JAVA Programs

## **String**

**Program:1->**

**Input: “My Name is Khan” Output: My**

**Name**

**Is**

**Khan**

**=====🡺**

String s= "My name is khan";

String[] ss = s.split(" ");

//split method will split the string based on space and stores in array

for(int i=0; i<ss.length; i++){

// Iterating array to print the stored stuffs

System.out.println(ss[i]);

// getting value of particular index from array

}

**Program:1->**

**Input: “India.Srilanka.Pakistan” Output: Expected**

**India**

**Pakistan**

**Srilanka**

**=====🡺**

String s= " India.Srilanka.Pakistan ";

String[] ss = s.split("\\.");

//split method will split the string based on space and stores in array

for(int i=0; i<ss.length; i++){

// Iterating array to print the stored stuffs

System.out.println(ss[i]);

// getting value of particular index from array

}

// The (\\.) is used because ‘.’ is a special character in regular expressions, so it needs to be escaped with a double backslash.

=======================================================

**Program:2->**

**Input: “My Name is Khan” Output: reverse the string -> nahk si eman yM**

**=====🡺**

String s= "My name is khan";

String output = "";

// Declare empty string to store values

String[] ss = s.split(" ");

// split the string and store in array

for(int j=ss.length-1; j>=0; j--) {

// Iterate array to get last index (last word)

String sa = ss[j];

// get the word from last index of array and store to a string

for(int i=sa.length()-1; i>=0; i--){

// iterate the string in reverse

char result=sa.charAt(i);

// get the last character from string and store to a variable

output = output+result; // now get the empty string and add the stored character

} above will update the value of output String every time

output = output+" "; // after one word iteration completes, add a space

} //now outer loop continues to next word and performs all above operations

System.out.println(output); // after all iteration print the output value

=======================================================

**Program:1->**

**Input: “Selenium” Output: muineleS**

**=====🡺**

String s= "Selenium";

String output = ""; // Declare empty string to store values

for(int i=s.length()-1; i>=0; i--){ // iterate the string in reverse

char result=s.charAt(i); // get the last character from string and store to a variable

output = output+result; // now get the empty string and add the stored character

} // This will update the value of output String every time

System.out.println(output);

=======================================================

**Program:2->**

**Input: “My Name is Khan” Output: reverse the string without changing the position -> yM eman si nahk**

**=====🡺**

String s= "My name is khan";

String output = ""; // Declare empty string to store values

String[] ss = s.split(" "); // split the string and store in array

for(String sss:ss) { // iterate the array and store in string

for(int i=sss.length()-1; i>=0; i--){ // now the first value from Array(i.e. a string) is iterated in reverse

char result=sss.charAt(i); // get the last character from string

output = output+result; // now get the empty string and add the stored character

} // above will update the value of output String every time

output = output+" "; // after one word iteration completes, add a space

} //now outer loop continues to next word and performs all above operations

System.out.println(output); // after all iteration print the output value

=======================================================

**Program:2->**

**Input: “capgemini” Output: get the occurrence of each character**

**=====🡺**

String s = "capgemini";

Map<Character,Integer> mp = new LinkedHashMap(); // declare a map

for (int i = 0; i < s.length(); i++) { // iterate the string to get each characters

char c = s.charAt(i); // get the character one by one

if (mp.containsKey(c)) { // check if the character is present in map

Integer count = mp.get(c); // if yes, then get the value of it(i.e. count)

mp.put(c, count+1); // add the value by 1

} else { // if key not present in map then

mp.put(c, 1); // add the key with value as 1(count)

} } // iteration continues

System.out.println(mp); // finally print the map

Output

======

{c=1, a=1, p=1, g=1, e=1, m=1, i=2, n=1}

=======================================================

To print the values one by one in map

Iterate map using Entry class and entrySet method

for(Map.Entry<Character, Integer> mpp:mp.entrySet()) {

System.out.println(mpp); }

Output

=====

c=2

a=1

p=1

g=1

=======================================================

**Program:2->**

**Input: “capgemini welcomes you” Output: get the occurrence of each character**

**=====🡺**

String s = "capgemini welcomes you";

Map<Character,Integer> mp = new LinkedHashMap(); //Declare a map to store characters and their count

String[] ss = s.split(" "); // split the string with space and store in array

for(String sss:ss) { // iterate array to get each word from string and store in another string

for (int i = 0; i < sss.length(); i++) { // iterate the string to get each characters

char c = sss.charAt(i); // get the character one by one

if (mp.containsKey(c)) { // check if the character is present in map

Integer count = mp.get(c); // if yes, then get the value of it(i.e. count)

mp.put(c, count+1); // add the value by 1

} else { // if key not present in map then

mp.put(c, 1); // add the key with value as 1(count)

} } } // iteration continues

System.out.println(mp); // finally print the map

Output

======

{c=2, a=1, p=1, g=1, e=3, m=2, i=2, n=1, w=1, l=1, o=2, s=1, y=1, u=1}

To print one by one

===============

for(Map.Entry<Character, Integer> mpp:mp.entrySet()) { //iterate map using entryset

System.out.println(mpp); // print

System.out.println(mpp.getKey()+" "+mpp.getValue()); // print in customized way

Mpp output

=========

c=2 c 2

a=1 a 1

p=1 p 1

g=1 g 1

e=3 e 1

m=2 m 2

i=2 i 2

n=1 n 1

w=1 w 1

l=1 l 1

o=2 o 2

s=1 s 1

y=1 y 1

u=1 u 1

=======================================================

**Program:2->**

**Input: “1, 2, 3” Output: Sum of values (6)**

**=====🡺**

int[] irr = {1 , 2 , 3};

int a = 0;

for(int i=0; i<irr.length; i++){

a=a+ irr[i];

}

System.out.println(a);

Output

=====

6

=======================================================

**Program:2->**

**Input: Compare 2 strings Output: check if s1 is anagram of s2.**

public static void main(String[] args) {

String word1 = "silent";

String word2 = "listen";

if (check(word1, word2)) {

System.out.println(word1 + " and " + word2 + " are anagrams.");

} else {

System.out.println(word1 + " and " + word2 + " are not anagrams.");

} }

public static boolean check(String s1 , String s2){

char[] c1 = s1.toCharArray();

char[] c2 = s2.toCharArray();

Arrays.sort(c1);

Arrays.sort(c2);

return Arrays.equals(c1 , c2); }

Output

=====

silent and listen are anagrams.

=======================================================

**Program:2->**

**Input: Given group arrays Output: find the shortest array and its length**

int[][] arrays = {

{1, 2, 3, 4},

{2, 4, 3},

{1, 10, 20},

{100, 200}

};

int[] shortestArray = arrays[0];

int shortestlength = arrays[0].length;

for (int[] is : arrays) {

if(is.length<shortestlength) {

shortestlength = is.length;

shortestArray = is;

}

}

System.out.println(“Length of shortest array is ”+shortestlength);

for (int i : shortestArray) {

System.out.print(i+" ");

}

Output

=====

Length of shortest array is 2

100 200

=======================================================

**Program:2->**

**Input: Reverse Output: Write a program to reverse an ArrayList.**

public static void main(String[] args) {

List<Integer> list = new ArrayList<>(Arrays.asList(1, 2, 3, 4, 5));

Collections.reverse(list);

System.out.println("Reversed ArrayList: " + list);

}

Output

=====

Reversed ArrayList: [5, 4, 3, 2, 1]

=======================================================

**Program:2->**

**Input: Reverse Output: Write a program to reverse an ArrayList.**

public static void main(String[] args) {

List<Integer> list = new ArrayList<>(Arrays.asList(5, 4, 6, 3, 5));

Integer max = Collections.max(list);

Integer min = Collections.min(list);

System.out.println(max+" and "+min);

}

Output

=====

6 and 3

CORE COMPETENCIES:

* **2+ Years** of experience as software tester in **Manual, Automation.**
* Experience in all phase of **Software Testing Life Cycle (STLC), Software Development Life Cycle (SDLC), Bug life Cycle** and **Agile Methodology**
* Responsible for planning and executing various functional and non-functional testing activities, such as risk-based testing, exploratory testing, browser compatibility testing, system testing, system integration testing, end-to-end testing, UAT (User Acceptance Testing), Black box testing and post implementation verification
* Having good experience in creating scripts in **Selenium Web Driver** using **Java**.
* Experience in frameworks such as **TestNG** and **Cucumber with JUnit.**
* Good experience in **Page Object Model** and **Object-Oriented Programming Concepts.**
* Strong knowledge on **locators** and **xpath.**
* Good Knowledge in handling pop up, multiple window and Excel file
* Experience in working with **SQL** queries for Backend Testing and Validation.
* Knowledge in API Testing using Rest Assured
* Performed Functionality testing, Regression testing and Sanity testing.
* Experience in developing test cases based on requirements in **JIRA**.
* Excellent experience with build tools like **Maven** and managing java automation projects.
* Maintained automation code and resources in source controls in **GIT**.
* Knowledge on scheduling frameworks and execution using **Jenkins**
* Executed automation scripts on different browsers / environments &reported defects results to the team.
* Participated in all the agile ceremonies planning, daily scrum call, delivery and reporting sprint level deliverables.
* Good analytical and communication skills and ability to work independently with minimal supervision and also perform as part of a team.

## AWARDS & RECOGNITION:

**Atos Accolade BRONZE Award:** For exceptional contribution and involvement in organizational initiatives.

## CERTIFICATIONS & TRAININGS:

Course name: **Automation Testing with Selenium Java**

Institute: **GUVI Geeks networks, IITM Research Park**

Duration: **Jan 2024 to Jun 2024**

Link**:** [**https://www.guvi.in/share-certificate/029D76Ex23v891T011**](https://www.guvi.in/share-certificate/029D76Ex23v891T011)

## FUNCTIONAL SKILLS:

* Requirement Analysis • Web-based Reporting Tools • Data Analysis with SQL and Databases
* User Story Development • UI Development • Test case evaluation
* Cross Functional • Strategic planning • Project Management
* QA Process Improvements • Meeting release deadlines

## TECHNICAL SKILLS:

* **Programming Languages**: Core Java
* **Automation Tools**: Selenium WebDriver, Maven, Cucumber, TestNG, JUnit
* **Development Methodologies:** Test-Driven Development (TDD), Behavior-Driven Development (BDD).
* **Database**: SQL
* **Version Control**: GIT
* **Continuous Integration/Deployment**: Jenkins

## WORK EXPERIENCE:

## PROJECT DETAILS:

* **Project Name 1:** Best Buy E Commerce Website **(**<https://github.com/Hameed1206/BestBuyProject.git>)

**Duration:** Apr 2024 to May 2024

**AIM:** To develop a comprehensive Selenium automation framework for the Best Buy e-commerce website, ensuring robust functionality testing, seamless user interactions, and efficient integration with continuous integration tools.

**Tools:** Selenium, Maven, TestNG, GIT and Jenkins

**Responsibilities:**

* Developed and implemented automated test scripts using Selenium WebDriver and TestNG framework to validate functionality across the Best Buy e-commerce website, including navigation, form filling, and order verification
* Conducted thorough testing of web elements such as URLs, menu navigation, page titles, and bottom links to ensure seamless user experience and functionality adherence
* Optimized testing processes and enhanced QA efficiency by integrating continuous integration tools (e.g., Jenkins) to automatically run test suites, ensuring robust and consistent validation of new code deployments
* Implemented headless browser testing (e.g., Headless Chrome) to capture and analyze screenshots during testing phases, facilitating efficient debugging and issue resolution
* Utilized Extent Report to generate detailed test reports, including screenshots, providing stakeholders with clear visibility into test results and facilitating informed decision-making during project lifecycles
* **Project Name 2:** SpiceJet Airlines Website **(**<https://github.com/Hameed1206/SpiceJetProject.git>)

**Duration:** May 2024 to Jun 2024

**AIM:** To develop a comprehensive Selenium automation framework for the SpiceJet Airlines Website, ensuring robust functionality testing, seamless user interactions, and efficient integration with continuous integration tools.

**Tools:** Selenium, Maven, TestNG, GIT and Jenkins

**Responsibilities:**

* Developed and implemented automated test scripts using Selenium WebDriver and TestNG frameworks to validate end-to-end functional validation from login to booking confirmation
* Automated flight search functionalities for different trip types (One-way and Round Trip) by entering dummy origin and destination details, ensuring accurate search results and optimal user experience
* Implemented code to validate essential fields on the booking page such as Check-in, Flight Status, and Manage Booking, ensuring accessibility and functionality for users
* Integrated automated test suite execution using Selenium and scheduling frameworks to ensure continuous validation of flight booking functionalities, enhancing reliability and efficiency of testing processes

EMPLOYMENT HISTORY:

Company Name**: ATOS SYNTEL**

Role**: Associate Consultant**

Duration**: Jan 2022 to Dec 2023**

Project Title: **FedEx (Revenue Execution).**

**Responsibilities:**

* Design automation scripts in selenium for monitoring active data load and processing. Data capture for validation purpose.
* Involved in preparing Test Plan and Test Cases based on business requirement.
* Involved in Automation framework implementing Page objects, Data Driven, Maven using Selenium WebDriver using Core Java.
* Performed Cross browser testing and parallel execution to check the compatibility and stability of the application.
* Worked effectively with Developers, Offshore Testing team and Project Management to achieve High Quality, Release deadlines and QA Process improvements.
* Involved in defect management and identify Blocker defects impacting project confidence.
* Actively participating in ceremonies such as Daily standup call, Monthly meeting with client and Release presentation meeting at end of release.
* Prepare Daily, Weekly and Monthly status reports

## EDUCATION:

Course name: **Bachelor of Engineering (Mechanical)**

Institute: **Saveetha University**

Duration: **Jun 2013 to Jun 2017**