**1.Narasimhulu**

*Good profile introduction*

*Strong in problem solving portion in automation*

*Strong in most of the automation tool*

*Good knowledge in multi tier architecture*

*Good interpretation skills wrt application*

*Mobile Testing*

*Java 8 Certification & Innovation ideas*

**2.Java Programming :**

1. *Strong understanding over****OOPS concept ,***

Object Oriented Programming is a programming concept that works on the certain principle where objects will be created for classes and do some operation to achieve business goals through certain rules. We have various concepts in OOPs like

**Class** - Collection of methods and variables (similar type of entities)

**Object** - Instances of class, can have multiple instances of single class - It will have a control of class

**Inheritance** - Acquiring the properties and behavior of other object through parent child relationship

**Extends** - keyword use to inherit one class from another.

Multiple inheritance is not possible due to ambiguity

Multiple inheritance is possible through interface there wont be ambiguity since all methods in interface are abstract by default

**Polymorphism** - Ability of behaving like different type based on situation

Method overloading - Different signature

**Method overriding** - Same signature through inheritance

**Abstraction** - Process of hiding the implementation details and showing only functionality to the user. while driving car we need steering, wheel, engine, etc., no need to know how it working internally

**Abstract class** (0 to 100%) - A class which is declared with the abstract keyword is known as an abstract class in Java.

It can have abstract and non-abstract methods (method with the body).

Abstract method - a method which doesn't have implementation and mentioned as Abstract while declaring in Abstract class

**abstract** **class** Shape{

**abstract** **void** draw();

}

//In real scenario, implementation is provided by others i.e. unknown by end user

**class** Rectangle **extends** Shape{

**void** draw(){System.out.println("drawing rectangle");}

}

**class** Circle1 **extends** Shape{

**void** draw(){System.out.println("drawing circle");}

}

//In real scenario, method is called by programmer or user

**class** TestAbstraction1{

**public** **static** **void** main(String args[]){

Shape s=**new** Circle1();//In a real scenario, object is provided through method, e.g., getShape() method

s.draw();

}

}

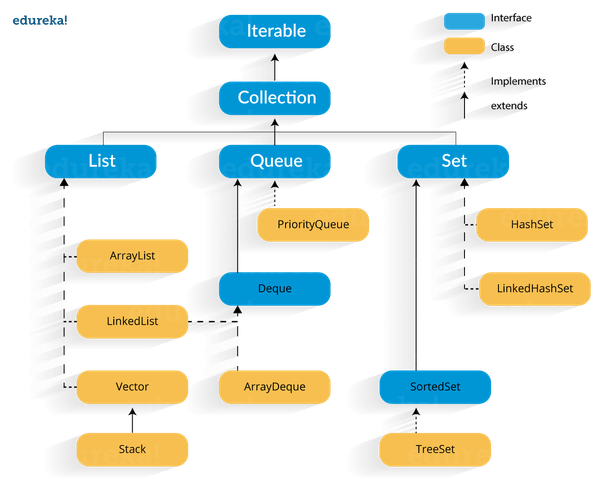
Interface class (100% abstract) - its type of class consist of static constants and abstract methods (only declaration part)

Encapsulation - Wrapping the data and method using access modifier - it can be implemented through private and protected

Difference between call by reference & call by value

1. ***Interface , Strings , I/O operation***
2. *Good understanding over****collection framework****is must.*

**Import java.lang.util.\*;**



Java Map Hierarchy

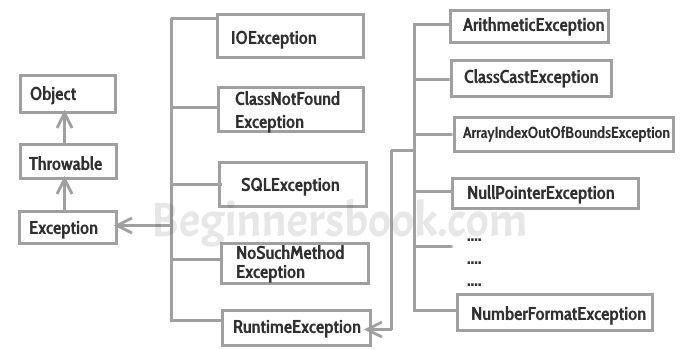
1. ***Data structure & Algorithm****knowledge is must.*
2. *Exception knowledge must*

Exception is an event that interrupts the normal flow of execution. It is a disruption during the execution of the Java program.

Compile time errors

Runtime errors

Exceptoin example : arithmetic exception, Nullpointer exception, Divide by zero exception



**3.Selenium :**

Same Origin Policy , Locator techniques , Handling complex scenarios and programs , Action classes and Java script executor.

**1. Explain about your Current Project?**

2. How you Selected Selenium Tools and Others for Test Automation?

3. Write a code for "Add Manufacturer" Functionality in your current project?

4. Write a code to capture and verify the Conformation message in your Application?

5. write a code to close the Popup Window?

6. write a code to click 3rd link in 2nd frame of your webpage?

7. How you identify the Synchronization problem in your Test Script?

8. How you conduct Data Driven Testing using Selenium?

9. How you conduct Cross Browser Testing?

10. How you are handling Exceptions in your current Project?

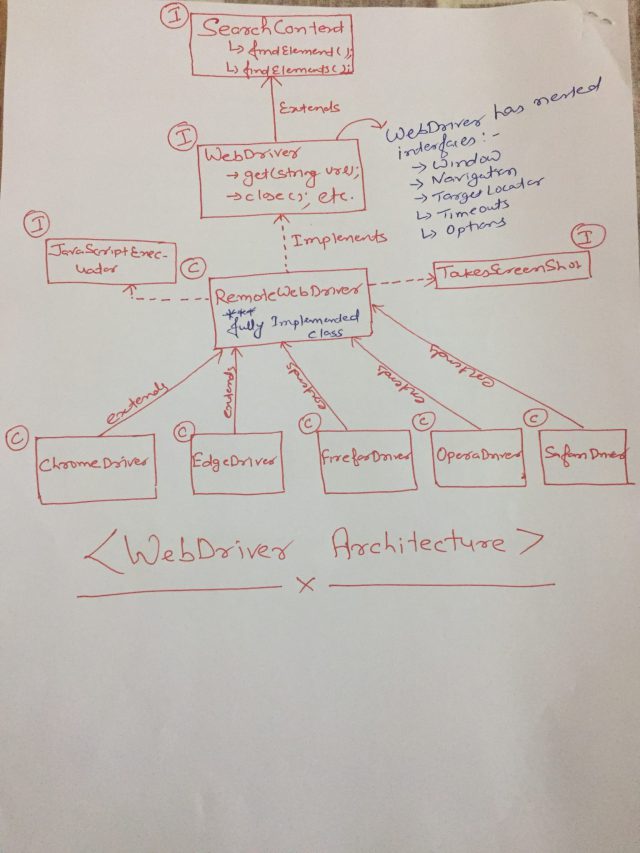
11. What are the challenges that you faced in Test Automation using Selenium?

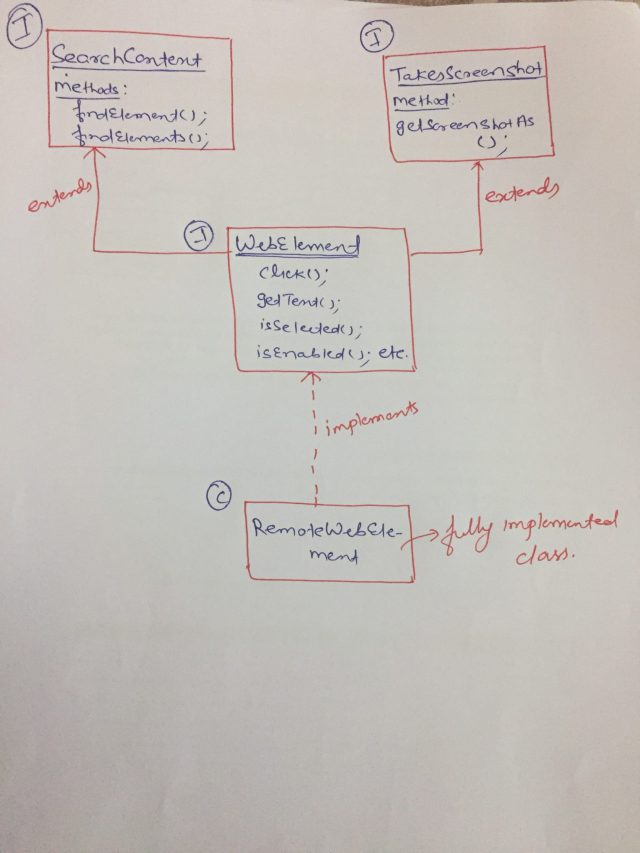
12. How you raise defects that you find in Automated Testing using Selenium?

13. **How you conduct Database Testing?**

14. Explain about the Page Object Model

15. Architecture of Selenium classes and Interface



**WebElement hierarchy** 

<http://makeseleniumeasy.com/2017/04/02/hierarchy-of-selenium-classes-and-interfaces/>

**4**. **Framework components:**

Good understanding over Maven, Jenkins, TestNg

Reporting tools and How to Integrate them together to form a robust framework.

***5.*Framework Development :**

Framework development is very important,

Page Object model

Integration with CI/CD

Jenkins

Implementation of Hybrid and Data driven framework .

***6.Mobile Testing***

***Sticky Notes :***

IndiaBuls Current account

Account : HHLVLC00417600

User id : narasim85@gmail.com

Password : Lakshmi@85

CMDA : <http://www.tnlayoutreg.in/frmHome.aspx#no-back-button>

User ID : 1091380

Password : Asd123

Consumer court portal user id and password

|  |  |
| --- | --- |
| narasim85@gmail.com | Lakshmi@85 |

EPFIndia

UAN: 100259084303

Pranaya@85

Lakshmi@85

Unaras

Lakshmi85 login password

Emai id : narasimhulu.u@fmr.com

Transaction Password :

Citi bank

|  |  |  |
| --- | --- | --- |
| Narasim85 | Pranaya@85 | Narasimhulu.u@gmail.com |

Questions

Wife name : hemalatha

Daugter name : pranaya

AMEX

User ID : narasim85

Password : lakshmi%85

DigiLock

Narasim85

Lakshmi@85

Consumer court port user id and password

narasim85@gmail.com

Lakshmi@85

Consumer court number Chennai - 044-25340063/65

Bajaj Loan Customer id

Flexi personal loan - 20032814

Flexi Home Loan - 61730850 ( 403TRH64374480 Loan amount )

<http://tnlayoutreg.in/home.aspx>

Dummy slip for Indiabulls loan purpose

User ID : 1068005

Password : Enter@85

0004346781009113762 HDFC credit card still in active

Need to confirm from bank

WhizLab : <https://www.whizlabs.com/ocajp-scja/>

Mala Gupta Book : <http://aad.tpu.ru/1955/Java%20books/OCA%20Java%20SE%208%20Programmer%20I%20Certification%20Guide.pdf>