

Training Report

Summer - 2020

...

Nandeesh Gupta
Roll No - 07014802718
MAIT, Rohini

Overview

Start Month	End Month	Training	Platform
Jan	May	Teaching Assistant Internship	Online at Coding Ninjas
May	June	Front End Web Development Training	Online from Coding Blocks
June	July	'Python For Everybody' Training	University of Michigan (Coursera)

Coding Ninjas Teaching Assistant Internship

...

Jan 2020 - May 2020

Responsibilities

1. Taking doubts of students related to Data Structures and Algorithms in C++
2. Everyday devoting 2-3 hours time, online at Coding Ninjas
3. Resolve the doubts of students by Call or Text and to Debug their Codes.

Outcomes

1. Strengthened my grasp at Data Structures and Algorithms with C++
2. Improved my teaching and communication soft skills
3. Received apt stipend



Certificate Of Experience

This is to certify that Nandeesh Gupta worked with us at Coding Ninjas from January 2020 to May 2020 as an intern.

Key responsibilities during the internship were -

- Taking doubt sessions
- Debugging codes
- Helping students in their assignments & various projects.

We have found Nandeesh to be a self starter who is motivated, duty bound and hard working.

We wish Nandeesh best of luck for future.



Authorised Signatory

(Sunrise Mentors Pvt. Ltd.)

Certificate

Front End Web Development Training

...

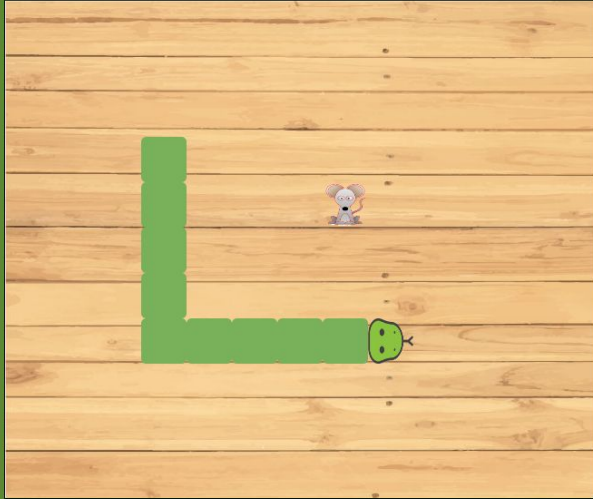
May 2020 - June 2020

Details

1. Tech-Stack - HTML, CSS, JavaScript, vis.js
2. This training has made me able to quickly realise my visions into a light front-end setup using the aforementioned tech-stack.
3. I made some projects, hosted live on the Web right now, explained ahead.

The Snake Game

The Snake Game



SCORE

10

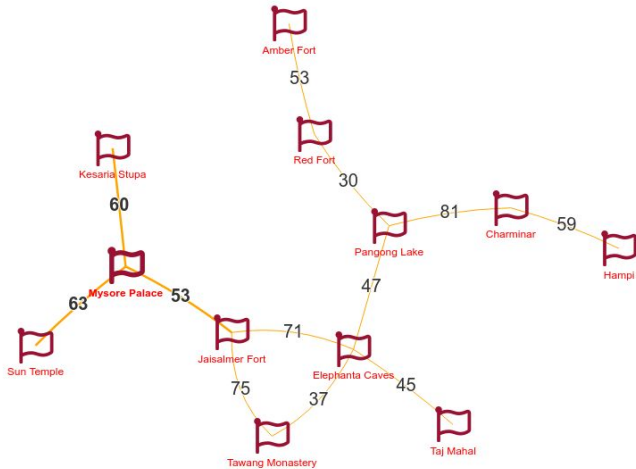
WILL THE SNAKE WIN?

Movement - Arrow Keys
Reset - r

1. Languages used only - HTML, CSS and JavaScript.
2. Required - Canvas Element, Game Loops and Event watchers.
3. Mouse placed randomly. Snake Length Calculation and Movement.
4. Score Keeping.
5. Live on <https://nandeeshg.github.io/SnakeGame/>
6. Code at <https://github.com/NandeeshG/SnakeGame>

Travel Planner

HOW TO REACH?



1. Utilised vis.js framework to make beautiful graphs.
2. Applied dijkstra's algorithm to find the shortest path.
3. Randomly generating the graph.
4. Live on <https://nandeeshg.github.io/Travel-Planner/>
5. Code at <https://github.com/NandeeshG/Travel-Planner/tree/master>

Find shortest path from Hampi to Mysore Palace

Get New Problem

Solve Problem



CERTIFICATE OF COMPLETION



This certificate is proudly presented to

Nandeesh Gupta

for successfully completing the Data Structures in Real
Life Projects by Coding Blocks

Apr 2020 - Jun 2020

Batch

Manmohan Gupta
(Founder, Coding Blocks)

<https://online.codingblocks.com/certificates/CB01-c7732a5d9>

Certificate

‘Python For Everybody’ Training

...

June 2020 - July 2020

Details

1. I understood the usage of python and its many libraries to interact with files both stored locally and from the web.
2. This training helped me comprehend the need of XML,JSON and how to work with them in python.
3. I also learnt how to work with APIs and Databases on the web using python.
4. I am now able to write rudimentary Web Crawlers with the help of URLLIB and RegEx.
5. Everything I learnt was accompanied with a mini project for better understanding.

Tech-Stack

1. Python 3
2. SQLite
3. Libraries -
 - a. JSON
 - b. BeautifulSoup
 - c. URLLIB
 - d. REGEX (RE)
 - e. XML
 - f. sqlite3

Web Scraping and Crawling

1. Using BeautifulSoup library in python.
2. It creates a parse tree for parsed pages that can be used to extract data from HTML, which is useful for web scraping.
3. 2 use cases demonstrated
 - a. Extracting some values from a webpage's HTML
 - b. Following links again and again in a website, to simulate web crawler
4. Demonstration

```
<html>
<head>
<title>Welcome to the comments assignment from www.py4e.com</title>
</head>
<body>
<h1>This file contains the actual data for your assignment - good luck!</h1>

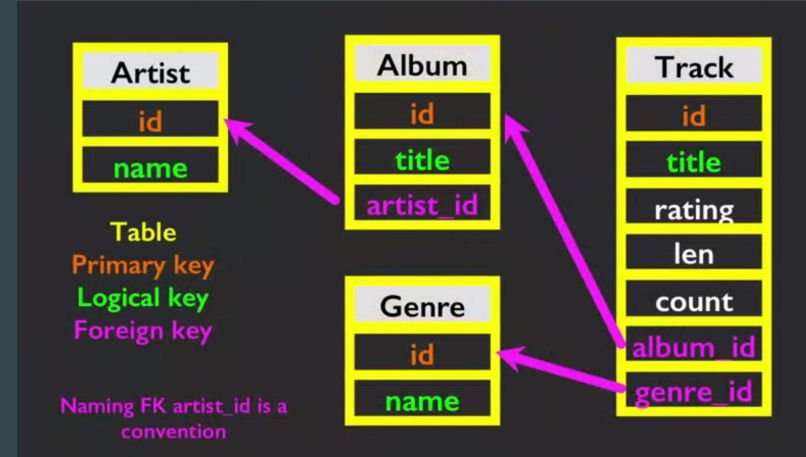
<table border="2">
<tr>
<td>Name</td><td>Comments</td>
</tr>
<tr>
<td>Dolci</td><td><span class="comments">98</span></td></tr>
<tr>
<td>Julita</td><td><span class="comments">95</span></td></tr>
<tr>
<td>Allister</td><td><span class="comments">95</span></td></tr>
<tr>
<td>Kinga</td><td><span class="comments">92</span></td></tr>
<tr>
<td>Eljon</td><td><span class="comments">92</span></td></tr>
<tr>
<td>Shaunpaul</td><td><span class="comments">90</span></td></tr>
<tr>
<td>Jasmine</td><td><span class="comments">90</span></td></tr>
<tr>
<td>Kevauigh</td><td><span class="comments">77</span></td></tr>
<tr>
<td>Lilley</td><td><span class="comments">76</span></td></tr>
<tr>
<td>Banah</td><td><span class="comments">75</span></td></tr>
<tr>
<td>Judah</td><td><span class="comments">73</span></td></tr>
<tr>
<td>Tamzin</td><td><span class="comments">71</span></td></tr>
<tr>
<td>Khajustac</td><td><span class="comments">69</span></td></tr>
<tr>
<td>Aristotelis</td><td><span class="comments">66</span></td></tr>
<tr>
<td>Kristal</td><td><span class="comments">66</span></td></tr>
<tr>
<td>Becky</td><td><span class="comments">64</span></td></tr>
<tr>
<td>Mackena</td><td><span class="comments">56</span></td></tr>
<tr>
<td>Caseyleigh</td><td><span class="comments">53</span></td></tr>
<tr>
<td>Janie</td><td><span class="comments">51</span></td></tr>
<tr>
<td>Lukmaan</td><td><span class="comments">50</span></td></tr>
<tr>
<td>Viki</td><td><span class="comments">49</span></td></tr>
<tr>
<td>Laurabeth</td><td><span class="comments">49</span></td></tr>
<tr>
<td>Miles</td><td><span class="comments">40</span></td></tr>
<tr>
<td>Jess</td><td><span class="comments">48</span></td></tr>
<tr>
<td>Xiong</td><td><span class="comments">38</span></td></tr>
<tr>
<td>Odhran</td><td><span class="comments">37</span></td></tr>
<tr>
<td>Roxie</td><td><span class="comments">35</span></td></tr>
<tr>
<td>Alexandrina</td><td><span class="comments">34</span></td></tr>
<tr>
<td>Euphemia</td><td><span class="comments">32</span></td></tr>
<tr>
<td>Karla</td><td><span class="comments">31</span></td></tr>
<tr>
<td>Lotte</td><td><span class="comments">31</span></td></tr>
<tr>
<td>Keeley</td><td><span class="comments">29</span></td></tr>
<tr>
<td>Mirryn</td><td><span class="comments">29</span></td></tr>
<tr>
<td>Elise</td><td><span class="comments">28</span></td></tr>
<tr>
<td>Kenza</td><td><span class="comments">28</span></td></tr>
<tr>
<td>Donald</td><td><span class="comments">23</span></td></tr>
<tr>
<td>Susanah</td><td><span class="comments">21</span></td></tr>
<tr>
<td>Gjan</td><td><span class="comments">21</span></td></tr>
<tr>
<td>Matthew</td><td><span class="comments">21</span></td></tr>
<tr>
<td>Suzanne</td><td><span class="comments">20</span></td></tr>
<tr>
<td>Sohan</td><td><span class="comments">15</span></td></tr>
<tr>
<td>Zili</td><td><span class="comments">14</span></td></tr>
<tr>
<td>Alishan</td><td><span class="comments">14</span></td></tr>
<tr>
<td>Alikber</td><td><span class="comments">13</span></td></tr>
<tr>
<td>Ines</td><td><span class="comments">9</span></td></tr>
<tr>
<td>Aaren</td><td><span class="comments">9</span></td></tr>
<tr>
<td>Maisy</td><td><span class="comments">8</span></td></tr>
<tr>
<td>Hopkins</td><td><span class="comments">3</span></td></tr>
<tr>
<td>Artur</td><td><span class="comments">3</span></td></tr>
<tr>
<td>Bazas</td><td><span class="comments">1</span></td></tr>
```

People that Dhavid knows

- [Adnan](#)
- [Wu](#)
- [Ashlee](#)
- [Derin](#)
- [Jordy](#)
- [Emilyann](#)
- [Ishaal](#)
- [Launi](#)
- [Leena](#)
- [Marlon](#)
- [Mical](#)
- [Muriel](#)
- [Isobel](#)
- [Doire](#)
- [Symon](#)
- [Jay](#)
- [Elias](#)
- [Tiana](#)
- [Damla](#)
- [Cheyenne](#)
- [Yasmeen](#)
- [Alickber](#)
- [Katerina](#)

SQLite Music Database

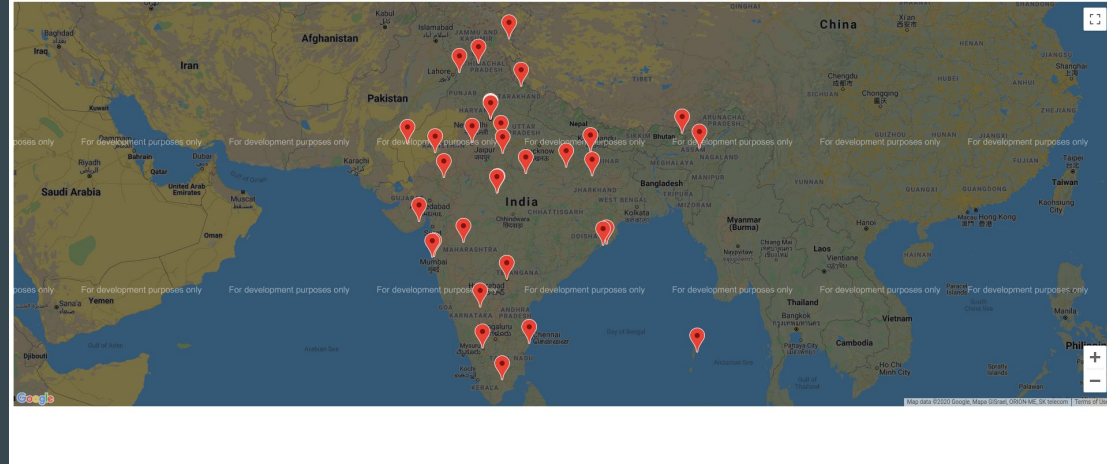
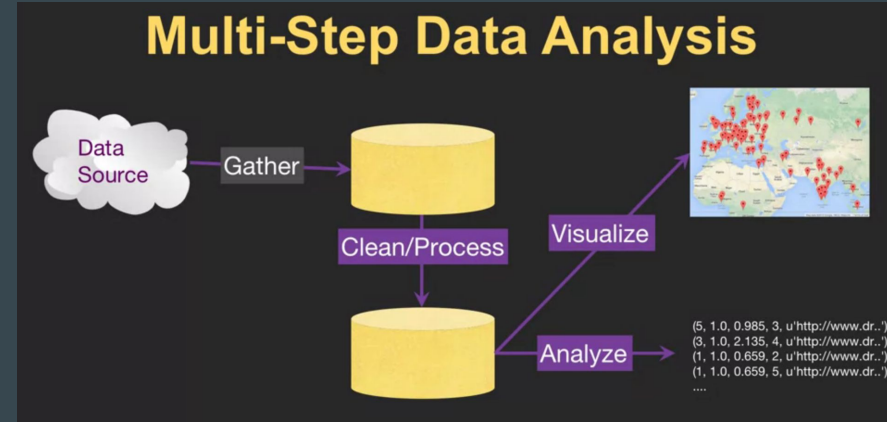
1. Using SQLite3 library in python. SQLite is a RDBMS which is not a client-server database engine. Rather, it is embedded into the end program.
2. Directly Create, Remove, Update Data from the Database using python.
3. Reading XML (or JSON) and use SQLite3 to fill the Database
4. Queries can be done both from python or SQLite browser.
5. Demonstration



Name	Type	Schema
Tables (5)		
Album		CREATE TABLE Album (id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE
id	INTEGER	'id' INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE
artist_id	INTEGER	'artist_id' INTEGER
title	TEXT	'title' TEXT UNIQUE
Artist		CREATE TABLE Artist (id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,
id	INTEGER	'id' INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE
name	TEXT	'name' TEXT UNIQUE
Genre		CREATE TABLE Genre (id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE
id	INTEGER	'id' INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE
name	TEXT	'name' TEXT UNIQUE
Track		CREATE TABLE Track (id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,
id	INTEGER	'id' INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE
title	TEXT	'title' TEXT UNIQUE
album_id	INTEGER	'album_id' INTEGER
genre_id	INTEGER	'genre_id' INTEGER
len	INTEGER	'len' INTEGER
rating	INTEGER	'rating' INTEGER
count	INTEGER	'count' INTEGER

Find and Visualize Tourist Places

1. Gathering is restartable and happens Many times (maybe due to rate limits too) instead of once. Database Helps us in this manner by keeping the Data yet received safe and organised.
2. To visualize, I have taken help of Google API and the tutorials from there.
3. Demonstration





07/10/2020

Nandeesh Gupta

has successfully completed

Using Python to Access Web Data

an online non-credit course authorized by University of Michigan and offered through Coursera

A handwritten signature in black ink, appearing to read 'Charles', followed by a horizontal line.

Charles Severance
Clinical Professor, School of Information
University of Michigan

COURSE
CERTIFICATE



Certificate

Thank You!



Nandeesh Gupta
Roll no - 07014802718
MAIT, Rohini

Also Focused on Competitive Programming

1. Google Kickstart
2. Facebook Hackercup
3. Codeforces Rating 1500+ (Specialist)
4. Codechef 4Star
5. More than 500 problems solved across various judges.
6. Able to use advanced Data Structures and Algorithms like KMP-String Matching, Segment Trees, Tries and more.