

BASAVARAJESWARI GROUP OF INSTITUTIONS

Ballari Institute of Technology & Management

AUTONOMOUS INSTITUTE UNDER VISVESVARAYA TECHNOLOGICAL UNIVERSITY JNANA SANGAMA, BELAGAVI
590018

INTERNSHIP

Report On

28-09-2024

CHAMPIONSHIP HISTORY TRACKER

Submitted in partial fulfilment of the requirements for the award of degree of

Bachelor of Engineering
In
COMPUTER SCIENCE AND ENGINEERING –
DATA SCIENCE

Submitted by

k.Ambika

3BR23CD034

Internship Carried Out By
EZ TRAININGS & TECHNOLOGIES PVT.LTD HYDERABAD

Internal Guide

V P ANUSHYA

Asst.prof,CSE-DS

KAMALAPADU VARSHA

Teaching Asst.CSE-DS

External Guide

vishal kumar

Technical Trainer

BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

NACC Accredited Institution*

(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to Visvesvaraya Technological University, Belagavi)

"Jnana Gangotri" Campus, No.873/2, Ballari-Hospet Road, Allipur,
Ballar1-583 104 (Karnataka) (India) Ph: 08392 – 237100
/ 237190, Fax: 08392 – 237197

2023-2024

BASAVARAJESWARI GROUP OF INSTITUTIONS

BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

Autonomous institute under VISVESVARAYA TECHNOLOGICAL UNIVERSITY JNANA SANGAMA,
BELAGAVI 590018



NACC Accredited Institution*
(Recognized by Govt. of Karnataka, approved by AICTE, New Delhi & Affiliated to
Visvesvaraya Technological University, Belagavi)
"JnanaGangotri" Campus, No.873/2, Ballari-Hospet Road, Allipur,
Ballari 583 104 (Karnataka) (India)
Ph: 08392 - 237100 / 237190, Fax: 08392 - 237197



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Internship entitled "**CHAMPIONSHIP HISTORY TRACKER**" has been successfully completed by K.Ambika bearing USN **3BR23CD034** a bonafide student of Ballari Institute of Technology and Management,

Ballari. For the partial fulfillment of the requirements for the **Bachelor's Degree in Computer**

Science and Engineering-Data science of the VISVESVARAYA TECHNOLOGICAL UNIVERSITY, Belagavi during the academic year 2023-2024.

Signature of Internship

Co-ordinator

V P ANUSHYA

Asst.prof,CSE-DS

KAMALAPADU VARSHA

Asst. prof,CSE-DS

Signature of HOD

DR.AARADHANA

Prof. and HOD(CSE)

DECLARATION

I, K.Ambika , second year student of Computer Science and Engineering, Ballari Institute of Technology, Ballari, declare that Internship entitled CHAMPIONSHIP HISTORY TRACKER is a part of Internship Training successfully carried out by **EZ TECHNOLOGIES & TRAININGS PVT.LTD ,Hyderabad** at "**BITM,BALLARI**". This report is submitted in partial fulfillment of the requirements for the award of the degree, Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belagavi.

Date : 28/09/2024

Place : BALLARI.

Signature of the Student

ACKNOWLEDGEMENT

The satisfactions that a company the successful completion of my internship on "CHAMPIONSHIP HISTORY TRACKER " would be incomplete without the mention of people who made it possible, whose noble gesture, affection, guidance, encouragement and support crowned my efforts with success. It is my privilege to express my gratitude and respect to all those who inspired me in the completion of my internship.

I am grateful to our respective coordinator "**V P Anushya (Asst.prof,CSE-DS) , Kamalapadu Varsha (Teaching.Asst.CSE-DS)**" for his noble gesture, support co-ordination and valuable suggestions given to me in the completion of Internship.

I also thank **DR. AARADHANA**, H.O.D. Department of Computer science and engineering-Data Science for extending all his valuable support and encouragement.

Table of Contents

Chapter No.	Chapter Name	Page No.
1	Day to day activity(student diary extract)	01-02
2	Company Profile	03
3	Abstract	04
4	Introduction of the project	05
5	Description	06-07
6	Algorithm	08-09
7	Flowchart	10
8	Source Code	11-15
9	Output	16-20
10	Conclusion	21
11	References	22

CHAPTER-1

DAY TO DAY ACTIVITIES



Basavarajeshwari Group of Institutions
BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT
Autonomous Institute under VTU - Belagavi
"Jnana Gangotri" Campus, Bellary-Hospet Road, Near Allipura Village,
BALLARI - 583 104 (Karnataka)
Ph: 08392-237167/237153 Fax: 237197, e-mail: bitmhs@gmail.com
Website: www.bitm.edu.in



Internship Program on Python for BE-3rd Sem students
From 9th September to 28th September 2024 (During 3rd semester vacations).

Student Name: K.Ambika _ USN No: 3BR23CD034 Branch: CSE-DS

INDEX PAGE

Day	Date	Content Covered	Signature of the faculty in-charge
1	9.09.24	Introduction to Python, Setup & Installation, First Python Program, Variables, Data Types, and Basic I/O	
2	10.09.24	Control Structures: If-else, Loops, Functions and Modules	
3	11.09.24	Lists, Tuples, and Dictionaries, File Handling	
4	12.09.24	Exception Handling, Practice exercises on Python basics	
5	13.09.24	Introduction to OOP, Classes, and Objects	
6	14.09.24	Inheritance, Polymorphism, and Encapsulation	
7	15.09.24	Abstract Classes and Interfaces	
8	17.09.24	Practice exercises on OOP concepts	
9	18.09.24	Introduction to DSA, Arrays, and Linked Lists	
10	19.09.24	Stacks and Queues	
11	20.09.24	Trees and Graphs	
12	21.09.24	Searching and Sorting Algorithms	
13	23.09.24	Project Building & Presentations	
14	24.09.24	Project Building & Presentations	

15	25.09.24	Project Building & Presentations	
16	26.09.24	Project Building & Presentations	
17	27.09.24	Project Building & Presentations	
18	28.09.24	Project Building & Presentations	

Chapter 2

COMPANY PROFILE

Company Name: EZ Trainings and Technologies Pvt. Ltd.

Introduction:

EZ Trainings and Technologies Pvt. Ltd. is a dynamic and innovative organization dedicated to providing comprehensive training solutions and expert development services. Established with a vision to bridge the gap between academic learning and industry requirements, we specialize in college trainings for students, focusing on preparing them for successful placements. Additionally, we excel in undertaking development projects, leveraging cutting-edge technologies to bring ideas to life.

Mission:

Our mission is to empower the next generation of professionals by imparting relevant skills and knowledge through specialized training programs. We strive to be a catalyst in the career growth of students and contribute to the technological advancement of businesses through our development projects.

Services:

College Trainings:

- Tailored training programs designed to enhance the employability of students.
- Industry-aligned curriculum covering technical and soft skills.
- Placement assistance and career guidance.

Development Projects:

- End-to-end development services, from ideation to execution.
- Expertise in diverse technologies and frameworks.
- Custom solutions to meet specific business needs.

Locations: Hyderabad | Delhi NCR

At EZ Trainings and Technologies Pvt. Ltd., we believe in transforming potential into excellence

Chapter 3

ABSTRACT

- ▶ This project presents a Visa Application System designed to manage visa applications efficiently. The system employs Python programming language, utilizing JSON for data storage. It offers functionalities for creating, reading, updating, deleting, processing visa applications, and tracking application statuses.
- ▶ The system is encapsulated within a Python class named `VisaApplicationSystem`. Upon instantiation, it loads existing visa applications from a JSON file if available. Users interact with the system through a command-line interface.
- ▶ The core functionalities of the system include:
- ▶ Creating Visa Applications: Users can input details such as passport number, application ID, status ID, and countries involved to create new visa applications. Upon creation, the application is appended to the list of applications and saved to the JSON file.
- ▶ Reading Visa Applications: Users can retrieve the details of a specific visa application by providing its application ID. The system searches through existing applications and displays the details if found.
- ▶ Updating Visa Applications: Users can update specific fields of a visa application, such as passport number, status ID, or countries involved. The system searches for the application by its ID, applies the updates, and saves the changes to the JSON file.
- ▶ Deleting Visa Applications: Users can remove a visa application from the system by providing its application ID. The system locates the application, deletes it from the list, and updates the JSON file accordingly.
- ▶ Processing Visa Applications: This functionality marks a visa application as processed. Users input the application ID, and the system updates the status ID of the application to "Processed" and saves the changes.
- ▶ Tracking Application Status: Users can view all visa applications with a specific status ID. The system filters applications based on the provided status ID and displays relevant details.
- ▶ The system ensures data persistence by saving application details to a JSON file after every modification operation. Error handling is implemented to manage scenarios such as file not found or application not found.
- ▶ The command-line interface guides users through various options, enabling seamless interaction with the system. This Visa Application System can be utilized by visa

INTRODUCTION

The Championship History Tracker is a comprehensive and innovative tool designed to record, manage, and analyze championship data across various sports and competitions. This application provides a centralized platform for championship organizers, sports analysts, and historians to track and evaluate team performance, trends, and outcomes.

Description

Problem statement :

Code: Below is a Python solution that encapsulates an object-oriented approach for managing championship records, including creating, reading, updating, and deleting records.

It also allows for documenting championship outcomes and analyzing historical performance trends. We'll use dictionaries and lists for data storage and management.

POC: CRUD: Championship records.
document_championship_outcomes(champion
ship_id): Document and store historical records
of championships

•

analyze_historical_performance(performanc
e_da
ta): Analyze performance trends over time.

Chapter 6

Algorithm

Step 1: Initialize Program*

1. Initialize an empty dictionary `championships` to store championship data.
2. Initialize an empty dictionary `outcomes` to store outcome data.

Step 2: Main Menu

1. Display main menu options:
 - Create Championship
 - Read Championship
 - Update Championship
 - Delete Championship
 - Document Outcome
 - Analyze Performance
 - Exit
2. Get user input for chosen action.

Step 3: Create Championship

1. Get user input for championship ID, year, and winner.
2. Create a new dictionary entry in `championships` with ID as key and year, winner as values.
3. Store championship data in `championships`.

Step 4: Read Championship

1. Get user input for championship ID.
2. Retrieve championship data from `championships` using ID.
3. Display championship data.

Step 5: Update Championship*

1. Get user input for championship ID, year, and winner.
2. Update existing championship data in `championships` using ID.
3. Store updated championship data in `championships`.

Step 6: Delete Championship

1. Get user input for championship ID.
2. Remove championship data from `championships` using ID.

Step 7: Document Outcome

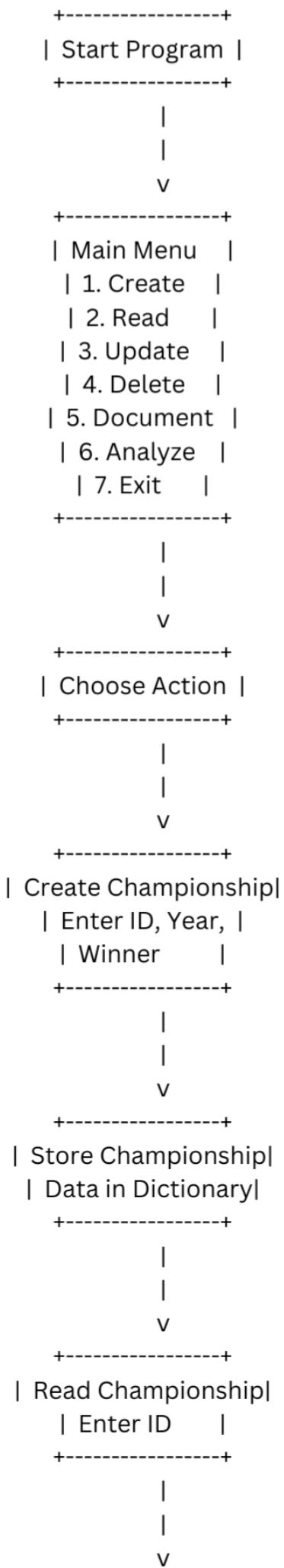
1. Get user input for championship ID and outcome.
2. Create a new dictionary entry in `outcomes` with ID as key and outcome as value.
3. Store outcome data in `outcomes`.

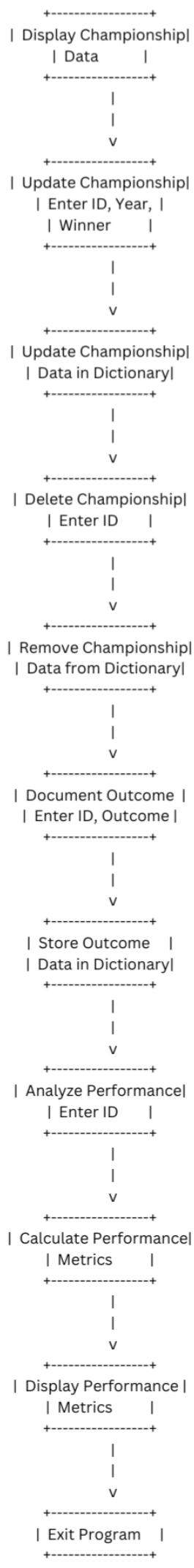
Step 8: Analyze Performance

1. Get user input for championship ID.
2. Retrieve outcome data from `outcomes` using ID.
3. Calculate performance metrics (wins, losses, win rate).

Chapter 7

Flowchart





Chapter 8

Source code

```
# Initialize an empty dictionary to store championship history
championship_history = {}

# Function to create a new championship entry
def create_championship():
    championship_name = input("Enter the championship name: ")
    year = input("Enter the year of the championship: ")
    outcome = input("Enter the outcome of the championship: ")
    championship_history[championship_name] = {"Year": year,
                                                "Outcome": outcome}
    print("Championship entry created successfully.")

# Function to read the championship history
def read_championship():
    for championship, details in championship_history.items():
        print(f"Championship: {championship} - Year: {details['Year']}"
              "- Outcome: {details['Outcome']}")

# Function to update a championship entry
def update_championship():
    championship_name = input("Enter the championship name to
                               update: ")
    if championship_name in championship_history:
        new_outcome = input("Enter the new outcome: ")
        new_year = input("Enter the new year: ")
        championship_history[championship_name]["Outcome"] =
            new_outcome
        championship_history[championship_name]["Year"] =
            new_year
        print("Championship entry updated successfully.")
    else:
        print("Championship not found in the history.")
```

```

# Function to delete a championship entry
def delete_championship():
championship_name = input("Enter the championship name to delete: ")
    if championship_name in championship_history:
        del championship_history[championship_name]
        print("Championship entry deleted successfully.")
    else:
        print("Championship not found in the history.")

# Function to document outcomes (not implemented in this example)
def document_outcome():
    print("Document Outcome feature not implemented.")

# Function to analyze performance (not implemented in this example)
def analyze_performance():
    print("Performance analysis feature not implemented.")

# Main program loop
while True:
    print("\nChampionship History Tracker Menu:")
    print("1. Create Championship")
    print("2. Read Championship")
    print("3. Update Championship")
    print("4. Delete Championship")
    print("5. Document Outcome")
    print("6. Analyze Performance")
    print("7. Exit")

    choice = input("Enter your choice: ")

        if choice == '1':
            create_championship()
        elif choice == '2':
            read_championship()
        elif choice == '3':
            update_championship()
        elif choice == '4':
            delete_championship()
        elif choice == '5':
            document_outcome()
        elif choice == '6':
            analyze_performance()
        elif choice == '7':
            print("Exiting the Championship History Tracker. Goodbye!")
            break
        else:
            print("Invalid choice. Please enter a valid option.")

```

Chapter 9

Output

Championship History Tracker Menu:

1. Create Championship
2. Read Championship
3. Update Championship
4. Delete Championship
5. Document Outcome
6. Analyze Performance
7. Exit

Enter your choice: 1

Enter the championship name: CH001

Enter the year of the championship: 2020

Enter the outcome of the championship: team A

Championship entry created successfully.

Championship History Tracker Menu:

1. Create Championship
2. Read Championship
3. Update Championship
4. Delete Championship
5. Document Outcome
6. Analyze Performance
7. Exit

Enter your choice: 2

Championship: CH001 - Year: 2020 - Outcome: team A

Championship History Tracker Menu:

1. Create Championship
2. Read Championship
3. Update Championship
4. Delete Championship
5. Document Outcome
6. Analyze Performance
7. Exit

Enter your choice: 3

Enter the championship name to update: CH001

Enter the new outcome: CH002

Enter the new year: 2020

Championship entry updated successfully.

Chapter 9 Championship History Tracker Menu:

1. Create Championship
2. Read Championship
3. Update Championship
4. Delete Championship
5. Document Outcome
6. Analyze Performance
7. Exit

Enter your choice: 3

Enter the championship name to update: CH001

Enter the new outcome: CH002

Enter the new year: 2020

Championship entry updated successfully.

Championship History Tracker Menu:

1. Create Championship
2. Read Championship
3. Update Championship
4. Delete Championship
5. Document Outcome
6. Analyze Performance
7. Exit

Enter your choice: 4

Enter the championship name to delete: CH001

Championship entry deleted successfully.

Championship History Tracker Menu:

1. Create Championship
2. Read Championship
3. Update Championship
4. Delete Championship
5. Document Outcome
6. Analyze Performance
7. Exit

Enter your choice: 7

Exiting the Championship History Tracker. Goodbye!

Conclusion

Summary: A Championship History Tracker is essential for preserving the legacy and history of competitive events.

Final Thought: It's a valuable tool for sports analysts, historians, and enthusiasts alike