**Date:**

**Exercise – 1**

**WEBSITE INFORMATION EXTRACTION AND DISPLAY**

**AIM:**

To write a program to display the following information of the given website.

1. Basic Information
2. Web Server Configuration and Session Details
3. Application Details

**PROBLEM DESCRIPTION:**

**1 a) Basic Information**

Extracts and displays key details from a given website URL, including response code, content, meta information, links, server info, IP address, and domain age.

**1 b) Web Server Configuration and Session Details**

Checks and outputs server configuration and session status based on the provided URL.

**1 c) Application Details**

Retrieves PHP version, configuration settings, and resource limits for the server hosting the website.

**SOURCE CODE:**

from flask import Flask, request, render\_template\_string

import requests

from bs4 import BeautifulSoup

import whois

import socket

app = Flask(\_\_name\_\_)

def extract\_website\_info(url):

results = {}

response = requests.get(url)

results['url'] = url

results['response\_code'] = response.status\_code

soup = BeautifulSoup(response.text, 'html.parser')

results['title'] = soup.title.string if soup.title else 'N/A'

results['description'] = soup.find('meta', attrs={'name': 'description'})['content'] if soup.find('meta', attrs={'name': 'description'}) else 'N/A'

results['number\_of\_links'] = len(soup.find\_all('a'))

results['page\_size'] = len(response.content)

results['server\_info'] = response.headers.get('server', 'N/A')

domain = url.split('//')[-1].split('/')[0]

ip\_address = socket.gethostbyname(domain)

results['ip\_address'] = ip\_address

domain\_info = whois.whois(domain)

results['domain\_age'] = domain\_info.creation\_date if domain\_info.creation\_date else 'N/A'

results['server\_software'] = response.headers.get('server', 'N/A')

results['server\_ip'] = ip\_address

results['server\_port'] = 443 if url.startswith('https') else 80

session = requests.Session()

session\_response = session.get(url)

results['session\_id'] = session.cookies.get\_dict().get('sessionid', 'N/A')

results['session\_name'] = 'sessionid' if 'sessionid' in session.cookies.get\_dict() else 'N/A'

results['session\_status'] = 'active' if session\_response.ok else 'inactive'

return results

@app.route('/', methods=['GET', 'POST'])

def index():

if request.method == 'POST':

url = request.form['url']

info = extract\_website\_info(url)

return render\_template\_string(TEMPLATE, info=info)

return render\_template\_string(TEMPLATE, info=None)

TEMPLATE = '''

<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<title>Website Info Extractor</title>

<style>

body {

font-family: 'Arial', sans-serif;

line-height: 1.6;

background-color: #f4f4f4;

margin: 0;

padding: 20px;

}

.container {

max-width: 800px;

margin: auto;

background-color: #fff;

padding: 20px;

border-radius: 8px;

box-shadow: 0 0 10px rgba(0,0,0,0.1);

}

h1 {

color: #333;

text-align: center;

margin-bottom: 20px;

}

form {

text-align: center;

margin-bottom: 20px;

}

label {

display: block;

margin-bottom: 10px;

color: #555;

}

input[type="text"] {

width: 100%;

padding: 10px;

font-size: 16px;

border: 1px solid #ccc;

border-radius: 4px;

}

button {

padding: 12px 24px;

font-size: 16px;

background-color: #007bff;

color: #fff;

border: none;

border-radius: 4px;

cursor: pointer;

transition: background-color 0.3s ease;

}

button:hover {

background-color: #0056b3;

}

ul {

list-style-type: none;

padding: 0;

margin-top: 20px;

}

li {

margin-bottom: 10px;

border-bottom: 1px solid #eee;

padding-bottom: 10px;

}

strong {

width: 150px;

display: inline-block;

font-weight: bold;

color: #333;

}

</style>

</head>

<body>

<div class="container">

<h1>Website Info Extractor</h1>

<form method="post">

<label for="url">Enter Website URL:</label>

<input type="text" id="url" name="url" required>

<br>

<button type="submit">Get Info</button>

</form>

{% if info %}

<h2>Website Information</h2>

<ul>

<li><strong>URL:</strong> {{ info.url }}</li>

<li><strong>Response Code:</strong> {{ info.response\_code }}</li>

<li><strong>Title:</strong> {{ info.title }}</li>

<li><strong>Description:</strong> {{ info.description }}</li>

<li><strong>Number of Links:</strong> {{ info.number\_of\_links }}</li>

<li><strong>Page Size (bytes):</strong> {{ info.page\_size }}</li>

<li><strong>Server Info:</strong> {{ info.server\_info }}</li>

<li><strong>IP Address:</strong> {{ info.ip\_address }}</li>

<li><strong>Domain Age:</strong> {{ info.domain\_age }}</li>

<li><strong>Server Software:</strong> {{ info.server\_software }}</li>

<li><strong>Server IP:</strong> {{ info.server\_ip }}</li>

<li><strong>Server Port:</strong> {{ info.server\_port }}</li>

<li><strong>Session ID:</strong> {{ info.session\_id }}</li>

<li><strong>Session Name:</strong> {{ info.session\_name }}</li>

<li><strong>Session Status:</strong> {{ info.session\_status }}</li>

</ul>

{% endif %}

</div>

</body>

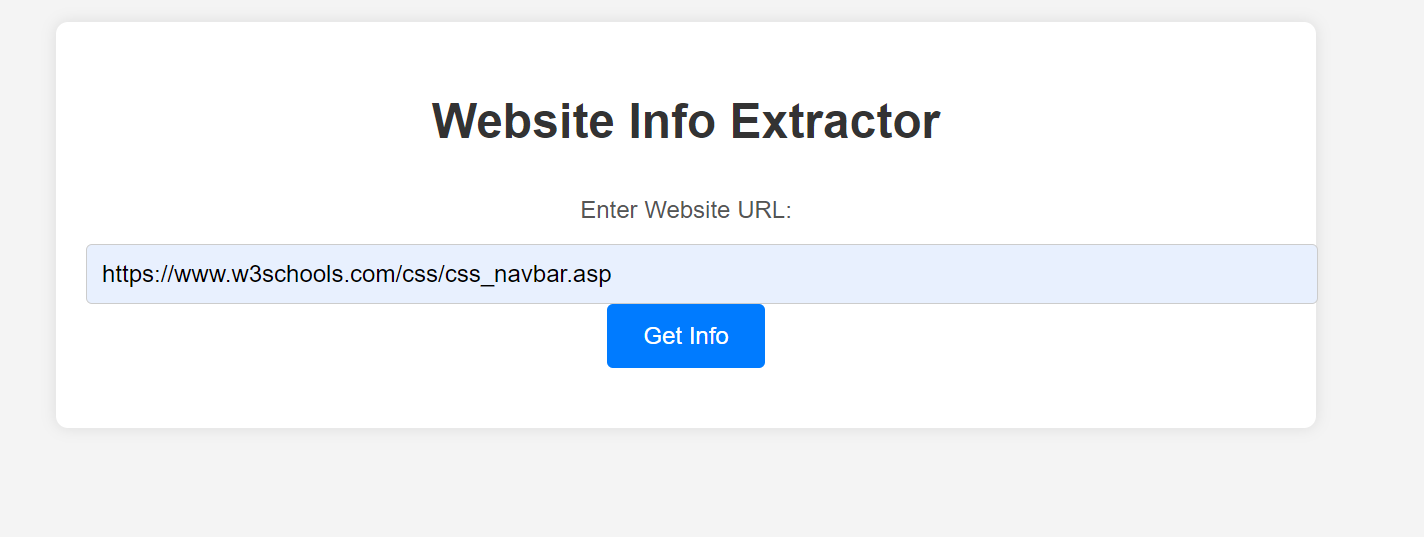
</html>

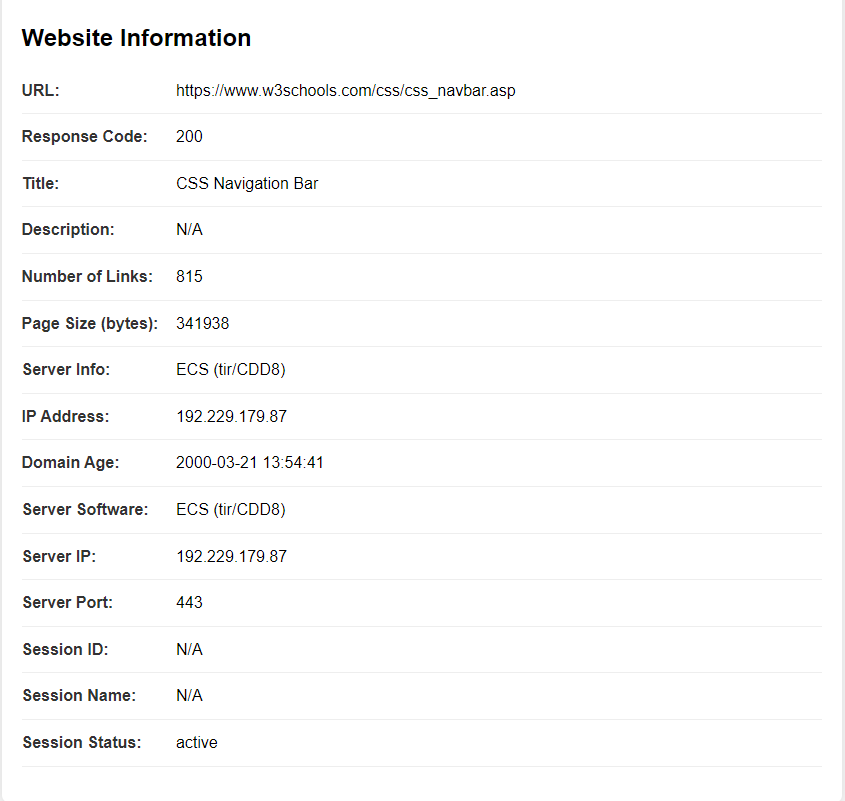
'''

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**OUTPUT:**





**RESULT:**

Thus, basic information, web server configuration details, session details and application details of the given website are displayed and implemented successfully.

**Date:**

**Exercise - 2**

**CREATE A WORDPRESS PERSONAL WEBSITE USING INFINITYFREE WEBHOSTING AND INTEGRATE SSL CERTIFICATION**

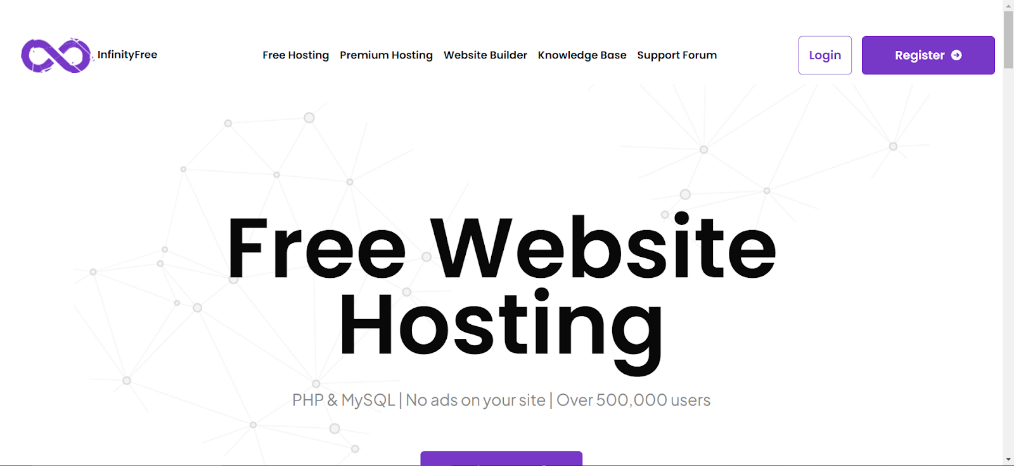
**AIM:**

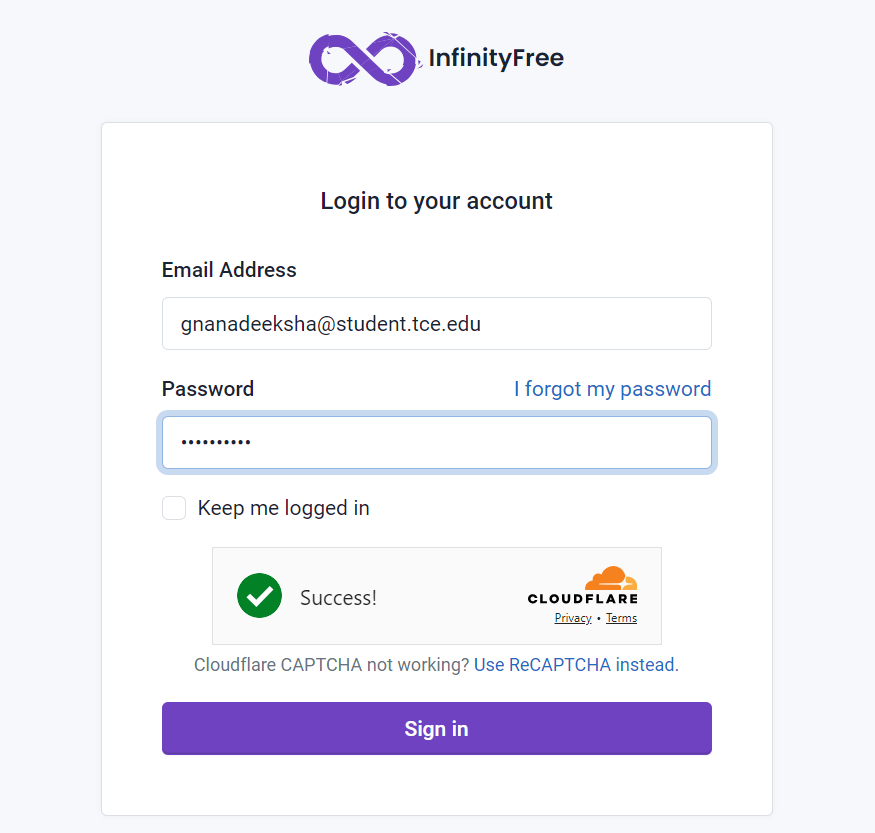
To create a personal website using wordpress, host it on Infinityfree, and secure it with an SSL certificate.

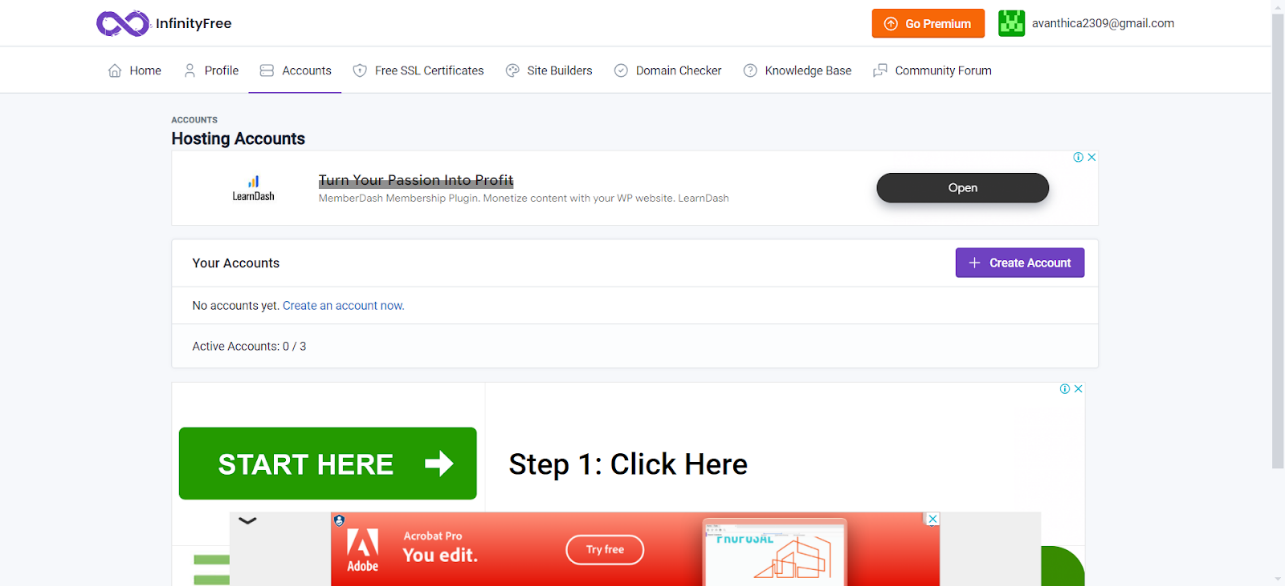
**PROBLEM DESCRIPTION:**

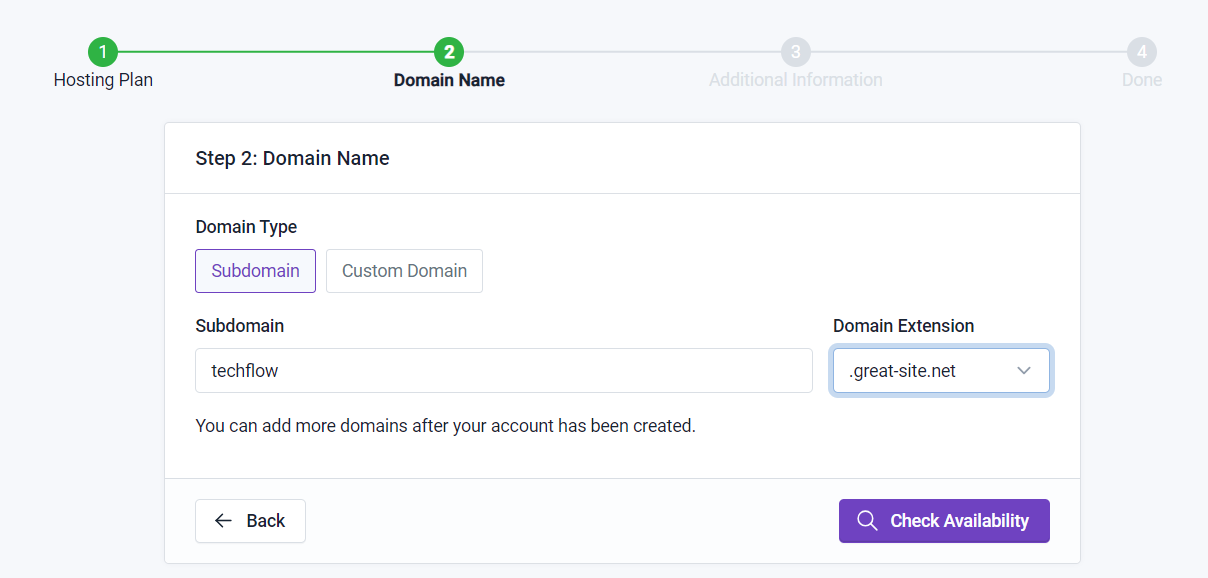
The task involves creating a personal website using WordPress, hosted on Infinityfree, a free web hosting service. The process includes registering for a hosting account, setting up a domain, installing WordPress, and customizing the website with themes and plugins. The website will be secured with an SSL certificate, ensuring encrypted communication. Finally, the site will be reviewed, launched, and shared publicly.

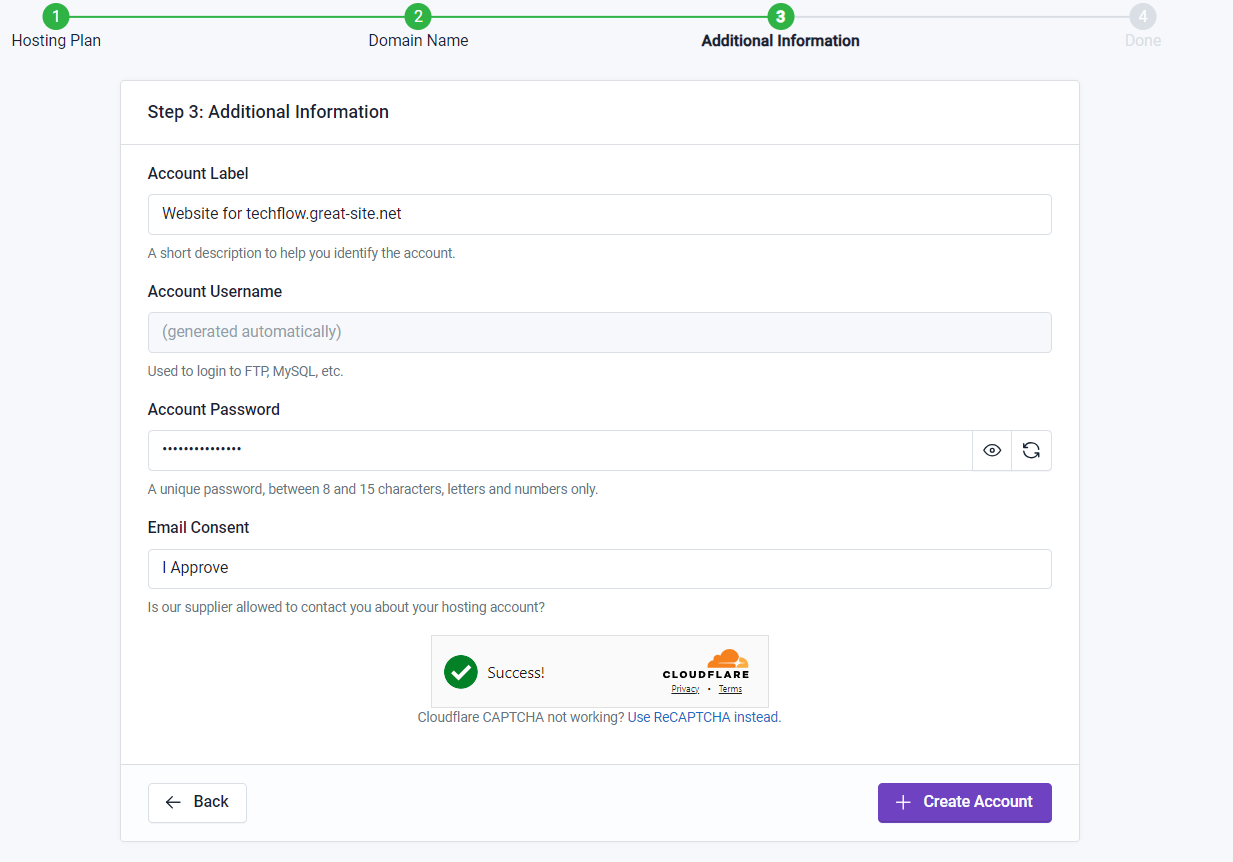
**OUTPUT:**

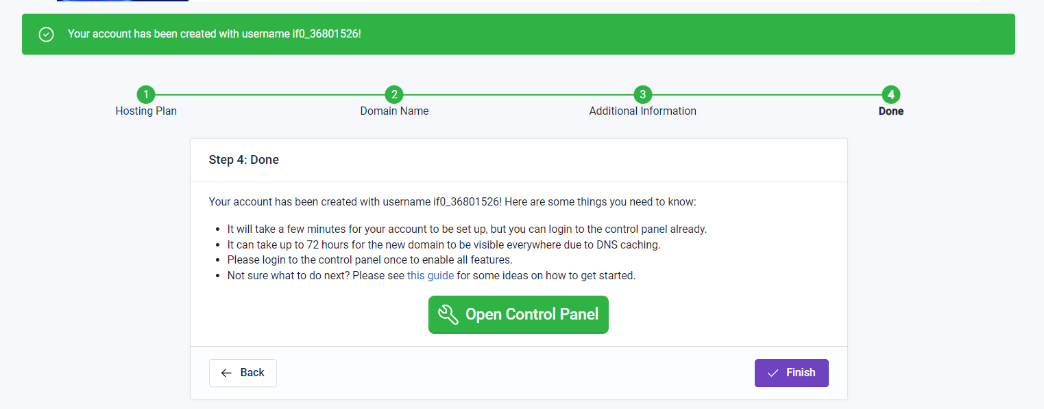


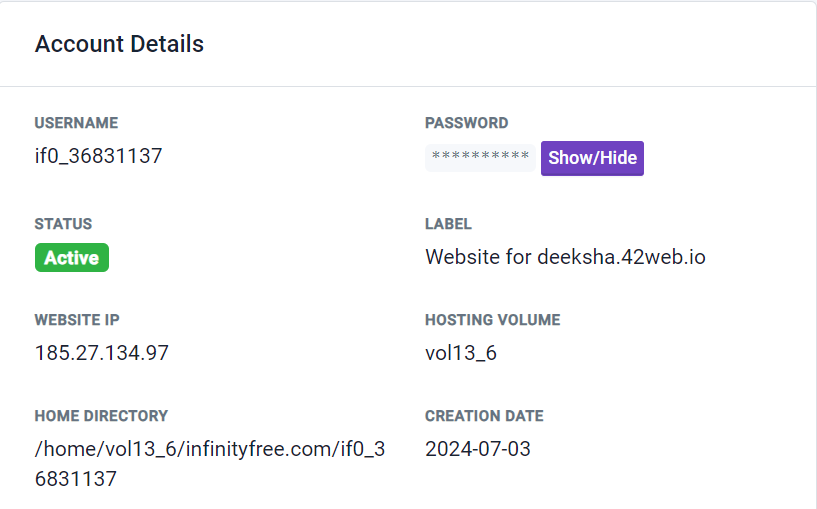


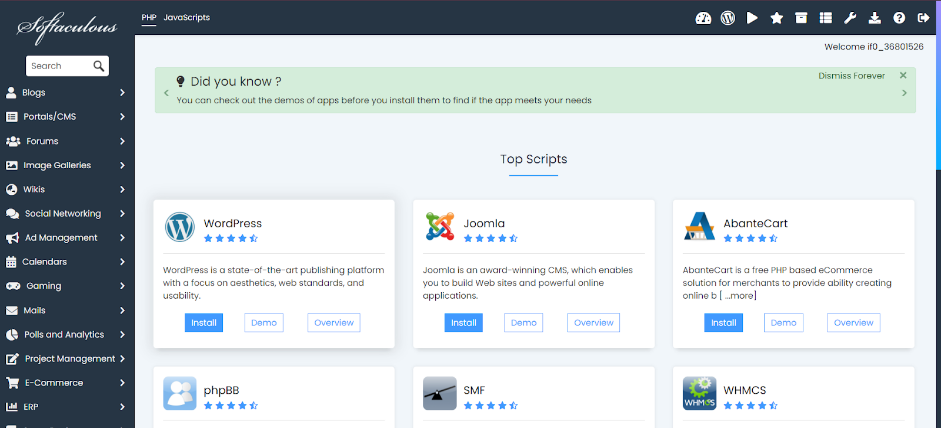


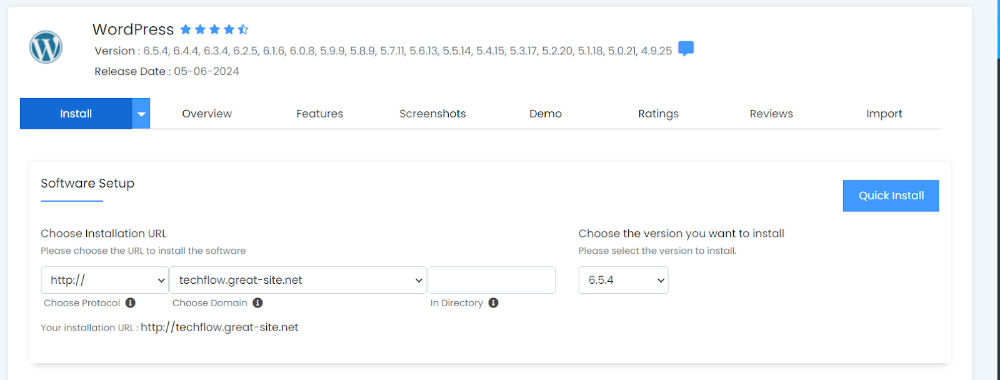


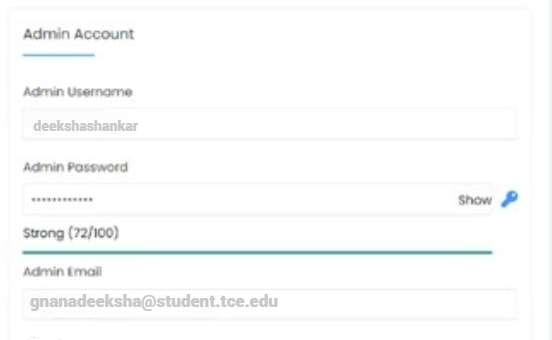


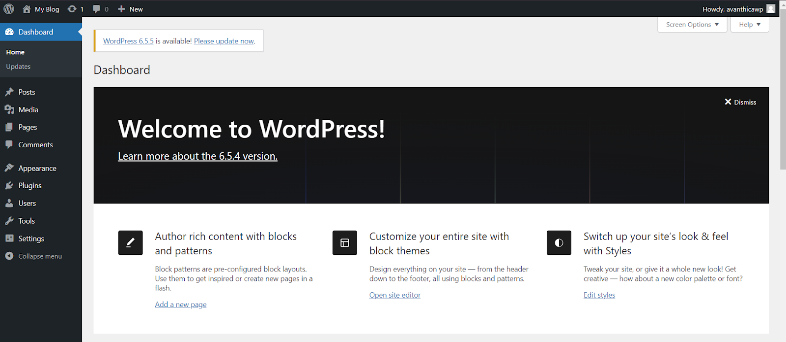


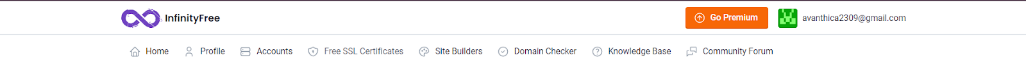


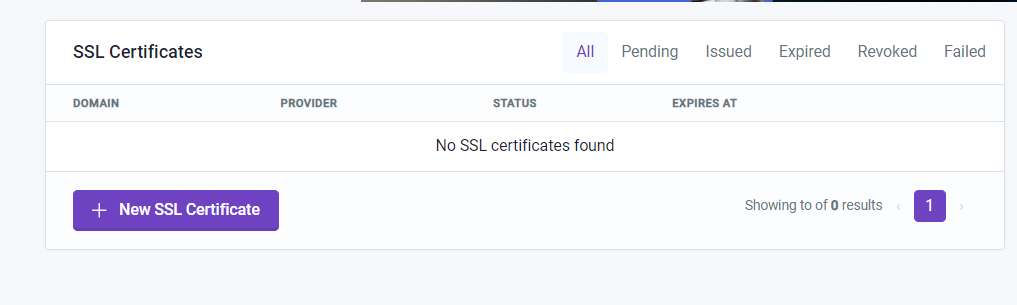


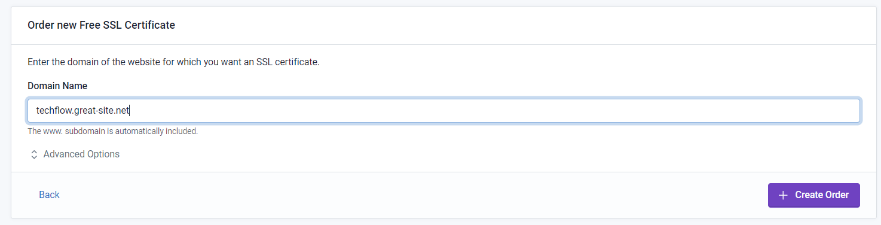


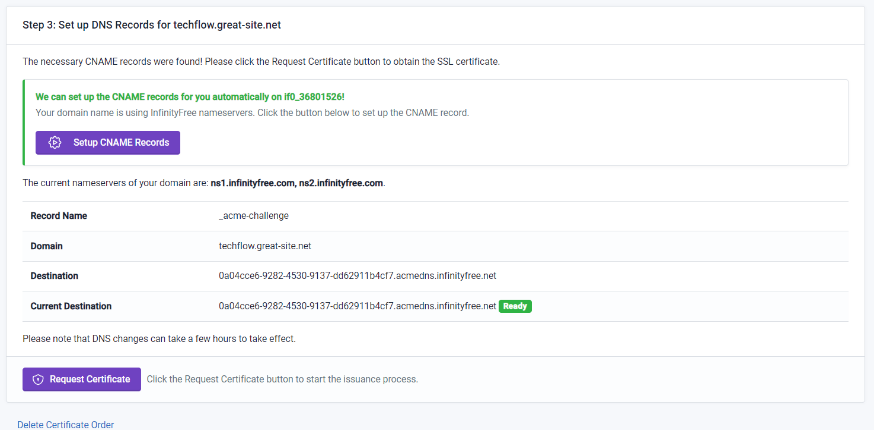


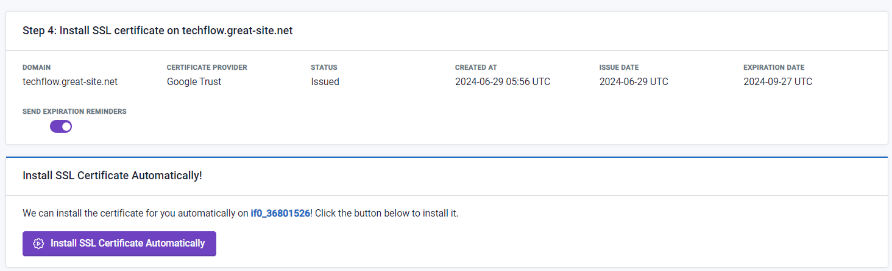


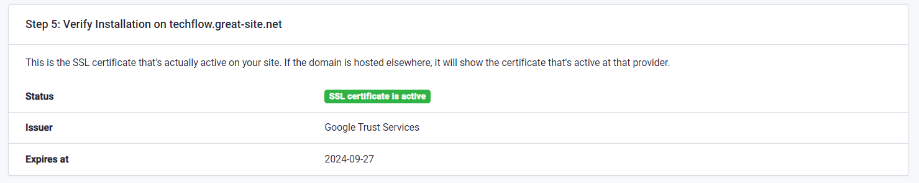


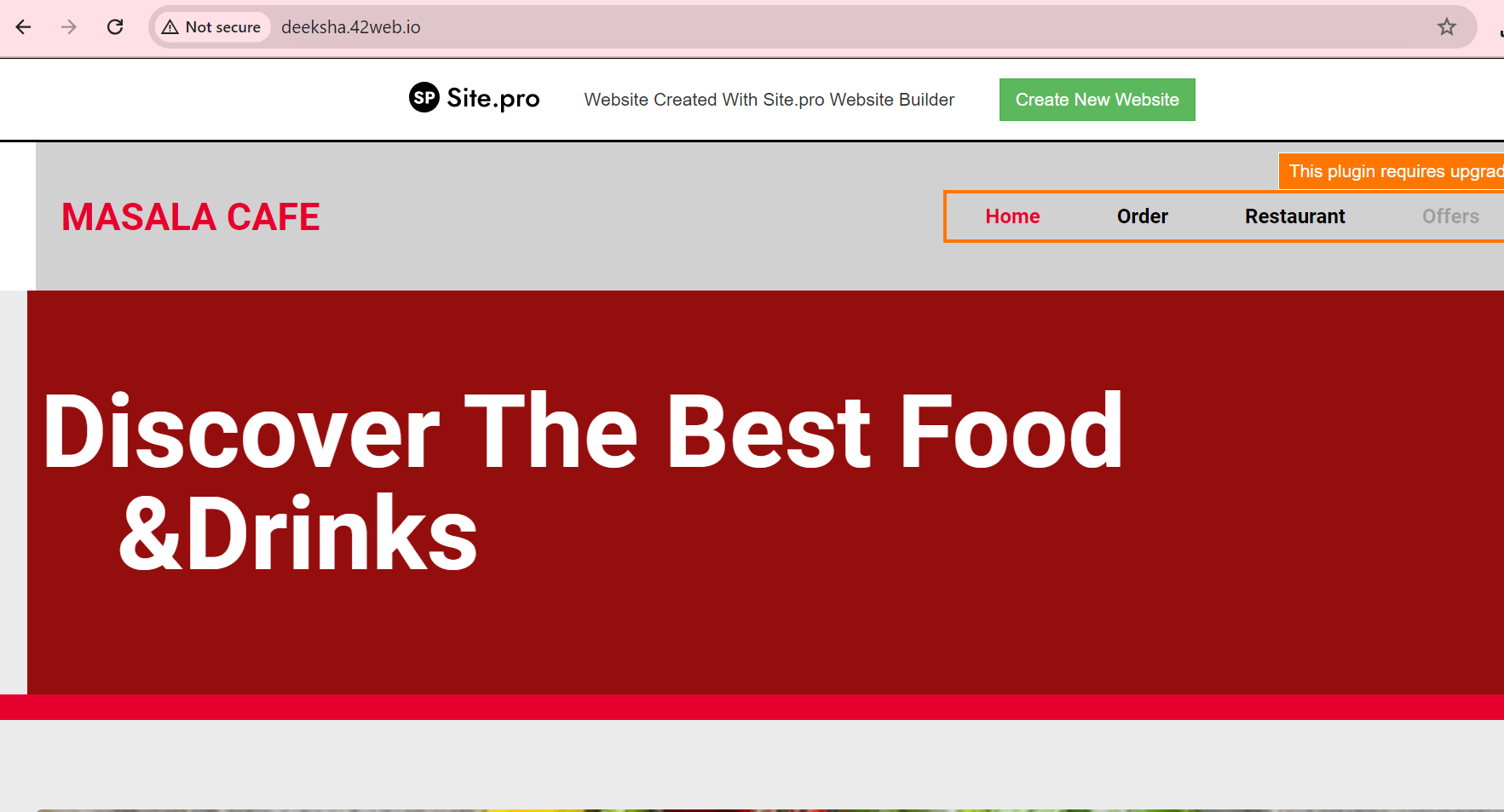
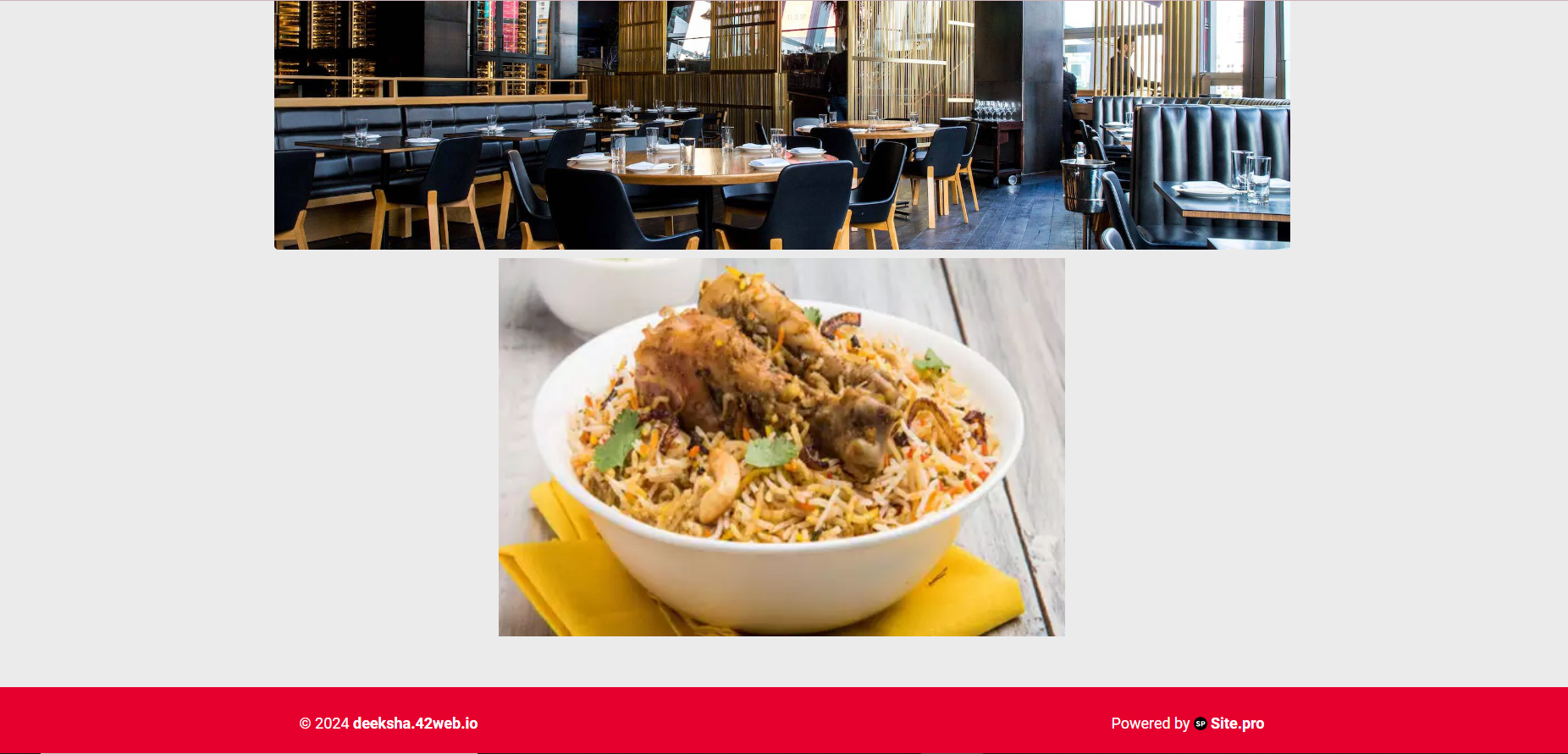










**RESULT:**

Thus, a wordpress personal website using infinityfree webhosting and SSL certification integration is created and implemented successfully.

**Date:**

**Exercise - 3**

**CREATE A BLOG WITH YOUR CHOSEN TITLE AND INCORPORATE ALL ESSENTIAL FEATURES ON IT**

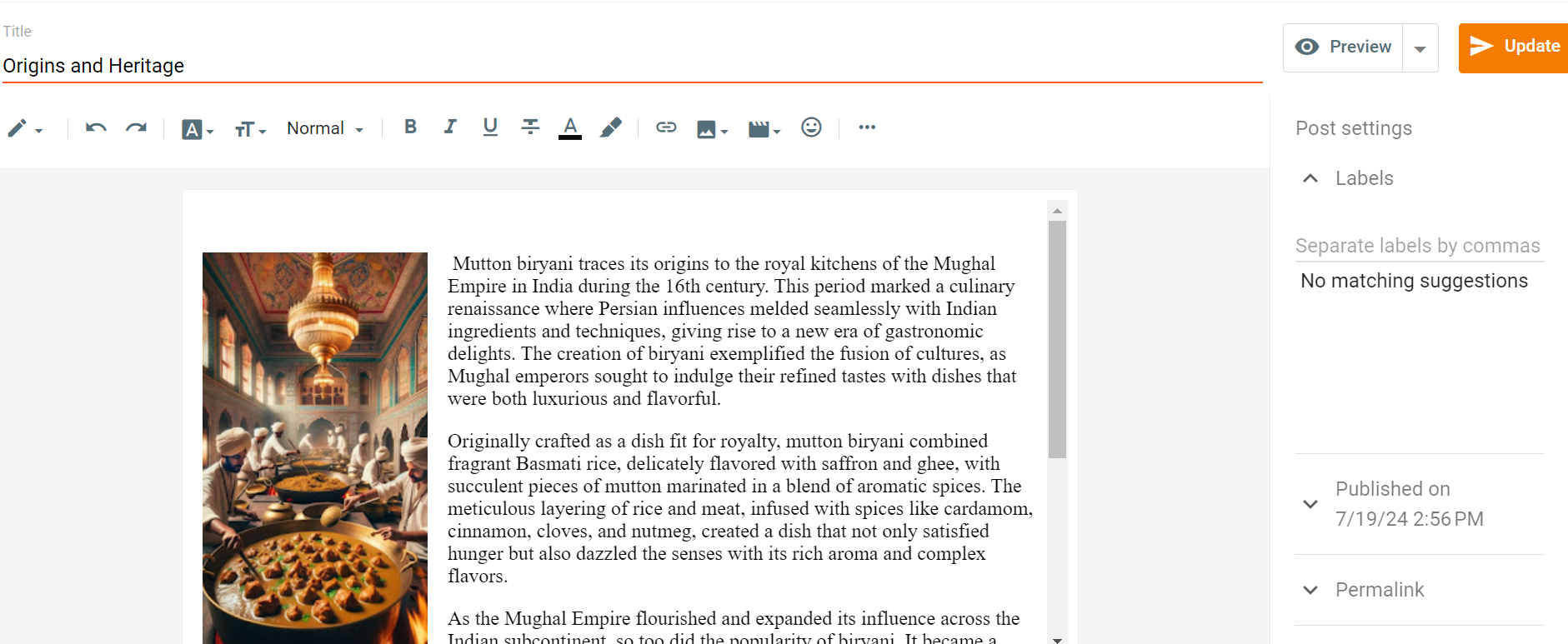
**AIM:**

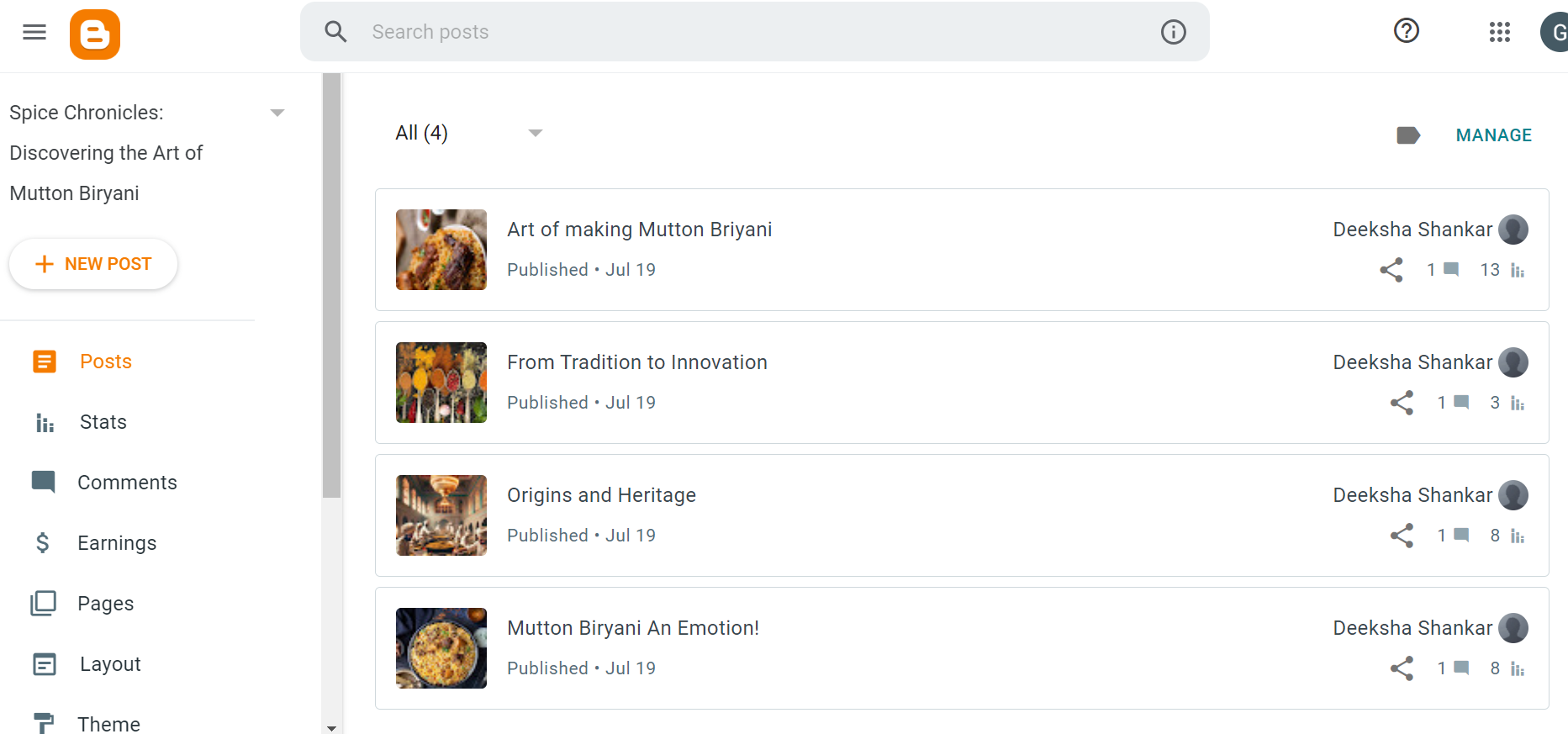
To create a blog with your chosen title and incorporate all essential features on it.

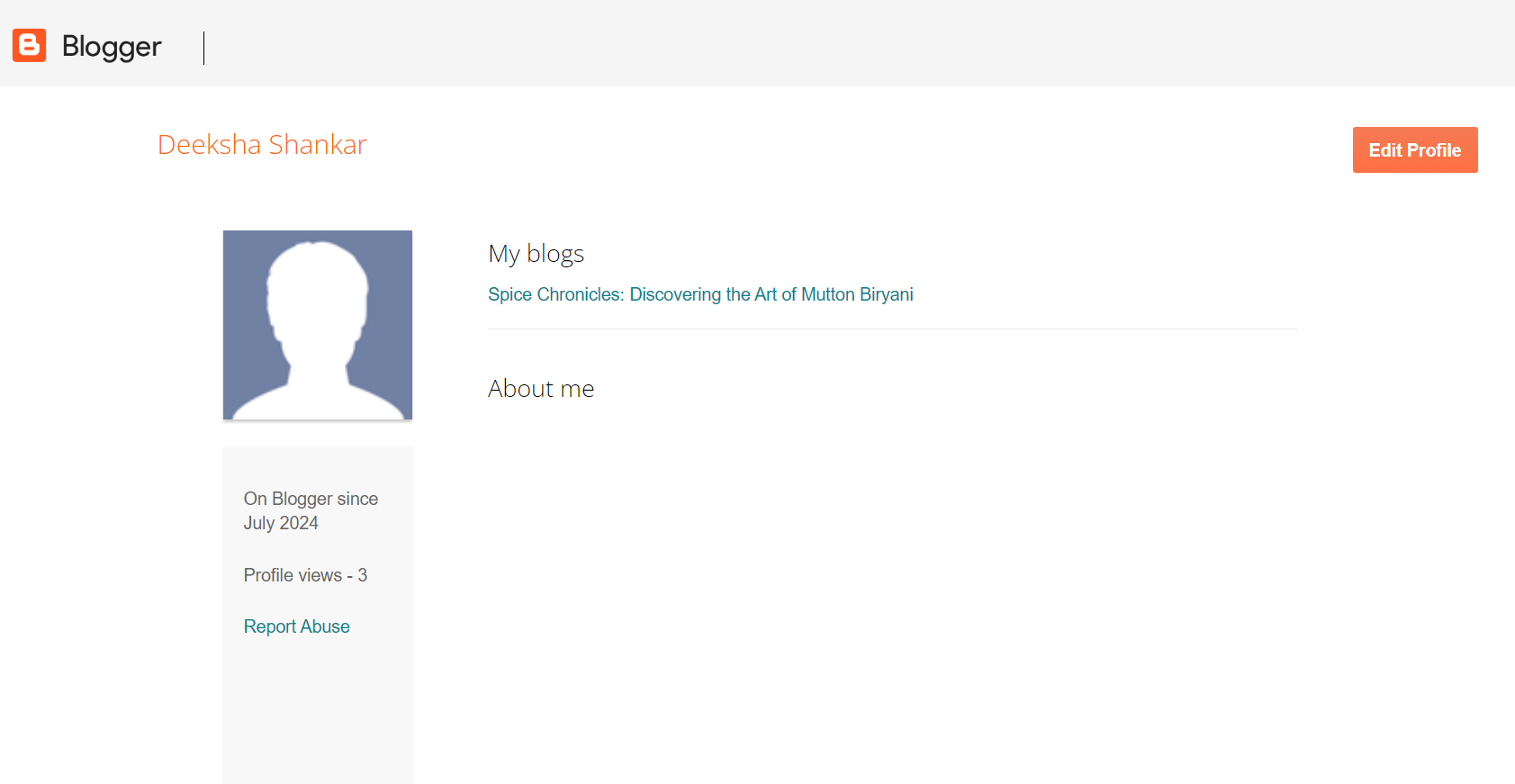
**PROBLEM DESCRIPTION:**

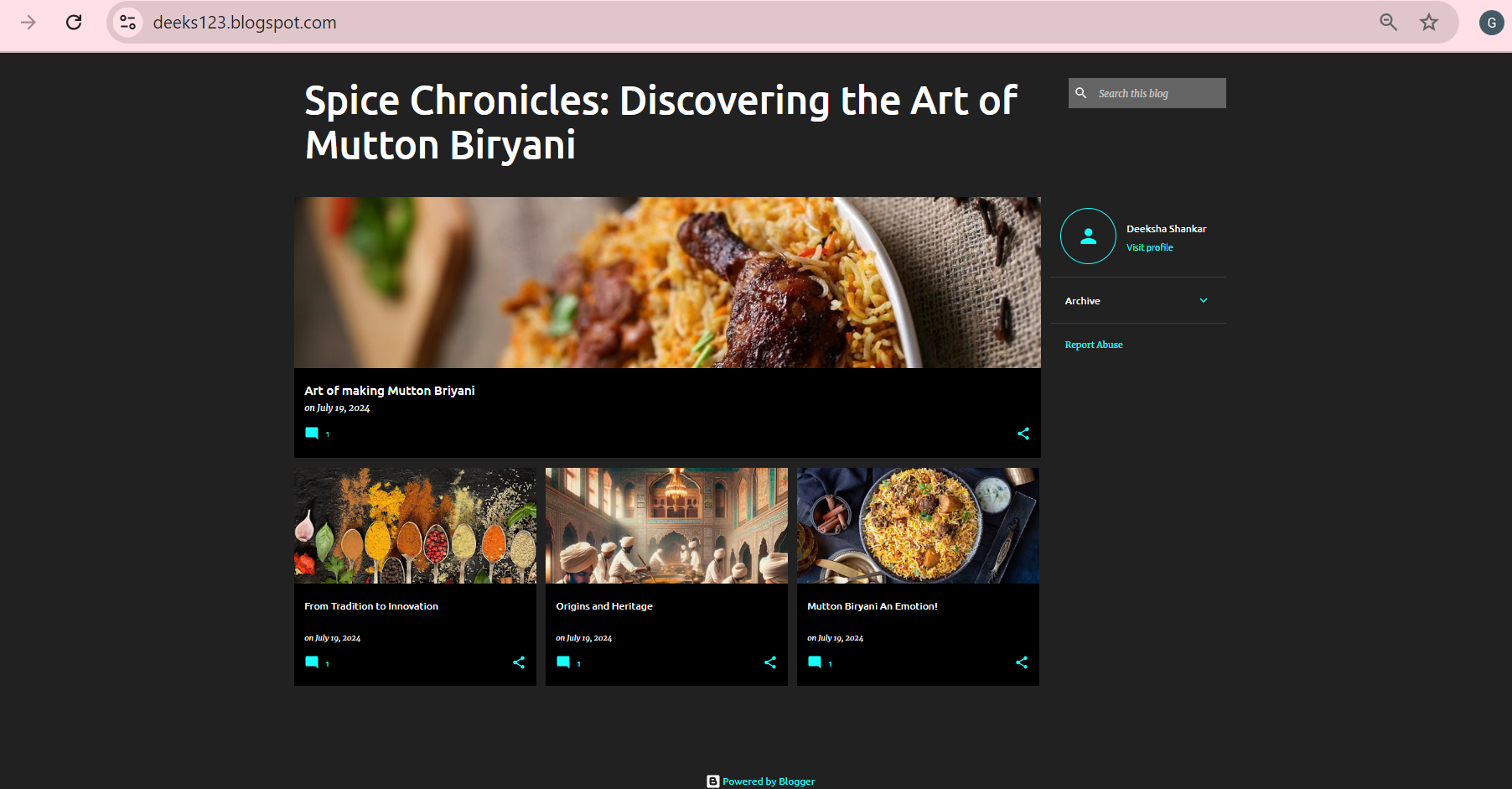
The problem involves creating a blog on Blogger.com, focusing on customizing the blog's appearance, creating essential pages, and incorporating features like a search bar, social media integration, and a commenting system. The project includes writing and publishing at least 10 engaging blog posts, integrating multimedia elements, and sharing the blog to attract readers and gather comments.

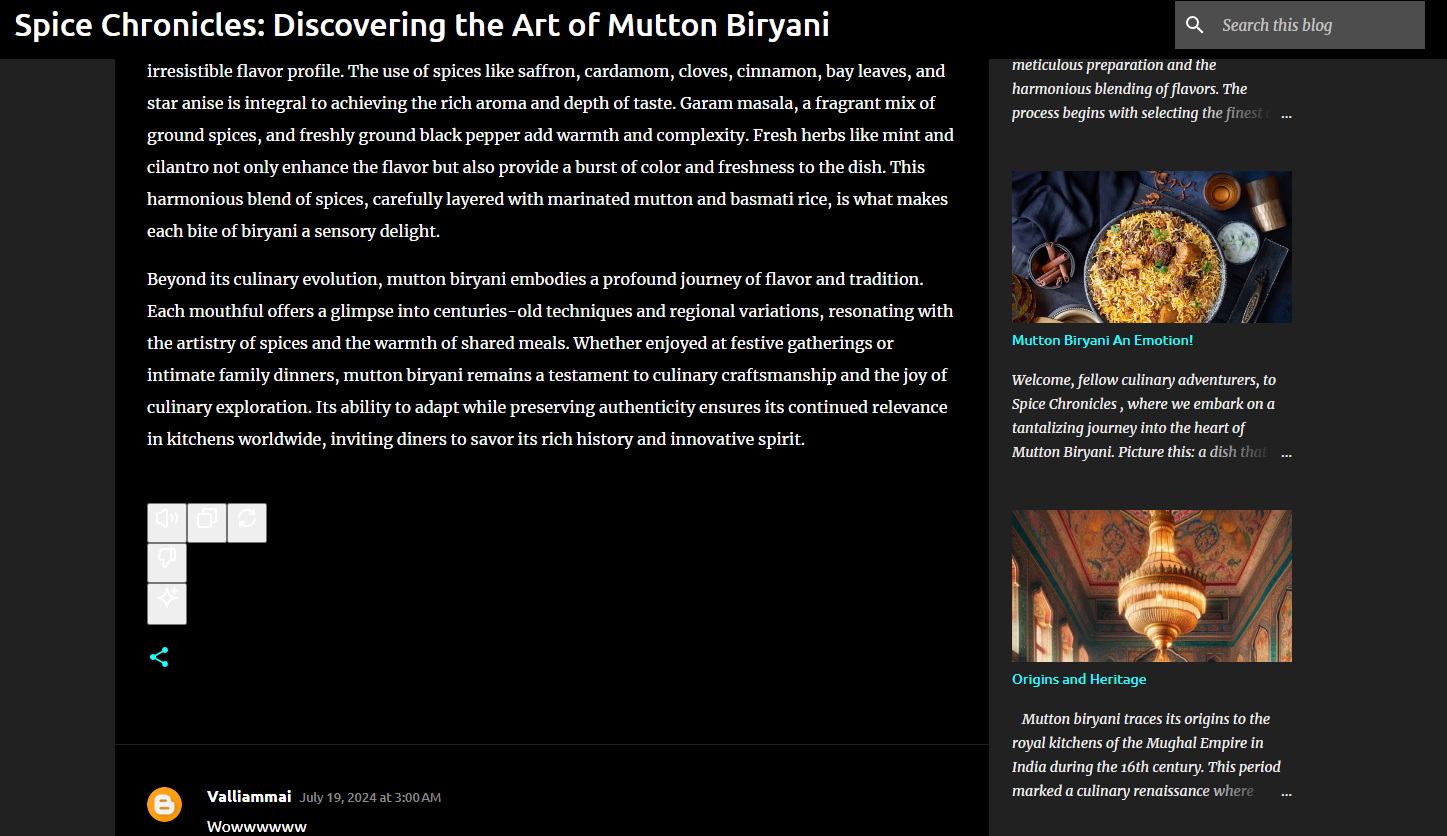
**OUTPUT:**

****

****

****

****

****

**RESULT:**

Thus, a blog is created with chosen title and all essential features are incorporated on it and implemented successfully.

**Date:**

**Exercise - 4**

**ENABLE GOOGLE ANALYTICS FEATURES FOR YOUR EXISTING BLOG AND GENERATE VARIOUS REPORTS.**

**AIM:**

To enable Google analytics feature on our existing blogs and generate various reports such as audience overview reports, behaviour report and acquisition report.

**PROBLEM DESCRIPTION:**

To perform Google analytics on the created blogs (using Blogger.com) and generate reports such as audience overview reports, behaviour report and acquisition report. The User acquisition report is a detail report about how new users find your website or app for the first time. The audience overview report instantly analyse the target audience to know users interests, intentions and behaviour and check what differentiate them from general Internet user. The Behaviour Overview report focuses on individual pages of our website, opposed to the entire site.

**OUTPUT:**

Interpretations**:**

By analysing various reports generated the following questions are answered.

a)Which geographic locations are driving the most traffic to your blog?

India, TamilNadu, Madurai, Thiruparankundram.

b)What are the primary sources of traffic to your blog?

Web, Chrome.

c)Which pages on your blog are most popular among visitors?

The linear regression blog is popular among visitors

d)How much time, on average, do visitors spend on your blog?

6 minutes and 52 seconds

e)What is the overall bounce rate of your blog?

Overall bounce rate is 35%

**RESULT:**

Thus Google analytics features are enabled to existing blogs and various reports are generated.

**Date:**

**Exercise - 5**

**CREATING SEO-OPTIMIZED CONTENT FOR YOUR BLOGGER POST**

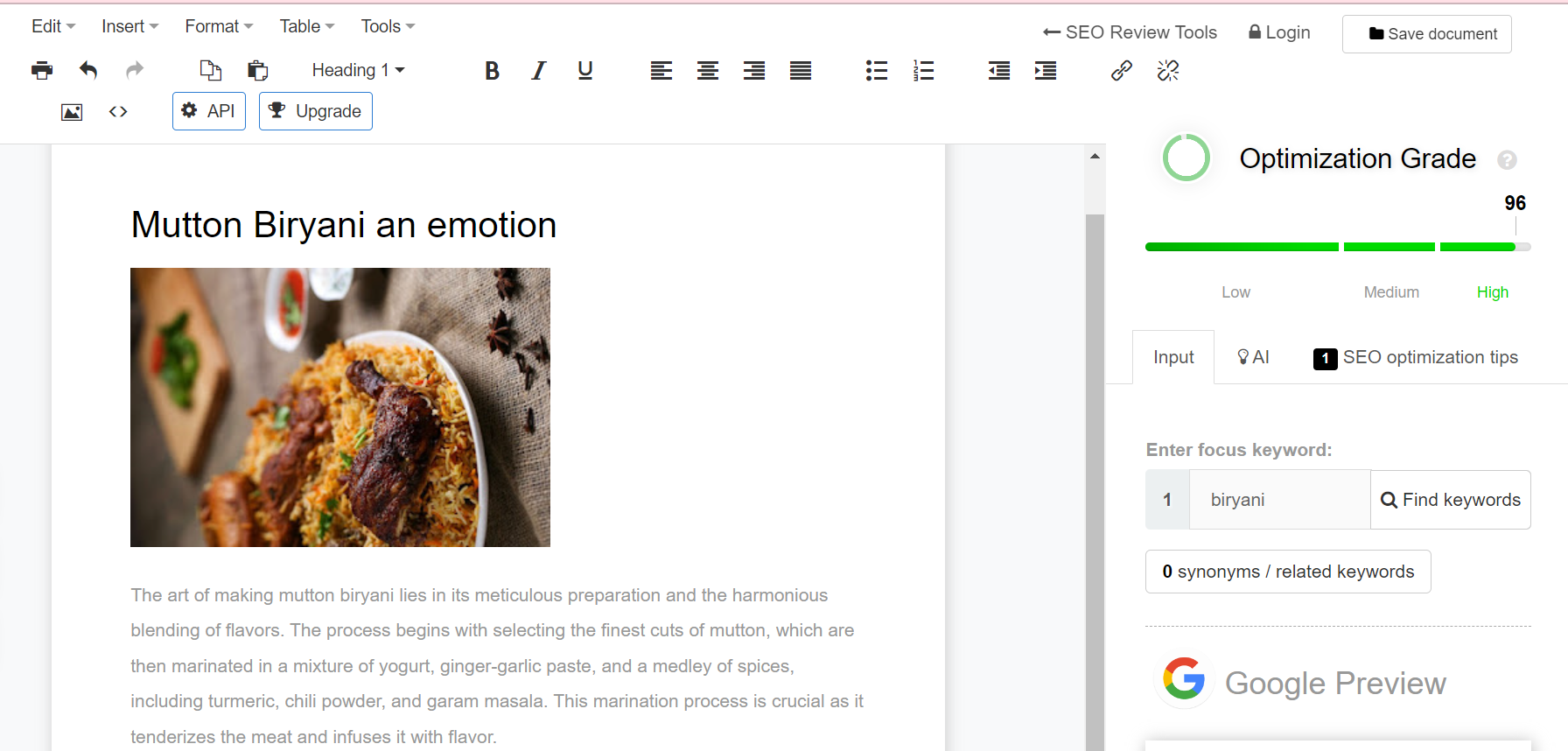
**AIM:**

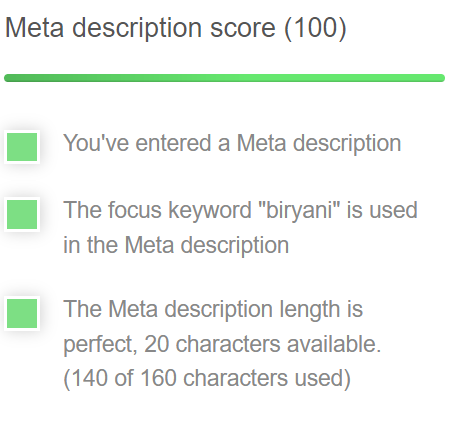
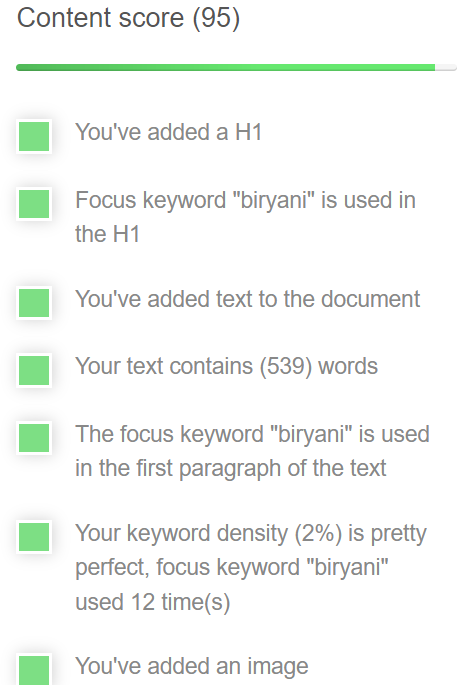
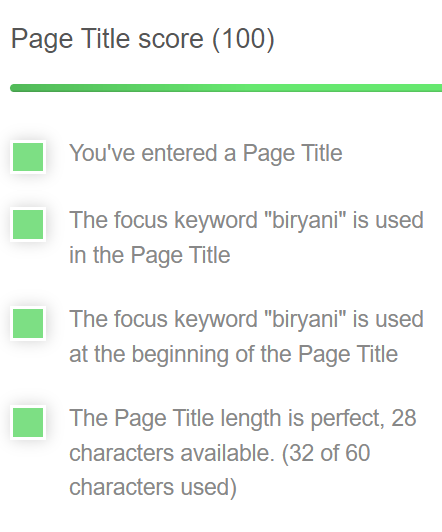
To create SEO-optimized content for the blogger post using SEO-content Editor Tool.

**PROBLEM DESCRIPTION:**

To implement content analysis, first access the content analysis tool using SEO-review tool for analysis. Then, copy the content from the blogger content that includes title, description, headings and headings into the tool. Start the analysis and optimization techniques.

**OUTPUT:**

****

**** ****

Interpretations:

Primary keyword: Biryani

Title tag: The art of making mutton biryani

Meta description: Mutton biryani isn't just a meal; it's a celebration, the spices are well balanced, making mutton biryani a true masterpiece of culinary art.

Here, the overall score was 96% with content score as 95%, meta description score as 100% and a title score as 100%.

**RESULT:**

Thus, implementation of SEO optimized content for the blog with 96% using SEO- ONTENT editor tool was implemented and verified.