queries – join, update
89. How to setup a pipeline using Jenkins
90. TestNG questions
91. How to handle multiple windows using selenium
92. Why do we use to try catch?
93. What are the different methods In API?
94. Difference between put and post
95. Why do we do API testing?
96. What are the different listeners in Jmeter for performance testing?
97. How do you resolve conflict if the developer disagrees and says it’s not a bug?
98. What do you do when you don’t have much time to test and the build will deliver to the client in EOD?
99. What challenges do you face in these four years in your career, and how do you resolve them?
100. What is agile? How it’s different from a waterfall
101. Have you faced any challenges in Agile?
102. How do you handle the pressure?
103. What are your strength and weakness?
104. Tell me about yourself.
105. Explain previous project frameworks and domains you worked in.
106. Explain which type of selenium framework you worked in.
107. Gave me a java program and asked me to provide the output.
108. He asked me to write a program where I have to use add(), get(), remove(), and size() methods on a dynamic array in the same program.
109. How many TestNG annotations are available, and when should we use them?
110. What is the significance of actions class in selenium?
111. Why hashcode is used in java.
112. Asked me about the BDD framework, like why the BDD framework is used, what type of language you will use in the feature file of the BDD framework, and how you will establish a connection between the feature file and the step definition file.
113. They asked me to write a scenario in the feature file format, and the methods inside the feature file will represent the methods in step definition and types of keywords used in the feature file.
114. Difference between scenario and scenario outline keywords in the feature file.
115. How can we use multiple data in the feature file?
116. Explain the Defect lifecycle.
117. Asked me how familiar I was with SQL. They gave two tables and asked them to explain the output for different kinds of joins.
118. What is the difference between union and union All?
119. When will you use where clause, having clause and difference between them?
120. SQL: Which comes first, having or where?
121. Asked output for query – Select eight from the table.
122. Asked about basic GIT commands.
123. Asked about Jenkins integration with maven
124. My due diligence-
125. Tools QA selenium and bdd questions and extent reports.
126. Framework and programs which I have written, in eclipse.
127. Java qs..

“^when user enters \“(.\*)\” and ()$”

Given user is already logged in to the application.

And user enters username and password.

|userName|password|

|Anjalimishra|Rudrasun.123|

@Runwith(Cucumber.class)

@CucumberOptions(

features={“src/test/resources/AppFeatures”},

glue={“stepdefinitions”,”AppHooks”},

plugin={“pretty”,

“com. aventstack. extentreports.cucumber.adapter.ExtentCucumberAdapter:”,

“timeline:test-output-thread/”

}

)

Public static ThreadLocal<> tldriver= new ThreadLocal<>();

public static ThreadLocal<>tldriver= new ThreadLocal<>();

WebDriverManager.firefoxdriver().setup();

Tldriver.set(new ChromeDriver());

Public static synchronized WebDriver gerDriver(){

Return tLDriver.get();

}

WebDriverManager.firefoxdreiver().setup()

WebDriverManger.chromedreiver().setup();

tlDriver.set(new ChromeDriver());

@Given(“User enters \“(.\*)”\ and \”(.\*)\”)

Public static void enterusernameAndPassword(){

}

Given user is on Home Page

When User Navigate to login page

And User enters “<username>” and “<password>”

Then message will be displayed login successfully

Examples:

|username|password|

|testUser1|rudrsun.123|

|testUser2|rudrasun.456|

@RunWith(Cucumber.class)

@CucumberOptions(

features

)

List<Map<String, String>>data= usercredentials.asMaps(String.class, String.class) throws throwable {

List<<>> data = usercredentials.asMaps(String.class, String.class)

}

npm install -g protractor

nodejs.org

protractor –version

npm install -g protractor.

Npm install -g protractor.

Npm install -g protractor

Protractor conf,js –suite chrome 2.36

Appdata Roaming npm node\_modules protractor

Appdata roaming npm node modules protractor

Appdata roaming npm node module protractor.

Webdrive – manager update

Webdriver – manager update

Webdriver -manager update

Webdriver – manager update/

**Rest Assured –**

End point: Address where API is hosted on the Server.

HTTP methods which are commonly used to communicate with Rest API’s are

**GET, POST, PUT, and DELETE**

GET- The GET method is used to extract information from the given server using a given URI. While using GET request, it should only extract data and should have no other effect on the data. No Payload/Body required

**How to send input data in GET?**  
Ans: Using Query Parameters

POST- A POST request is used to send data to the server, for example, customer information, file upload, etc. using HTML forms.

**How to send input data in POST?**  
Ans: Using Form Parameters /Body Payload

PUT- Replaces all current representations of the target resource with the uploaded content.

DELETE- Removes all current representations of the target resource given by a URI.

**Resources:  
Resources represent API/Collection which can be accessed from the Server**

Google.com/maps  
google.com/search  
google.com/images

**Path Parameters:**  
**Path parameters** are variable parts of a URL path. They are typically used to point to a specific resource within a collection, such as a user identified by ID

<https://www.google.com/Images/1123343>  
<https://www.google.com/docs/1123343>  
<https://amazon.com/orders/112>

<https://www.google.com/search?q=newyork&oq=newyork&aqs=chrome..69i57j0l7.2501j0j7&sourceid=chrome&ie=UTF-8>

**Query Parameters:**  
Query Parameter is used to sort/filter the resources.

Query Parameters are identified with?””

https://amazon.com/orders?sort\_by=2/20/2020

**Headers/Cookies**:

Headers represent the meta-data associated with the API request and response. In layman terms, we were sending Additional details to API to process our request.  
Example : Authorization details

**End Point Request URL can be constructed as below**  
Base URL/resource/(Query/Path)Parameters