

Tech Job Portal Project

A Web Application to Browse and Download Tech Job Listings

By Manisha P

Team 2

Cybernaut Intern

ABSTRACT :

The demand for technology talent has surged, creating a need for efficient tools to search, analyze, and track tech job openings. The **Tech Job Portal** is a Python-based solution designed to scrape and display job listings from online sources such as Indeed. The system extracts key job information including title, company, location, type, and application link. Users can search jobs, filter by type (Full-time, Internship, Remote), view details in a responsive web interface, and download listings as CSV for offline use. Built with Flask, BeautifulSoup, and Pandas, it provides a scalable, automated solution for job seekers and researchers.

PROJECT DESCRIPTION :

The **Tech Job Portal** is a Python-based web application that automatically collects tech job information for analysis and browsing. It supports:

- **Job Search & Extraction:** Title, Company, Location, Type, Link
- **Filtering & Sorting:** By job type
- **Responsive Browsing:** Desktop and mobile-friendly UI
- **CSV Download:** Export listings for offline use
- **Dark Mode:** Improved readability
- **Automation:** Scraping data from multiple pages using Python

The portal is designed for job seekers, recruiters, and market researchers who need insights into employment trends and company hiring activities.

INTRODUCTION :

The tech industry is highly dynamic, with thousands of job openings posted daily across multiple platforms. Manually tracking these jobs is inefficient and error-prone. Automated web scraping and data presentation solutions provide:

- Real-time access to job listings
- Efficient browsing of multiple postings
- CSV export for offline analysis
- Insights into company hiring trends

The Tech Job Portal addresses these needs by using Python, Flask, and BeautifulSoup to extract structured job data and display it in a user-friendly interface.

EXISTING METHODS :

Current approaches to accessing job listings include:

1. **Manual Search** – Time-consuming and prone to errors
2. **Job Portal APIs** – Limited by API keys, request quotas, and access restrictions
3. **Third-Party Job Tools** – May require subscriptions and have limited customization

Limitations :

- Restricted access to large-scale data
- Lack of control over filtering and display
- Limited offline export options

PROPOSED SOLUTION :

The **Tech Job Portal** overcomes these limitations by scraping job data directly from Indeed using Python. It allows:

- Real-time access to tech job listings
- Customizable search and filtering

- Downloading results as CSV
- Responsive UI with light/dark mode

KEY FEATURES :

1. **Job Search & Extraction** – Title, company, location, type, application link
2. **Filtering & Sorting** – Filter by Full-time, Internship, or Remote
3. **Pagination Handling** – Scrape multiple pages automatically
4. **CSV Download** – Save results for offline analysis
5. **Responsive UI** – Works on desktop, tablet, and mobile
6. **Dark Mode** – Toggle between light and dark themes
7. **Automation** – Run scraping scripts periodically

Advantages :

- Fully automated scraping
- No API restrictions
- Flexible export options

TECHNOLOGIES USED :

1. **Language:** Python
2. **Libraries:**
 - BeautifulSoup – HTML parsing
 - Requests – Fetch web pages
 - Pandas – Data storage and manipulation
 - Flask – Web application framework
3. **Frontend:** HTML, CSS, Bootstrap 5, Bootstrap Icons
4. **Data Storage:** CSV
5. **Browser Automation (Optional):** Selenium for dynamic content

METHODS :

1. **Initialization** – Import libraries and read CSV
2. **Access Job Portal** – Send requests to Indeed
3. **Data Extraction** – Parse job title, company, location, type, link
4. **Pagination Handling** – Scrape multiple result pages
5. **Data Processing** – Store in Pandas DataFrame
6. **Export Results** – Save as CSV
7. **Web Display** – Serve data via Flask
8. **Dark Mode Toggle** – Frontend enhancement

IMPLEMENTATION / FOLDER STRUCTURE :

- app.py – Flask application
- scraper.py – Scraping script
- jobs.csv – Stored job data
- templates/index.html – Web interface
- static/style.css – Styling and dark mode
- static/default-logo.png – Placeholder company logo

app.py :

```
from flask import Flask, render_template, send_file
import pandas as pd
import os

app = Flask(__name__, template_folder="templates", static_folder="static")

@app.route("/")
def home():
```

```

csv_file = "jobs.csv"
jobs = []
if os.path.exists(csv_file):
    jobs = pd.read_csv(csv_file).to_dict(orient="records")
return render_template("index.html", jobs=jobs)

@app.route("/download")
def download_csv():
    return send_file("jobs.csv", as_attachment=True)

if __name__ == "__main__":
    app.run(debug=True)

```

scraper.py :

```

import requests
from bs4 import BeautifulSoup
import pandas as pd

def scrape_indeed(query="python developer", location="India", pages=2):
    base_url = "https://www.indeed.com/jobs"
    jobs = []

    for page in range(pages):
        params = {"q": query, "l": location, "start": page*10}
        response = requests.get(base_url, params=params, headers={"User-Agent": "Mozilla/5.0"})
        soup = BeautifulSoup(response.text, "html.parser")

        for job_card in soup.find_all("div", class_="job_seen_beacon"):
            title = job_card.find("h2", class_="jobTitle")

```

```

company = job_card.find("span", class_ ="companyName")
location = job_card.find("div", class_ ="companyLocation")
link = job_card.find("a", href=True)

jobs.append({
    "title": title.text.strip() if title else None,
    "company": company.text.strip() if company else None,
    "location": location.text.strip() if location else None,
    "link": "https://www.indeed.com" + link["href"] if link else None
})

df = pd.DataFrame(jobs)
df.to_csv("jobs.csv", index=False)
print(" ✅ Jobs scraped and saved to jobs.csv")
return df

if __name__ == "__main__":
    scrape_indeed()

```

jobs.csv :

title,company,location,link			
Python Developer,TechSoft,"Bangalore, India",https://www.indeed.com/q-python-developer-jobs.html			
Data Analyst,DataWorks,"Hyderabad, India",https://indeed.com/job/data-analyst-2			
Full Stack Engineer,InnovateX,"Chennai, India",https://indeed.com/job/fullstack-engineer-3			
Machine Learning Engineer,AI Labs,"Pune, India",https://indeed.com/job/ml-engineer-4			
Backend Developer,CloudNet,"Delhi, India",https://indeed.com/job/backend-developer-5			
Frontend Developer,WebWorks,"Mumbai, India",https://indeed.com/job/frontend-developer-6			
DevOps Engineer,SysOps Ltd,"Noida, India",https://indeed.com/job/devops-engineer-7			

AI Researcher,DeepAI,"Bangalore, India",<https://indeed.com/job/ai-researcher-8>
Cloud Architect,CloudMasters,"Hyderabad, India",<https://indeed.com/job/cloud-architect-9>
Database Administrator,DataSecure,"Chennai, India",<https://indeed.com/job/dba-10>
Cybersecurity Analyst,SafeNet,"Pune, India",<https://indeed.com/job/cybersecurity-analyst-11>
Software Engineer,TechGlobal,"Delhi, India",<https://indeed.com/job/software-engineer-12>
Big Data Engineer,DataLake,"Bangalore, India",<https://indeed.com/job/big-data-engineer-13>
Business Analyst,BizSolutions,"Hyderabad, India",<https://indeed.com/job/business-analyst-14>
QA Engineer,QualityTech,"Mumbai, India",<https://indeed.com/job/qa-engineer-15>
Mobile App Developer,AppDev,"Chennai, India",<https://indeed.com/job/mobile-app-developer-16>
Data Scientist,DataVision,"Pune, India",<https://indeed.com/job/data-scientist-17>
System Administrator,NetSys,"Delhi, India",<https://indeed.com/job/system-admin-18>
Cloud Engineer,CloudNext,"Bangalore, India",<https://indeed.com/job/cloud-engineer-19>
AI Product Manager,SmartAI,"Hyderabad, India",<https://indeed.com/job/ai-product-manager-20>
Tech Lead,LeadTech,"Chennai, India",<https://indeed.com/job/tech-lead-21>
Solutions Architect,Innovarch,"Pune, India",<https://indeed.com/job/solutions-architect-22>
UI/UX Designer,DesignHub,"Mumbai, India",<https://indeed.com/job/ui-ux-designer-23>
Game Developer,PlayWorks,"Bangalore, India",<https://indeed.com/job/game-developer-24>
IT Support Engineer,HelpDeskPro,"Delhi, India",<https://indeed.com/job/it-support-engineer-25>

index.html :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tech Job Portal</title>
```

```

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">

<link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.5/font/bootstrap-icons.css"
rel="stylesheet">

<link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">

</head>

<body>

<!-- Navbar -->

<nav class="navbar navbar-expand-lg navbar-dark bg-primary sticky-top shadow">
  <div class="container-fluid">
    <a class="navbar-brand fw-bold" href="/">Tech Job Portal</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNav">
      <ul class="navbar-nav ms-auto align-items-center">
        <li class="nav-item me-2">
          <a class="nav-link" href="/download"><i class="bi bi-download"></i>
          Download CSV</a>
        </li>
        <li class="nav-item me-2">
          <a class="nav-link" href="/login"><i class="bi bi-person-circle"></i>
          Login</a>
        </li>
        <!-- Dark Mode Toggle -->
        <li class="nav-item">
          <button id="darkModeToggle" class="btn btn-light btn-sm">
            <i class="bi bi-moon"></i>
          </button>
        </li>
      </ul>
    </div>
  </div>
</nav>

```

```

        </ul>
    </div>
</div>
</nav>

<!-- Hero Section -->
<div class="text-center my-5 p-5 bg-light rounded shadow-sm hero">
    <h1 class="fw-bold text-primary">Find Your Dream Tech Job</h1>
    <p class="lead">Browse thousands of tech jobs from top companies.</p>
</div>

<div class="container">

    <!-- Search Bar -->
    <form class="d-flex mb-4 p-3 rounded shadow-sm search-bar" method="GET"
action="/">
        <input class="form-control me-2" type="search" name="q" placeholder="Search
jobs..." aria-label="Search">
        <select class="form-select me-2" name="type">
            <option value="">All Types</option>
            <option value="Full-time">Full-time</option>
            <option value="Internship">Internship</option>
            <option value="Remote">Remote</option>
        </select>
        <button class="btn btn-outline-light btn-primary" type="submit">Search</button>
    </form>

    <!-- Job Listings -->
    {% if jobs %}
    <div class="row">
        {% for job in jobs %}

```

```

<div class="col-md-4">
    <div class="card shadow-sm mb-4 border-primary h-100 job-card">
        <div class="card-body text-center">
            <!-- Company Logo -->
            {% if job.logo %}
                
            {% else %}
                
            {% endif %}
            <h5 class="card-title text-primary">{{ job.title }}</h5>
            <p class="card-text">
                <span class="badge bg-success mb-2">{{ job.type or 'Full-time' }}</span>
                <br>
                <strong>Company:</strong> {{ job.company }} <br>
                <strong>Location:</strong> {{ job.location }} <br>
                <strong>Posted:</strong> {{ job.date_posted or 'N/A' }}</p>
            <a href="{{ job.link }}" target="_blank" class="btn btn-sm btn-success w-100">Apply Now</a>
        </div>
    </div>
</div>
{% endfor %}

<% else %>
    <p class="text-muted text-center">No jobs found. Please run <code>scraper.py</code> to fetch job listings.</p>
    {% endif %}</div>

```

```

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

<!-- Dark Mode Script --&gt;
&lt;script&gt;

const toggleBtn = document.getElementById('darkModeToggle');

toggleBtn.addEventListener('click', () =&gt; {

    document.body.classList.toggle('dark-mode');

    if (document.body.classList.contains('dark-mode')) {

        toggleBtn.innerHTML = '&lt;i class="bi bi-sun"&gt;&lt;/i&gt;';

        toggleBtn.classList.remove('btn-light');

        toggleBtn.classList.add('btn-dark');

    } else {

        toggleBtn.innerHTML = '&lt;i class="bi bi-moon"&gt;&lt;/i&gt;';

        toggleBtn.classList.remove('btn-dark');

        toggleBtn.classList.add('btn-light');

    }

});

&lt;/script&gt;

&lt;/body&gt;

&lt;/html&gt;
</pre>

```

Style.css

```

/* ----- Body ----- */

body {

background-color: #f0f2f5;

font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

margin: 0;

padding: 0;

transition: background-color 0.3s, color 0.3s;

```

```
}

/* ----- Navbar ----- */
.navbar {
    box-shadow: 0 2px 8px rgba(0,0,0,0.1);
}

/* ----- Hero Section ----- */
.hero {
    background: linear-gradient(135deg, #d9e7ff, #ffffff);
    padding: 4rem 2rem;
    border-radius: 12px;
    text-align: center;
    margin-bottom: 2rem;
    box-shadow: 0 4px 15px rgba(0,0,0,0.1);
    transition: background 0.3s;
}

/* ----- Search Bar ----- */
.search-bar {
    background: rgba(255, 255, 255, 0.4);
    backdrop-filter: blur(10px);
    border: 1px solid rgba(0,0,0,0.1);
    border-radius: 12px;
    padding: 1rem;
    display: flex;
    flex-wrap: wrap;
    gap: 0.5rem;
}
```

```
/* ----- Job Cards ----- */
```

```
.card {  
    border-radius: 12px;  
    transition: all 0.3s ease-in-out;  
    box-shadow: 0 2px 8px rgba(0,0,0,0.1);  
    opacity: 0;  
    transform: translateY(20px);  
    animation: fadeIn 0.5s forwards;  
    height: 100%;  
}  
  
/* Card Hover */
```

```
.card:hover {  
    transform: translateY(-5px);  
    box-shadow: 0 8px 20px rgba(0,0,0,0.2);  
}
```

```
/* ----- Company Logo ----- */
```

```
.company-logo {  
    width: 60px;  
    height: 60px;  
    object-fit: contain;  
    border-radius: 8px;  
    background: #fff;  
    padding: 5px;  
    box-shadow: 0 2px 6px rgba(0,0,0,0.1);  
}  
  
/* ----- Fade-in Animation ----- */
```

```
@keyframes fadeIn {
```

```
        to { opacity: 1; transform: translateY(0); }

    }

/* ----- Badges ----- */

.badge {
    font-size: 0.75rem;
}

/* ----- Buttons ----- */

.btn-primary {
    background-color: #0d6efd;
    border-color: #0d6efd;
    transition: background-color 0.2s;
}

.btn-primary:hover {
    background-color: #0b5ed7;
}

/* ----- Dark Mode ----- */

.dark-mode {
    background-color: #181a1b;
    color: #e4e6eb;
}

.dark-mode .navbar {
    background-color: #242526 !important;
}

.dark-mode .hero {
```

```
background: linear-gradient(135deg, #1e2022, #2a2c2f);  
}  
  
.dark-mode .card {  
background-color: #242526;  
color: #e4e6eb;  
border: 1px solid #333;  
}  
  
.dark-mode .search-bar {  
background: rgba(36, 37, 38, 0.8);  
border: 1px solid #333;  
}  
  
.dark-mode .btn-primary {  
background-color: #3a3b3c;  
border-color: #3a3b3c;  
}  
  
.dark-mode .btn-primary:hover {  
background-color: #555;  
}  
  
/* ----- Responsive ----- */  
 @media (max-width: 768px) {  
 .search-bar {  
 flex-direction: column;  
 }  
 }
```

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface with the following details:

- File Explorer (Left):** Shows the project structure under "OPEN EDITORS".
 - TECH_JOB_PORTAL:** Contains "snapshots", "Templates", "index.html", "style.css", and "app.py".
 - jobs (3).csv**: A CSV file with 3 columns.
 - index.html**: A template file.
 - style.css**: A CSS file.
 - scraper.py**: A Python script.
- Code Editor (Center):** Displays the content of "app.py".

```
1  from flask import Flask, render_template, send_file
2  import pandas as pd
3  import os
4
5  app = Flask(__name__, template_folder="templates", static_folder="static")
6
7  @app.route("/")
8  def home():
9      csv_file = "jobs.csv"
10     jobs = []
11     if os.path.exists(csv_file):
12         jobs = pd.read_csv(csv_file).to_dict(orient="records")
13     return render_template("index.html", jobs=jobs)
14
15  @app.route("/download")
16  def download_csv():
17      return send_file("jobs.csv", as_attachment=True)
18
19 if __name__ == "__main__":
20     app.run(debug=True)
21
22
```
- Terminal (Top):** Shows the command "Tech.Job_Portal".
- AI Assistant (Bottom Right):** A floating window from "BLACKBOX" with the title "Build with agent mode." It includes a "Share" icon, a note about AI responses being inaccurate, and instructions to onboard AI onto the codebase.
- System Tray (Bottom):** Shows icons for battery, signal, and date/time (03-10-2025).

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface with the following details:

- File Structure (EXPLORER):** The left sidebar shows a tree view of files and folders. Under "OPEN EDITORS", there are "scraper.py", "app.py", "jobs (3).csv", "index.html", and "style.css". Under "TECH_JOB_PORTAL", there are "index.html", "style.css", "app.py", "jobs.csv", and "scraper.py".
- Code Editor:** The main editor area displays the "scraper.py" file. The code uses BeautifulSoup to parse Indeed job listings and save them to a CSV file. It includes imports for requests, BeautifulSoup, pandas, and os. It defines a function "scrape_indeed" that takes a query, location, and number of pages as parameters, then iterates through the pages to extract job titles, companies, locations, and links.
- Bottom Status Bar:** Shows the current file is "scraper.py", line 6, column 45. It also displays icons for connection status, Python environment, user sign-in, and system information like temperature and battery level.
- Bottom Taskbar:** Includes icons for search, file operations, terminal, and workspace navigation.
- Right Sidebar:** Features a "CHAT" section with a message "Build with agent mode.", a note about AI responses being inaccurate, and instructions to onboard AI. It also has buttons for "Add Context...", "Agent", and "BLACKBOX Autocomplete: ON".

```

title,company,location,link
1 Python Developer,techSoft,"Bangalore, India",https://www.indeed.com/q-python-developer-jobs.html
2 Data Analyst,datworks,"Hyderabad, India",https://www.indeed.com/q-data-analyst-jobs.html
3 Full Stack Engineer,innoVATE,"Chennai, India",https://www.indeed.com/q-full-stack-developer-jobs.html
4 Machine Learning Engineer,AT Labs,"Pune, India",https://www.indeed.com/q-machine-learning-engineer-jobs.html
5 Backend Developer,CloudNet,"Delhi, India",https://www.indeed.com/q-backend-developer-jobs.html
6 Frontend Developer,webWorks,"Mumbai, India",https://www.indeed.com/q-frontend-developer-jobs.html
7 DevOps Engineer,Sysops Ltd,"Noida, India",https://www.indeed.com/q-devops-engineer-jobs.html
8 AI Researcher,DeepAI,"Bangalore, India",https://www.indeed.com/q-ai-researcher-jobs.html
9 Cloud Architect,CloudMasters,"Hyderabad, India",https://www.indeed.com/q-cloud-architect-jobs.html
10 Database Administrator,DataSecure,"Chennai, India",https://www.indeed.com/q-database-administrator-jobs.html
11 Cybersecurity Analyst,SafeNet,"Pune, India",https://www.indeed.com/q-cybersecurity-analyst-jobs.html
12 Software Engineer,TechGlobal,"Delhi, India",https://www.indeed.com/q-software-engineer-jobs.html
13 Big Data Engineer,DataLake,"Bangalore, India",https://www.indeed.com/q-big-data-engineer-jobs.html
14 Business Analyst,nisolutions,"Hyderabad, India",https://www.indeed.com/q-business-analyst-jobs.html
15 QA Engineer,QualityTech,"Mumbai, India",https://www.indeed.com/q-qa-engineer-jobs.html
16 Mobile App Developer,AppDev,"Chennai, India",https://www.indeed.com/q-mobile-app-developer-jobs.html
17 Data Scientist,DataVision,"Pune, India",https://www.indeed.com/q-data-scientist-jobs.html
18 System Administrator,Netsys,"Delhi, India",https://www.indeed.com/q-system-administrator-jobs.html
19 Cloud Engineer,CloudNext,"Bangalore, India",https://www.indeed.com/q-cloud-engineer-jobs.html
20 AI Product Manager,SmartAI,"Hyderabad, India",https://www.indeed.com/q-ai-product-manager-jobs.html
21 Tech Lead,LeadTech,"Chennai, India",https://www.indeed.com/q-tech-lead-jobs.html
22 Solutions Architect,innoVarch,"Pune, India",https://www.indeed.com/q-solutions-architect-jobs.html
23 UI/UX Designer,DesignHub,"Mumbai, India",https://www.indeed.com/q-ui-ux-designer-jobs.html
24 Game Developer,Playworks,"Bangalore, India",https://www.indeed.com/q-game-developer-jobs.html
25 IT Support Engineer,HelpDeskPro,"Delhi, India",https://www.indeed.com/q-it-support-engineer-jobs.html
26
27

```

```

<!DOCTYPE html>

    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>Tech Job Portal</title>
        <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
        <link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.5/font/bootstrap-icons.css" rel="stylesheet">
        <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}>
    </head>
    <body>
        <!-- Navbar -->
        <nav class="navbar navbar-expand-lg navbar-dark bg-primary sticky-top shadow">
            <div class="container-fluid">
                <a class="navbar-brand fw-bold" href="/">Tech Job Portal</a>
                <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav">
                    <span class="navbar-toggler-icon"></span>
                </button>
                <div class="collapse navbar-collapse" id="navbarNav">
                    <ul class="navbar-nav ms-auto align-items-center">
                        <li class="nav-item me-2">
                            <a class="nav-link" href="/download"><i class="bi bi-download"></i> Download CSV</a>
                        </li>
                        <li class="nav-item me-2">
                            <a class="nav-link" href="/login"><i class="bi bi-person-circle"></i> Login</a>
                        </li>
                    <!-- Dark Mode Toggle -->
                    <li class="nav-item">
                        <button id="darkModeToggle" class="btn btn-light btn-sm">
                            <i class="bi bi-moon"></i>
                        </button>
                    </li>
                </ul>
            </div>
        </nav>
    </body>

```

```

body {
    background-color: #f0f2f5;
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    margin: 0;
    padding: 0;
    transition: background-color 0.3s, color 0.3s;
}

.navbar {
    box-shadow: 0 2px 8px rgba(0,0,0,0.1);
}

.hero {
    background: linear-gradient(135deg, #d9e7ff, #ffffff);
    padding: 4rem;
    border-radius: 12px;
    text-align: center;
    margin-bottom: 2rem;
    box-shadow: 0 4px 15px rgba(0,0,0,0.1);
    transition: background 0.3s;
}

.search-bar {
    background: #d9e7ff;
    backdrop-filter: blur(10px);
    border: 1px solid #d9e7ff;
    border-radius: 12px;
    padding: 1rem;
    display: flex;
    flex-wrap: wrap;
    gap: 0.5rem;
}

```

OUTPUT :

- **CSV File containing:**
 - Job Title
 - Company
 - Location
 - Type
 - Application Link
- **Web Interface** displaying job cards with:
 - Company Logo
 - Job Title
 - Location & Type
 - Apply Now button

```

1  /* ----- Body ----- */
2  body {
3      background-color: #f0f2f5;
4      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
5      margin: 0;
6      padding: 0;
7      transition: background-color 0.3s, color 0.3s;
8  }
9
10 /* ----- Navbar ----- */
11 .navbar {
12     box-shadow: 0 2px 8px rgba(0,0,0,0.1);
13 }
14
15 /* ----- Hero Section ----- */
16 .hero {
17     background: linear-gradient(to bottom, #f0f2f5, #e0e0e0);
18 }

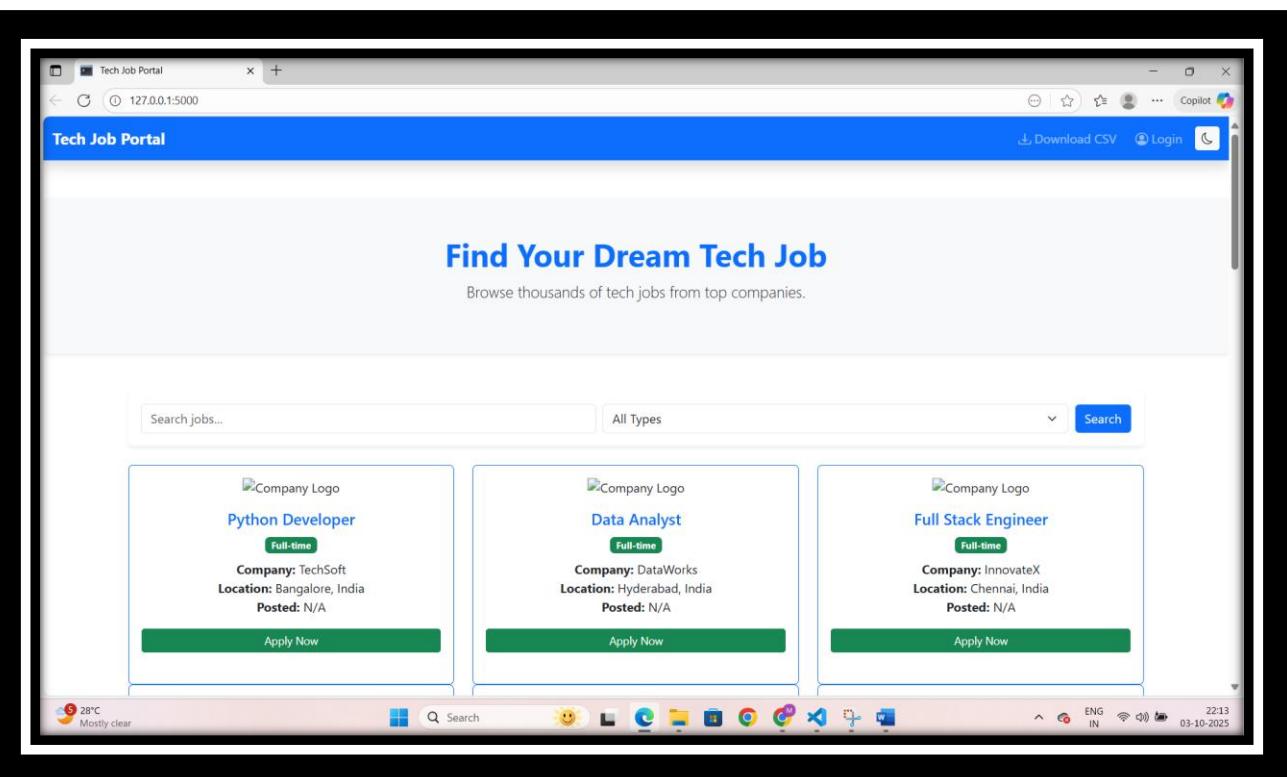
```

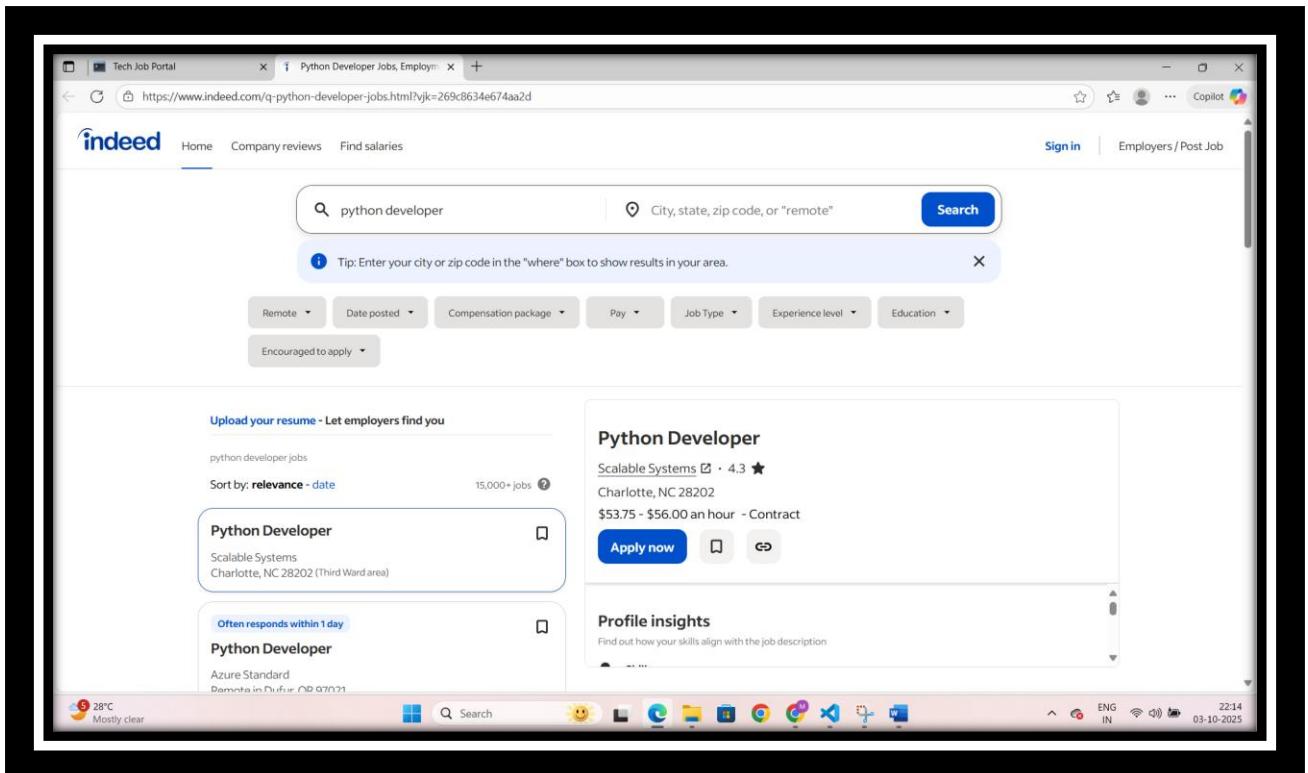
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

C:\Users\manis\Downloads\PROJECTS\tech_job_portal> python app.py
* Serving Flask app "app"
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 569-505-529

Timeline: BLACKBOX Agent Open Website

Add Context... style.css + Add context (#), extensions (@), cor Agent >





FUTURE ENHANCEMENTS :

1. **User Authentication** – Save favorite jobs
2. **Advanced Filtering** – Filter by experience, salary, company
3. **Real-Time Updates** – Periodic scraping and notifications
4. **Dashboard** – Visualize job trends
5. **Database Integration** – MySQL/MongoDB for large datasets
6. **Multi-Website Support** – Scrape multiple job portals

CONCLUSION :

The **Tech Job Portal** demonstrates how Python, BeautifulSoup, Pandas, and Flask can automate job data collection and presentation. It provides a **customizable, efficient, and scalable solution** for job seekers, recruiters, and researchers. By combining automated scraping with a responsive web interface and CSV export functionality, the portal simplifies monitoring of tech job trends. Future enhancements like dashboards, notifications, and multi-site scraping will extend its utility as a comprehensive employment analytics platform.

REFERENCES :

[Indeed](#)

[BeautifulSoup Documentation](#)

[Pandas Documentation](#)

Flask Documentation

Bootstrap 5 Documentation