

“SOFTWARE DEVELOPMENT ENGINEER IN TEST: COGNIZANT INTERNSHIP”

Project report submitted in partial fulfilment of the requirement for the degree of

Bachelor of Technology

IN

Computer Science and Engineering

By

Pranjal Sharma (171226)

UNDER THE SUPERVISION OF

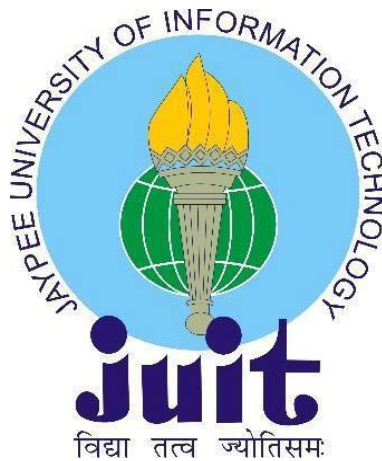
Dr. Mrityunjay Singh

Assistant Professor, JUIT

Mr. Abhijit Joshi

Technical Instructor, Cognizant

To



Department of Computer Science & Engineering and Information Technology

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY WAKNAGHAT

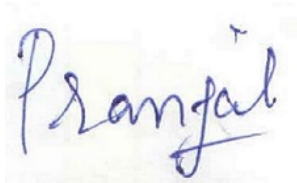
SOLAN-173234

HIMACHAL PRADESH

Acknowledgement

I am highly indebted to all the members of **Computer Science Department, Jaypee University of Information Technology** and **Cognizant** for their steering and constant supervision yet providing necessary data related to the project and additionally for their support in finishing off the project. I would also like to express our gratitude towards **Dr. Mrityunjay Singh**, as the project Supervisor and **Mr. Abhijit Joshi**, my technical instructor, for their kind co-operation and encouragement which helped me in completion of this project and for giving us such attention and time.

Signature:

A handwritten signature in blue ink that reads "Pranjali". The signature is written in a cursive style with a small mark above the 'i' in "Pranjali".

Contents

Project Report Undertaking.....	i
Acknowledgement.....	ii
Chapter 1: Introduction	
1.1 About the Training.....	1
1.2 Role of SDET.....	2
1.3 Technologies Overview.....	2
Chapter 2: Software Testing	
2.1 Agile Testing.....	3
2.2 Functional Testing.....	3
2.3 Automation Testing.....	3
2.4 Performance Testing.....	4
2.5 API Testing.....	4
2.6 Mobile Testing.....	4
2.7 Summary and Learning.....	4
Chapter 3: Java Programming	
3.1 Introduction to Java.....	6
3.1.1 Java Compilation Process.....	7
3.1.2 Features of Java.....	8
3.2 Object Oriented Programming.....	8
3.3 Collection Framework.....	9
3.4 Java Database Connectivity (JDBC).....	12

3.5 Summary and Learning.....	13
Chapter 4: WebUI and Data Source	
4.1 HTML and CSS Fundamentals.....	19
4.2 XML and XPATH.....	21
4.3 JavaScript and JSON.....	21
4.4 Bootstrap.....	23
4.5 SQL and Database Fundamentals.....	23
4.5.1 SQL Joins	25
4.5.2 Common Table Expression and Pivoting.....	26
4.5.3 PL/SQL.....	27
4.6 Summary and Learning.....	27
Chapter 5: Conclusion	
References.....	34

CHAPTER 1

INTRODUCTION

1.1 About the training

The training commenced on 06th March,2021 .A learning path has been laid by Cognizant for the trainees and the learning path for SDET has been designed as per the industry standard requirement for this role. The duration of training for SDET batch is 18 months. After the completion of training an end test will be organised by Cognizant for the rollout of Full Time Employee role.

In the training students are expected to go through various Udemy Courses along with the content being taught by the Technical instructor and the mentor. A set timeline has been shared by the company mentioning objectives for each day. Each day we are expected to complete sections from the Udemy courses, then based on these sections daily assignments called as Hands-on needs to be completed.

By the end of every week, we need to attempt an assessment which opens only when the objectives for the week are fulfilled. The assessments are of two types- Code Challenge (CC) and Integrated Capability Test (ICT). In both these assessments passing marks are 70, below this the student has failed a particular assessment. Both these assessments are proctored. CC can be attempted only once i.e., there's no retest in case a student score below 70. In case of ICT, we have two chances. Weightage of ICT is the highest followed by CC and then hands-on.

1.2 Role of SDET

SDET stands for Software Development Engineer in Test. This role demands professionals who are capable of both software development as well as testing of the software developed. SDETs are a mix of developer and tester with exposure to product management. Quality and high performance codes are expected from an SDET engineer and these codes are useful in designing of the testing framework and even are useful in automation of test cases. SDETs are skilled developers therefore are expected to do the job of developer in test automation as well as job of tester but with White Box Testing. SDETs in context to testing are majorly focused on White Box Testing since having

software development background they are able to understand the working of the code thus making it more feasible for them to test the code based on the working of code. SDETs are expected to be proficient in multiple programming languages, database, user interface, data and product design.

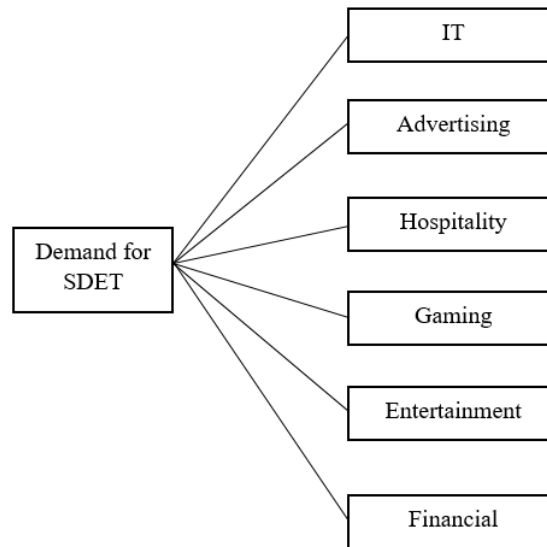


Figure 1. 1 Demand in fields

1.3 Technologies Overview

Software Development Role in Test is a diversified role that involves development of application as well as testing. Therefore, we are not just being trained in one technology but a variety of technologies. The overview of the technologies being taught is as follows:

- Functional Testing
- Java Programming Fundamentals
- Web UI, Data Source
- Spring Core, Spring Boot and MVC
- Micro Services
- Automation Concepts, Selenium configuration, WebDriver
- Selenium Automation Techniques, Dynamic XPath
- Selenium WebDriver with POM and Apache POI
- Automation Testing – Selenium with TestNG
- Digital Technologies

CHAPTER 2

SOFTWARE TESTING

Testing is done so as to evaluate system along with entities or components and check whether that system is satisfying particular requirements or not. The basic purpose of doing testing is to find out errors, gaps, and any missing requirements when compared to the actual needed requirements. If testing is done in the early stage, it will save both cost and time. This testing process is not a time-consuming process. The productive activity in the testing phase is to diagnose and fix bugs or errors that occur in the system. There are many features of a software such as portability, reliability, usability, functionality, etc. and the aim of testing any software is to determine these properties or features.

2.1 Agile Testing

In this testing, agile development is followed in which incremental approach is designed and used for testing. This testing is done when the features of a software are developed.

2.2 Functional Testing

In this testing, each function of a particular software application is verified with the specified requirements. This testing is only concerned with the functions of a software and is not at all concerned with the source code. Test input is used in this type of testing for testing the functionality of the software. This testing aims to check user interface, security, client and server application and functionality of s software that is under test phase.

2.3 Automation Testing

In this type of testing, tester makes use of an automation tool for testing any software. Tester first writes its own script and uses only that script which will be suitable for testing the software. In this testing, repetitive task is performed and implemented without making use of a manual tester. So, we can say that in this testing only automated

tools are used and no human intervention is there. This type of testing is faster and more reliable than manual testing. Test cases that are used in this testing is not complex and requires less maintenance.

2.4 Performance Testing

This type of testing is also known as Perf Testing. This testing analyses quality and capability of a software. In this testing, the main aim is to check whether the software applications are performing and functioning properly or not. In this testing, performance of a particular software is determined in terms of stability, sensitivity and reactivity. The main use of this testing is for removing any performance congestion and it also focuses on enhancing the software before its complete development.

2.5 API Testing

Application Programming Interface (API) comprises of protocols, tools and routines set and specifies how different components are interacting with each other. This type of testing is performed for API collection that is to be tested. This testing verifies the behaviour of API by considering conditions like files and other peripheral devices. In this testing, requirements are checked such as security, performance, and functionality.

2.6 Mobile Testing

Mobile testing is the practise of testing applications for portable mobile devices for reliability, usability, and accuracy. Mobile device monitoring can be either automatic or manual. Mobile testing is a requirement of all product development processes before an app is submitted to the public. This move guarantees that the finished product works well during the customer service journey.

2.7 Summary and Learning

The technologies mentioned in this chapter were a part of week 1 of the complete SDET learning path. I not only got theoretical knowledge during this period, but also solved various assignments and tasks regularly to gain hands on experience of what I learned in this time period.

During this time period I learned the testing technologies and at the end was given two hands on exercises to write test cases for different scenarios for a given website. I successfully identified the test scenarios and wrote well formed test cases to correctly check the application's working. Here is the snapshot of the grading for my submission.

Submission status



Submission status	Submitted for grading	
Grading status	Graded	
Last modified	Sunday, 28 March 2021, 11:36 PM	
File submissions	 hotelbooking1.xlsx	28 March 2021, 11:36 PM
Submission comments	 Comments (0)	

Figure 2.1: Grading of testing assignment.

CHAPTER 3

JAVA PROGRAMMING

3.1 Introduction to Java

Java is a popular third generation programming language that was developed at Sun Microsystems by James Gosling in the year 1995. Currently Java is owned by Oracle and is being run on a lot of devices as of now. Over the years Java has found a wide usage in the IT industry. This language is a class based language with the concepts of Object oriented programming and has been designed as a platform independent language. It facilitates with the feature of “Write Once Run Anywhere”.

```
public class HelloWorld {  
    public static void main(String[] args)  
    {  
        System.out.println("Hello, World!");  
    }  
}
```

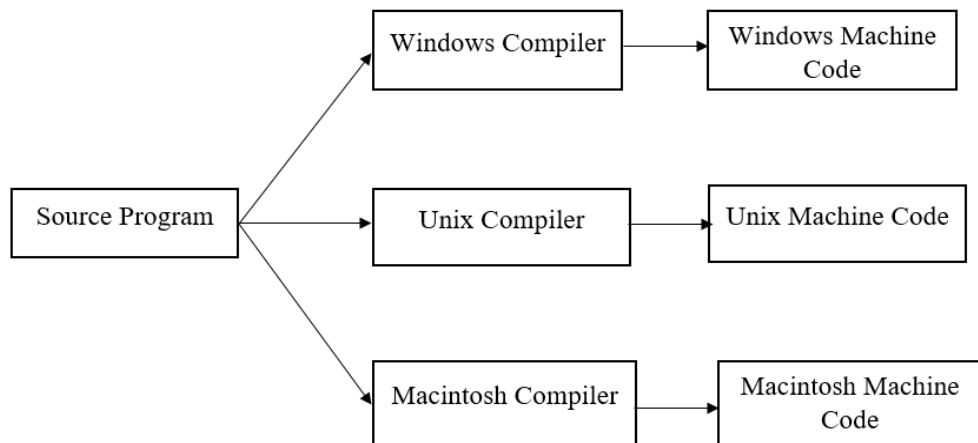
Figure 3. 1 Hello World!

For the above Figure 3.1 of Java syntax the explanation for the keywords used is as follows:

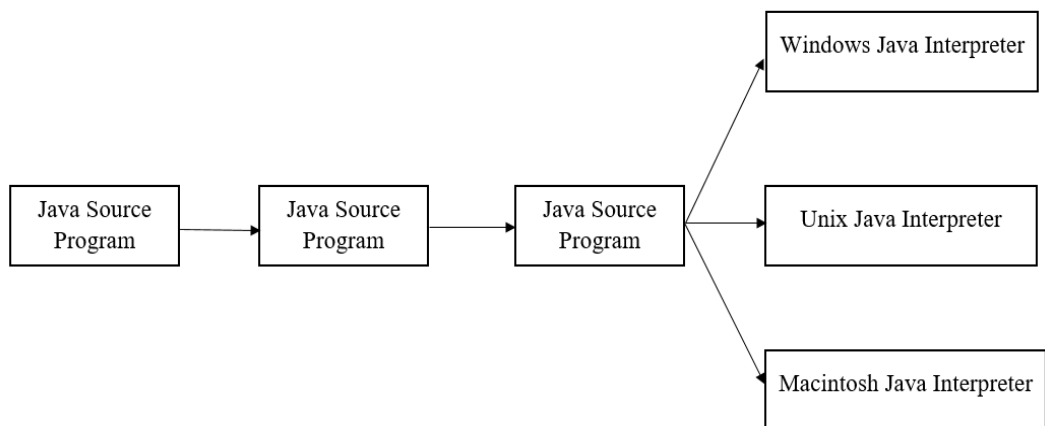
- **class:** The class contains the data and methods to be used in the program. Methods define the behaviour of the class. Class Hello World has only one method Main.
- **static void main():** The keyword static indicates that this function can be accessed without having to instantiate the class.
- **void:** this keyword indicates that this method will not return anything and the entry point of our application is the main() method.
- **System.in:** This is the standard input stream that is used to read characters.
- **System.out:** This is the standard output stream that is used to produce output of a code for the user.
- **println():** This method is used to display text on the console. It prints the text on the console and the cursor moves to the start of the next line of the console.

3.1.1 Java Compilation Process

In Java, ambiguity is tried to be solved by making platform independent applications which means that these applications can be executed on any platform. An ordinary compiler converts the code written by the user into a machine code. This Machine code is a platform dependent code. In case of Java Compiler, the code written by user is converted into Java Byte Code.



In Java, programs are not compiled into executable files instead, they are compiled into bytecode, which is then executed at runtime by the JVM (Java Virtual Machine). When we utilize the javac compiler, we convert Java source code to bytecode. The bytecode is saved on disk as a file with the extension .class. The bytecode is transformed using the just-in-time compiler when the application is ready to start. When the Java classes/bytecode are needed for the first time, the JVM compiles them to machine code and loads them into memory.



3.1.2 Features of Java

- **Simple:** The structure of the Java code is easy to understand.
- **Object-Oriented:** Java follows the four principles of OOPs namely, Abstraction, Encapsulation, Inheritance, Polymorphism.
- **Portable:** Java Byte Code can be carried on to any platform thus making it portable.
- **Platform independent:** Follows the concept of “Write Once Run Anywhere” i.e. once a code is written it can run on any OS.
- **Secured:** Java makes use of encryption system which makes transmission secure.
- **Robust:** Due to the unique compilation process the compiler checks for the program errors and the interpreter checks for runtime errors.
- **Interpreted:** Java is a compiled as well as interpreted language.
- **High Performance:** Java is faster than other interpreted programming languages because Java bytecode is almost a native code.
- **Multithreaded:** Multiple threads within the program manage different parts of code to enhance the capabilities of the program
- **Distributed:** Protocols such as HTTP and FTP have been developed in Java. So functions can be called to access the files from a remote system over the internet.
- **Dynamic:** The files lying on a local drive or on a globally placed system the file may be accessed using internet.

3.2 Object Oriented Programming

Abstraction: Abstraction is the concept used in OOPs wherein the user may see only the important details and the implementation details are hidden from the user. In Java for abstraction Abstract class and Interface are used.

Encapsulation: Abstraction is the concept used in OOPs wherein the data and the code acting on the data are bound as a single unit. The access specifiers used in

Encapsulation:

- **Public:** This modifier can be accessed with same class, in derived class and also outside the class.

- **Private:** This modifier can be accessed with same class but cannot be accessed by derived class or outside the class.
- **Protected:** This modifier can be accessed with same class, in derived class but not outside the class.

Inheritance: Inheritance is the concept used in OOPs wherein a class inherits common properties from other classes thus making the code more precise. There are various levels of inheritance depending on the inheritance between different sub classes. Types of inheritance are as follows:

- **Single Inheritance:** A class inheriting another class.
- **Multi-Level Inheritance:** A class inheriting from another class which itself is a subclass of a base class.
- **Hybrid Inheritance:** It is the combination of multiple and multilevel inheritance.
- **Hierarchical Inheritance:** In this type of inheritance one class has multiple subclasses.

Polymorphism: Polymorphism is the concept used in OOPs wherein a single interface is given multiple definitions.

- **Static Polymorphism:** Occurs at compile time. Method overloading is an example wherein more than one method in a class is given same name but the arguments passed are different.
- **Dynamic Polymorphism:** Occur at run time. Method overriding is an example wherein a child class or a subclass may redefine the methods present in the base class.

3.3 Collection Framework

In Java, a collection is a framework that provides an interface for storing and manipulating group of objects. Java Collections can be used for performing all the operations such as search, sort, insert, delete and manipulation on the data.

- **Class:** It is a user defined blueprint used for creating objects. Methods common to objects of the same type are representation of a class.

Example: ArrayList, Linked List, Hash Set, Linked List Hash Set, Priority Queue, Vector and Tree Set.

- Interface: Interface is the blueprint of the class i.e. it specifies what a Class should do.

Example: Set, List, Queue and Dequeue

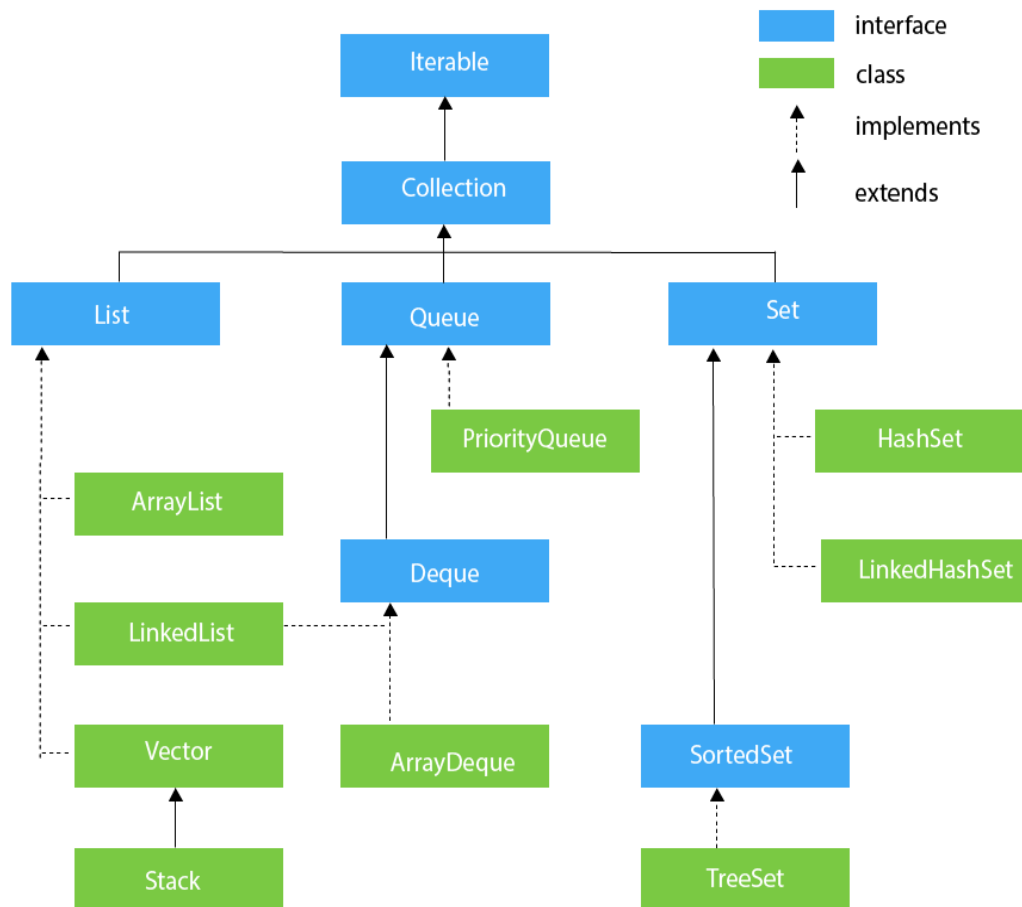


Figure 3. 2 Hierarchy of Collection Framework

The class which are part of the Collection Framework are as follows:

- **ArrayList:** This is a dynamic array available in Java. If the collection increases, the size of an ArrayList is automatically increased, and if the collection shrinks, the size of an ArrayList is automatically decreased. The Java ArrayList helps one to access the list at random. ArrayList is non-synchronized.

```

List <T> al = new ArrayList<> ();
//Where T is the type of the object
  
```

- **LinkedList:** This is the implementation of a linear data structure known as LinkedList wherein every element has a data part and an address part stored in contiguous location.

```
List <T> ll = new LinkedList<> ();  
//Where T is the type of the object
```

- **Vector:** Like ArrayList vector too is a dynamic array available in Java and its implementation too is almost same as to that of ArrayList with the key difference between these two being that Vector is synchronized.

```
List <T> v = new Vector<> ();  
//Where T is the type of the object
```

- **Stack:** This is the implementation of data structure known as stack. Being implementation of stack data structure this class too works on the principle of “Last in first out”. Stack data structure had push and pop operations but this class has additional functions empty, search and peek.

```
Stack<T> stack = new Stack<>();  
//Where T is the type of the object
```

- **PriorityQueue:** PriorityQueue is the implementation of Queue interface. The elements of this queue are to be processed by their priorities and not according to the “First in first out” principle of queues.

```
Queue <T> pq = new PriorityQueue<> ();  
//Where T is the type of the object
```

- **ArrayDeque:** This is a double ended-queue i.e. operations can be performed at both ends. This is a resizable array therefore it has no size restrictions.

```
Queue <T> ad = new ArrayDeque<> ();  
//Where T is the type of the object
```

- **HashSet:** The hash table data structure has its implementation here. The objects inserted here are assigned a hash code and are inserted based on this hash code.

```
Set<T> hs = new HashSet<> ();  
//Where T is the type of the object
```

- **LinkedHashSet:** This is the LinkedList implementation of the interface Set. A LinkedHashSet in order to store the data and retain ordering makes use of doubly linked list.

```
Set<T> lhs = new LinkedHashSet<> ();  
//Where T is the type of the object
```

- **TreeSet:** Here tree is used for storage. TreeSet stores unique elements in ascending order. Over the HashSet the retrieval time of TreeSet is faster.

```
Set<T> ts = new TreeSet<> ();
//Where T is the type of the object
```

3.4 Java Database Connectivity (JDBC)

Java Database Connectivity (JDBC) is an API that is used for connection and execution of a query with a database. The JDBC API connects to the Database using the JDBC Driver as shown in the figure below.

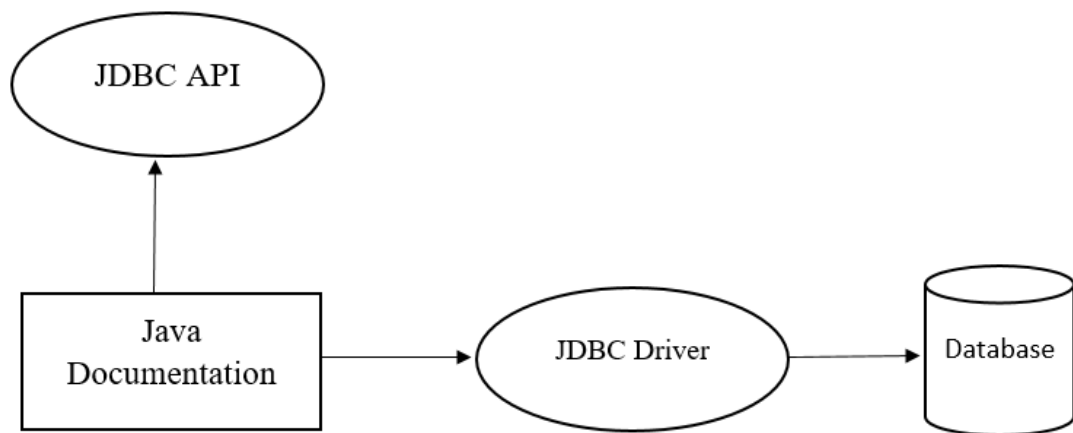


Figure 3. 3 JDBC

The JDBC API is Java based API and the JDBC Driver has been written in Java Language thus making this method a platform independent and secured method. The steps to connect java application with JDBC are as follows:

1. Register the Driver class: `forName()`
2. Create connection: `getConnection()`
3. Create statement: `createStatement()`
4. Execute queries: `executeQuery()`
5. Close connection: `close()`

3.5 Summary and Learning

The technologies mentioned in this chapter were a part of week 2 to week 4 of the complete SDET learning path. I not only got theoretical knowledge during this period, but also solved various assignments and tasks regularly to gain hands on experience

```
import java.util.*;

class SnacksDetails {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int pizza = 100;
        int puff = 20;
        int coolDrink = 10;

        System.out.println("Enter the no of pizzas bought:");
        int pizzaCount = scanner.nextInt();
        System.out.println("Enter the no of puffs bought:");
        int puffCount = scanner.nextInt();
        System.out.println("Enter the no of cool drinks bought:");
        int coolDrinkCount = scanner.nextInt();

        long price = pizza * pizzaCount + puff * puffCount + coolDrink * coolDrinkCount;

        System.out.println("Bill Details");
        System.out.println("No of pizzas:" + pizzaCount);
        System.out.println("No of puffs:" + puffCount);
        System.out.println("No of cooldrinks:" + coolDrinkCount);
        System.out.println("Total price=" + price);
        System.out.println("ENJOY THE SHOW!!!");
    }
}
```

Evaluation Result:

Proposed grade: 100 / 100

Result Description

Summary of tests

```
+-----+
| 6 tests run / 6 tests passed |
+-----+
```

*Note: All the test cases might not have same weightage

Figure 3. 4 Output and Grade for the assignment

```
import java.util.*;

class Product implements Comparable<Product> {
    private final String name;
    private final double price;
    private final double discount;

    public Product(String name, double price, double discount) {
        this.name = name;
        this.price = price;
        this.discount = discount;
    }

    public Double getDiscountPrice() {
        return price * discount / 100;
    }

    @Override
    public int compareTo(Product other) {
        return this.getDiscountPrice().compareTo(other.getDiscountPrice());
    }

    @Override
    public String toString() {
        return name;
    }
}

class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<Product> products = new ArrayList<>();
        int n = scanner.nextInt();
    }
}
```

Proposed grade: 100 / 100

Result Description

Summary of tests

```
+-----+
| 9 tests run / 9 tests passed |
+-----+
```

*Note: All the test cases might not have same weightage

Figure 3. 5 Output for assignment

Count of Each Words

Miss.Jane, an experienced English professor, gives practice tests to her students to improve their written skills. Everyday students write an article and they submit it to Jane. Jane is particular that the students use only special characters like , ; : . ? ! in the article.

[Note : Using the above mentioned special characters will help to split the words in a sentence.

All other special characters when used will be considered as a part of the word itself.]

She counts the total number of words used and the count of each word in the article. Based on this analysis she gives her feedback to the students.

Difficulty arises when the number of students increase. So she wanted to automate the process in the following format. Help her to write a java program to display the words and the number of times it has been used in the article and to display the words using lower case and in alphabetical order.

Enter Student's Article

hello Hello HELLo hi hi: hi! Welcome, welcome

Number of words 8

Words with the count

hello: 3

hi: 3

welcome: 2

Figure 3. 6 Problem Statement for Classes and Object

Proposed grade: 100 / 100

Result Description

Summary of tests

```
+-----+
| 6 tests run / 6 tests passed |
+-----+
```

*Note: All the test cases might not have same weightage

Figure 3. 7 Grade for assignment

```

import java.util.Scanner;

public class Main {
    public static Candidate getCandidateDetails() throws InvalidSalaryException {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Enter the candidate Details");

        System.out.println("Name");
        String name = scanner.next();

        System.out.println("Gender");
        String gender = scanner.next();

        System.out.println("Expected Salary");
        double salary = scanner.nextDouble();

        if (salary < 10000) {
            throw new InvalidSalaryException("Registration Failed. Salary cannot be less than 10000.");
        } else {
            Candidate candidate = new Candidate();
            candidate.setName(name);
            candidate.setGender(gender);
            candidate.setExpectedSalary(salary);

            return candidate;
        }
    }
}

```

Figure 3. 8 Code of assignment

Insurance Bazaar

Insurance Bazaar

Insurance Bazaar is developing an online website for showcasing various types of Insurance policies to their customers based on their needs. There are various types of Insurances provided by different insurance agencies. The admin of Insurance Bazaar wants to add different insurance policy names like Max Bupa Health Insurance, SBI Health Insurance, IFFCO Tokio Two Wheeler Insurance and New India Assurance Two Wheeler Insurance to his database with Policy ID as the Tags.

Customers can view the names of all the policies available in Insurance Bazaar based on the type of insurance.

Write a Java program to simulate this scenario. Key (Policy ID) should be an Integer and Value (Insurance policy name) should be a String. The key-value should be sorted based on the key. Use the appropriate Collection to Store all these details and display.

The **addPolicyDetails** method should add the Policy ID and the Policy name into the appropriate map.

The **searchBasedOnPolicyType** method should return the list of Insurance policy ID depending on the input provided. This method takes the input as string (Input can be either Health or Two Wheeler).

The signature of the above functions are given as part of code skeleton, do not change the function signature.

Figure 3. 9 Problem Statement for Collections Framework

Submission Status

Submitted By	Pranjal Sharma
Grade Acheived	100
Due Date	—
Last Submitted on	Sunday, 11 April 2021, 9:14 PM
Download Submitted File	Download
Download Submission Feedback	No Feedback Available
Download Auto Evaluation Feedback	Evaluate your code to generate feedback

Figure 3. 10 Grade of assessment

Add Flight using JDBC

Zaro Flight System wants to automate the process in their organization. As a start up, they need to automate the flight management system. Help them to develop this application.

You are provided with a public class Flight with following private attribute :

int flightId

String source

String destination

int noOfSeats

double flightFare

Appropriate setter and getter are written.

A public 5 argument constructor with arguments – flightId, source, destination, noOfSeats and flightFare is also provided.

Create a class FlightManagementSystem which has the following method. Use Database for manipulation.

public boolean addFlight(Flight flightObj) - This method should accept a flight object and add that flight details into the database. If flight details are added successfully, return true. Else, return false.

Figure 3. 11 Problem Statement for JDBC

Submission Status

Submitted By	Pranjal Sharma
Grade Acheived	100
Due Date	—
Last Submitted on	Wednesday, 21 April 2021, 6:19 PM
Download Submitted File	Download
Download Submission Feedback	No Feedback Available
Download Auto Evaluation Feedback	Evaluate your code to generate feedback

Figure 3. 12Grade for assignment

CHAPTER 4

WEB UI AND DATA SOURCE

Web development is a part of software engineering. The main focus of web development is to create web pages. In general, web pages, also known as web apps, are codebases that are executed on the browsers whenever a user enters the address for that website. This is in contrary to the normal apps that are present on one's phone. The main components that are essential for a good web page are HTML, CSS, and JavaScript.

4.1 HTML and CSS Fundamentals

Hypertext markup language (HTML) is the foundation of all the web pages that are created. The main purpose of HTML is to represent the basic structure of a web page that is being created and inform the web browser about how that web page is to be displayed when accessed. HTML consists of a number of elements that make up a web page. An element in a HTML code consists of an opening and closing tag. A basic html code along with its output is shown in Figure 4.1.

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

My First Heading

My first paragraph.

Figure 4.2: HTML Example

Cascading style sheets (CSS) is a language that is used in designing any HTML document. CSS is implemented to describe how various elements of HTML can be displayed on screen or paper. It also controls and manages several web pages layout at one go. All the external sheets are stored in CSS files that a user will create. These sheets are stored using file extension .css files. With the help of these external sheets, one can modify or change the entire look of a web page by just doing changes to one file. CSS example is given below in Figure 4.2.

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: lightblue;
}

h1 {
  color: white;
  text-align: center;
}

p {
  font-family: verdana;
  font-size: 20px;
}
</style>
</head>
<body>

<h1>My First CSS Example</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

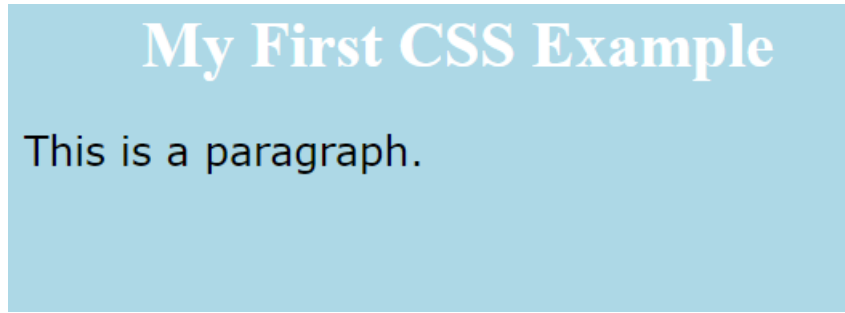



Figure 4.3: CSS Example

4.2 XML and XPATH

XML, short for eXtensible Markup Language, is a markup language that is quite same as HTML itself. XML does nothing in itself, it is just certain information that is written between two tags. The main purpose of XML is to save and transfer the data and is designed to be quite self-explanatory. XML is different from HTML in a way as it focuses on what data is rather than just display the data. Also, the tags in XML are not predefined unlike HTML. XML is used to complement the HTML code.

XPATH, known as the XML path language, is used to find any particular attribute in both HTML and XML documents. There are a total of 7 types of nodes in XPATH. Each XML or HTML document is considered to be a tree with the elements representing the nodes. These nodes are related to each other and represented with specific keywords like parents, children, siblings, ancestors etc. Figure 4.3 shows an example of XPATH using the ChroPath extension on google home page while inspecting the elements.

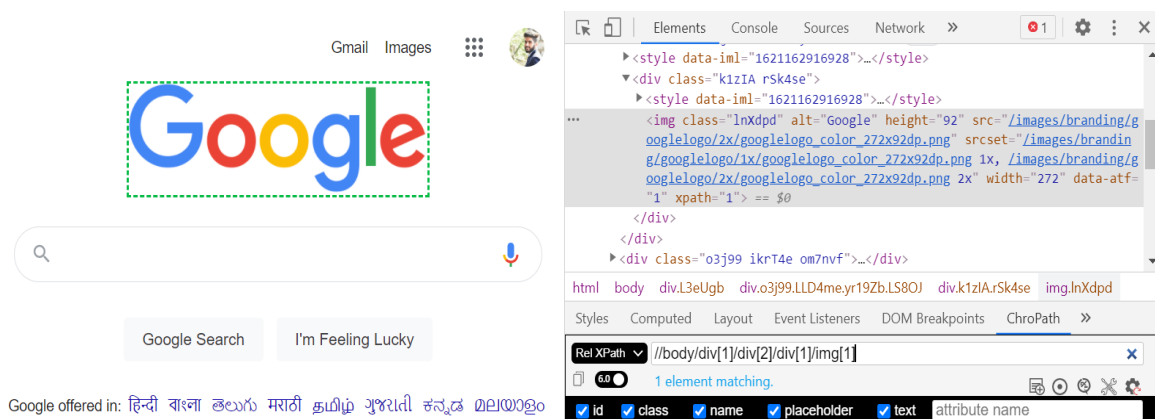


Figure 4.3: XPATH Example

4.3 JavaScript and JSON

JavaScript is one of the most popular programming languages in the world. It is very easy language and is used in web pages for programming their behaviours. JavaScript runs automatically when web page loads. Here scripts are imparted and implemented as plain text only and there is no requirement for any special compilation to run. JavaScript is used for both client and server-side developments. JavaScript encloses library of objects such as math, array, and language elements set such as statements, operators, and control structures. Forms that are created using HTML, are validated using JavaScript. The functions which are used in JavaScript are objects and these objects have the same features like other objects. These can be used as arguments for other functions and can manage date and time. JavaScript adds interactivity to the static web pages as shown in Figure 4.4. The web page changes its contents after clicking the button present on the web page. This demonstrates the use of JavaScript in a web page.

For creating strong web applications, JavaScript was introduced that enhances the browser using new emerging technologies. If we want to get details about anything or we want to explore something, then by just one click we will get the details of that particular thing and this is possible because of the JavaScript language. This language is also useful in creating different games. JavaScript along with HTML is very popular in developing games. They give easy solutions in order to work with heavy graphics. JavaScript is also helping in the field of smartwatch applications that require only internet connectivity for their working and functioning. Further, JavaScript is used for designing or drawing on canvas of HTML.

```
<!DOCTYPE html>
<html>
<body>

<h2>What Can JavaScript Do?</h2>

<p id="demo">JavaScript can change HTML
content.</p>

<button type="button"
onclick='document.getElementById("demo").in
nerHTML = "Hello JavaScript!'">Click Me!
</button>

</body>
</html>
```

What Can JavaScript Do? What Can JavaScript Do?

JavaScript can change HTML content.

Hello JavaScript!

Click Me!

Click Me!

Figure 4.4: JavaScript Example

JSON: JavaScript Object Notation, is used for transmitting a data from a server to a webpage. JSON is an open standard file system and data exchange format that stores and transmits data structures made up of attribute–value pairs and arrays using human-readable text.

4.4 Bootstrap

Bootstrap is one of the most popular frameworks that is used for creating dynamic and responsive web pages. It is a HTML, CSS and JavaScript framework used for the development of front end of the web pages and make the process faster and much easier. It is free to download. The key attributes of Bootstrap are:

- **Consistent:** Provides an aesthetic and subtle look while designing the web pages. It requires minimal effort and results in very appealing websites.
- **Responsive:** While using bootstrap, one need not worry about visual technicalities on different devices. Bootstrap is built in keeping the smaller screens in mind first than the larger screens.
- **Compatible:** Bootstrap has been built on standards and certain browser compliances that make it compatible for every site to work on any platform if implemented properly.

Bootstrap is mainly used for implementing dynamic navbars and the grid system.

4.5 SQL and Database Fundamentals

Structured Query Language (SQL) is a language that is used to access and manipulate databases. SQL can implement various different queries over a database such as retrieving data from a database, inserting record to a particular data, updating records of a data,

deleting or removing records of data from a database, etc. This language allows us to create new tables and store various procedures in a particular database. Further, we can set different permissions for tables, views and procedures. In SQL process, whenever we try to execute any SQL command for any database management system, SQL engine will interpret how to do that task. The different components that are the part of SQL process are: SQL Query Engine, Optimization Engine, Class Query Engine and Query Dispatcher. Figure is given below that shows the basic architecture for SQL process.

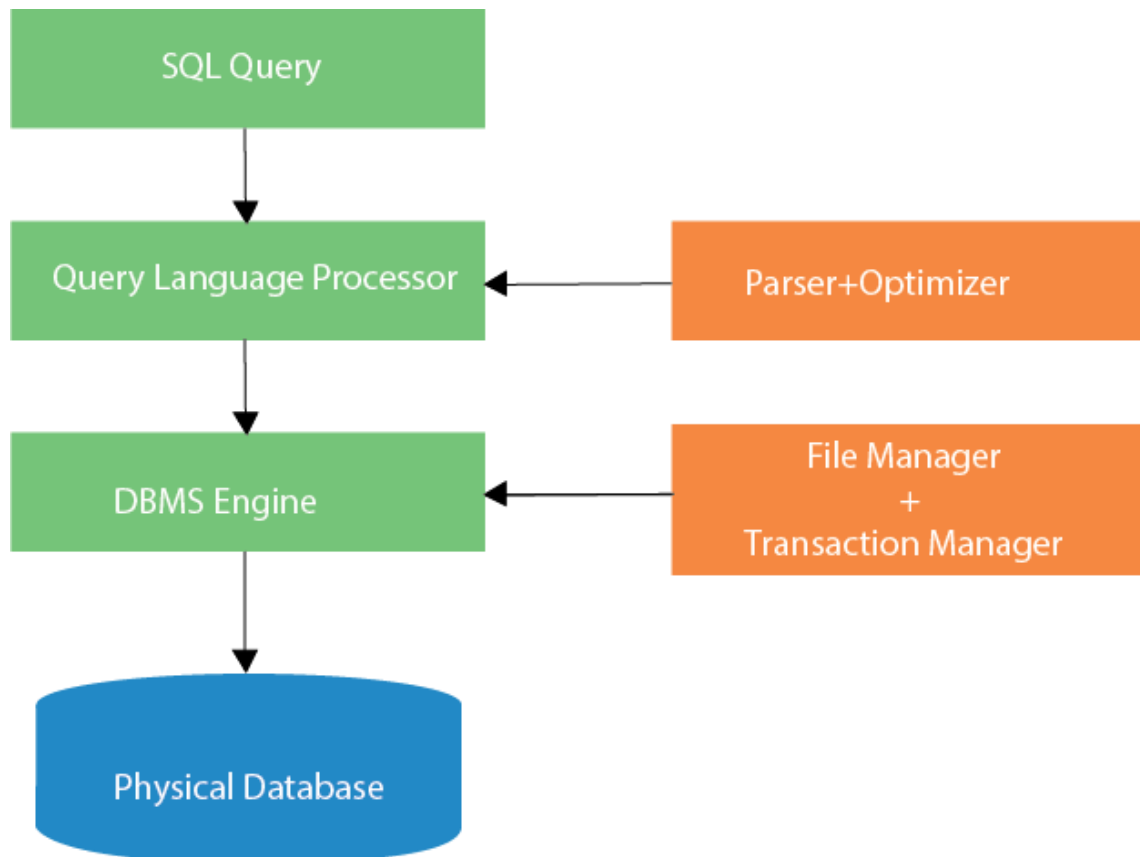


Figure 4.5: SQL architecture

The basic SQL commands that are most commonly used are:

- **DDL** - Data Definition Language includes the following commands:
 - Create:** Used for creating a new table in a database.
 - Alter:** Used for modifying an existing table.
 - Drop:** Used for deleting an entire table.
- **DML** - Data Manipulation Language includes the following commands:

Select: Used to retrieve particular records from a table.

Insert: Used to insert a row into a table.

Update: Used to modify the entries in a table.

Delete: Deletes a record from the selected table.

- **DCL** - Data Control Language includes the following commands:

Grant: Allows to user to do something.

Revoke: Takes back the right from the user.

4.5.1 SQL JOINS

Join statement in SQL is used to combine data from two or more tables using values that are common among them. The commonly used Joins are:

- Inner join
- Left join
- Right join
- Full join

Most commonly and frequently used join statement in SQL is Inner join which is also known as Equijoin. In this Inner join, using join-predicate values of two tables are combined to form a new table. The example of this join is given in Figure. Where Table 1 and Table 2 are considered on which Inner join have been executed to create a new Table.

ID	NAME	AGE	ADDRESS	SALARY
1	Ramesh	32	Ahmedabad	2000.00
2	Khilan	25	Delhi	1500.00
3	kaushik	23	Kota	2000.00
4	Chaitali	25	Mumbai	6500.00
5	Hardik	27	Bhopal	8500.00
6	Komal	22	MP	4500.00
7	Muffy	24	Indore	10000.00

Figure 4.6: "CUSTOMERS" Table 1

OID	DATE	CUSTOMER_ID	AMOUNT
102	2009-10-08 00:00:00	3	3000
100	2009-10-08 00:00:00	3	1500
101	2009-11-20 00:00:00	2	1560
103	2008-05-20 00:00:00	4	2060

Figure 4.7: "ORDERS" Table 2

```
SQL> SELECT ID, NAME, AMOUNT, DATE
FROM CUSTOMERS
INNER JOIN ORDERS
ON CUSTOMERS.ID = ORDERS.CUSTOMER_ID;
```

Figure 4.8: Query to be executed

ID	NAME	AMOUNT	DATE
3	kaushik	3000	2009-10-08 00:00:00
3	kaushik	1500	2009-10-08 00:00:00
2	Khilan	1560	2009-11-20 00:00:00
4	Chaitali	2060	2008-05-20 00:00:00

Figure 4.9: Resulting table from the query

4.5.2 Common Table Expressions (CTE) and Pivoting

CTE is a temporary named result set that is used inside a select, insert, delete or update queries. They can also be used with a view. CTE usually begin using the keyword “with”. It is of two types: Recursive and Non-recursive.

In SQL, the pivot and unpivot operators are used to transform one table statement into another. When we want to move data from row level to column level, we use pivot in SQL, and when we want to translate data from column level to row level, we use unpivot in SQL. The pivot column conducts a clustered grouping based on a list of tacit group-by columns in addition to the pivot column. Columns from the FROM clause that do not feature in any aggregate role or as the pivot column are considered conditional group-by columns.

4.5.3 PL/SQL

PL/SQL is a block-structured programming language. PL/SQL programmes are logical blocks that can have an unlimited number of nested sub-blocks. PL/SQL is an abbreviation for "Procedural Language Expansion of SQL," which is used in Oracle.

Main features of PL/SQL are:

- SQL and PL/SQL are inseparably connected.
- It has a robust error checking system.
- It has a wide range of data formats.
- It has a number of programming constructs.
- It facilitates organised programming through functions and procedures.
- It allows for object-oriented programming.
- It aids in the development of web apps and server sites.

4.6 Summary and Learning

The technologies mentioned in this chapter were a part of week 5 to week 8 of the complete SDET learning path. I not only got theoretical knowledge during this period, but also solved

various assignments and tasks regularly to gain hands on experience of what I learned in this time period.

During week 5, I had to complete 13 assignments that were mainly related to HTML, CSS, XML and XPATH. Below are some of the snapshots of the completed assessments:

Summary of your previous attempts

Attempt	State	Marks / 8.00	Grade / 100.00	Feedback
1	Finished Submitted Friday, 30 April 2021, 4:35 PM	8.00	100.00	Congratulations!! You have pas

Summary of your previous attempts

Attempt	State	Marks / 8.00	Grade / 100.00	Feedback
1	Finished Submitted Friday, 30 April 2021, 4:51 PM	8.00	100.00	Congratulations!! You have passed by securing more than 80%

Highest grade: 100.00 / 100.00.

Figure 4.10: Snapshot of few quizzes of XPATH.

In these quizzes, the aim was to correctly identify the XPATH for the elements of the given web page. I was able to successfully pass all these assignments.

This was followed by code writing using JSON and XML. The task was to design well formed XML codes for the given problem statement. Below are the snapshots of few of the assignments:

Create a well-formed document for the given scenario:

A small business has one department. That department has many employees. The employee details is given in the table below.

Parent Element is "**Department**"

Employee

empid	name	salary	email	phoneno
1001	Tom	20000	tom@gmail.com	9874563210
1002	Sam	25000	sam@gmail.com	7876545676
1003	Shiny	20000	shiny@hotmail.com	9876543210

Figure 4.11: Problem Statement for XML assignment.

Employee.xml

```
1 <Department>
2   <Employee>
3     <empid>1001</empid>
4     <name>Tom</name>
5     <salary>20000</salary>
6     <email>tom@gmail.com</email>
7     <phoneno>9874563210</phoneno>
8   </Employee>
9   <Employee>
10    <empid>1002</empid>
11    <name>Sam</name>
12    <salary>25000</salary>
13    <email>sam@gmail.com</email>
14    <phoneno>7876545676</phoneno>
15  </Employee>
16  <Employee>
17    <empid>1003</empid>
18    <name>Shiny</name>
19    <salary>20000</salary>
20    <email>shiny@gmail.com</email>
21    <phoneno>9876543210</phoneno>
22  </Employee>
23 </Department>
```

Grade

Reviewed on Saturday, 1 May 2021, 10:43 PM by Automatic grade

Grade 100 / 100

Assessment report

[\[+\]Summary of tests](#)

Figure 4.12: Output and Grade for the assignment.

This concluded the assignments for week 5. After successful completion of these 13 assignments, an online proctored assessment was held.

In week 6, the technologies that were to be covered were JavaScript and Bootstrap. A total of 11 assignments followed by 1 final proctored assessment were to be completed in this week. Snapshots of few of the assignments are given below:

Objective:

To add scripting to existing web page and work with JavaScript basic activities. The basic Javascript concepts like DOM, Operators and Control Statements are covered.

Problem Description:

The application manages the payment information of customers. As part of the entire application, one of the business logic done through Javascript is to change the mode of payment, when there's a need of updating the payment mode selected.

Following are the files that contains code snippets.

Index.html	HTML for webpage (complete implementation is given for you). You only have to run this. No change needs to be done in this file.
Script.js	In Script.js file, the replace function is provided as a part of code skeleton. Fill your code in replace function and in wordChecker function to fulfill the requirements.

Procedure to complete the exercise

1. Required function with empty body is already available in the javascript file

Hint : Do NOT change the function names.

Figure 4.13 Problem Statement for JS assignment.

```
!--DO NOT CHANGE THIS FILE-->
<html>
<head>
  <script src="script.js" type="text/javascript"> </script>
</head>
<body>
  <table>
    <tr>
      <td>Address Form</td>
    </tr><tr>
      <td>Name</td>
      <td><input type="text" id="name"></td>
    </tr><tr>
      <td>Street</td>
      <td><input type="text" id="street"></td>
    </tr><tr>
      <td>City</td>
      <td><input type="text" id="city"></td>
    </tr>
    <tr>
      <td><button id="address" onclick=displayAddress()>Display Address</button></td>
    </tr>
  </table>
  <div id="msg"></div>
</body>
</html>
```

Figure 4.14: Output and Grade for the assignment

Proposed grade: 100 / 100

Result Description

Assessment Completed

Grading and Feedback

Work With Javascript Basics - 6.0 / 6.0 (Success)

Design a web page which should contain Bootstrap's the responsive navigation bar, a responsive circle image, and a responsive footer.

Topic Coverage: Bootstrap's responsive navigation bar, circle image, footer and overriding Bootstrap classes.

The snapshot of the web page is given below:

Image 1: Web page in large devices

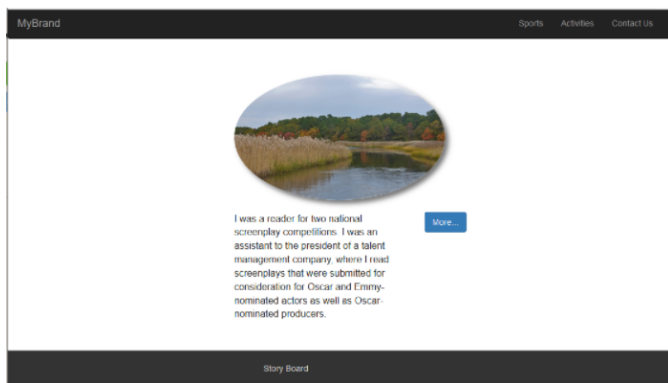


Figure 4.14: Problem statement for the bootstrap assignment

Proposed grade: 100.00 / 100

Result Description

Summary of tests

Successfully Completed. Good Work.

Figure 4.15: Output and Grade for the assignment

In weeks 7 and 8, the primary focus was on learning and implementing SQL concepts. Various DDL, DML, DCL statements were implemented. Hands on exercises of joins and various filtering keywords were used and demonstrated. Below are the snapshots of few of the assignments.

Display specific Customer details

Write a query to display first_name, id, email and phone_number of the customer whose first_name starts with 'M', ends with 'a' and contain a letter 'e' in it. Display the records sorted in ascending order based on the id.

Figure 4.16: Problem statement for the SQL assignment

```
Proposed grade: 100 / 100
Result Description
Summary of tests
+-----+
| 2 tests run / 2 test passed |
+-----+
```

Figure 4.17: Output and Grade for the assignment

This was the end of the WebUI and Data Source module. The module was concluded by a final proctored test wherein it was asked to demonstrate all the technologies learnt during these weeks.

CHAPTER 5

CONCLUSION

Currently I have completed 8 weeks of my SDET training program in Cognizant Technology Solutions. Figure 5.1 marks the completion of my two checkpoints out of four. During this period, I have learnt about Software testing, Java fundamentals, Object oriented programming, HTML, CSS, XML, JavaScript, Bootstrap, and SQL. The learning was not only theoretical, but was accompanied with various hands-on exercises and assessments. The green mark on the figure shows successful completion of the hands-on exercises and the assessments with more than 90% grade.

		CP 1				CP 2				CP 3				CP 4						
		W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18	W19
Hands-on	Due	2	12	11	6	13	11	30		12	6	10	4	6	9	4	3	8	4	7
	Actual	2	12	11	6	13	11	30												
Assessment Type 1	Due		1	1		1	1	1	2	1	1			1	1	1	1		1	
	Actual		1	1		1	1	1												
Assessment Type 2	Due	1			1				1										1	
	Actual	1			1															

Figure 5.1: Current assignment and assessment completion status.

There are total 19 weeks in the complete SDET training path. In the upcoming weeks I will learn about selenium and also complete a minor project, a major project and also a hackathon. After the successful submission of these projects, I will be offered a full-time role in the organization.

REFERENCES

Cognizant Hand Book

Cognizant Udemy

WebEx

Other Cognizant Resources