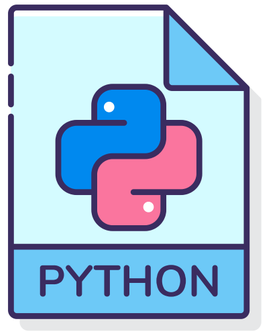
Codveda

Python Development



Content

1. About Us
2. Instructions
3. Submission
4. Task List

About Us

Welcome to Codveda Technology, where innovation meets excellence. Founded with a vision to empower businesses through cutting-edge IT solutions, we specialize in delivering tailored services that drive success in the digital era.

At Codveda, we offer a diverse range of services, including web development, app development, digital marketing, SEO optimization, AI/ML automation, and data analysis.

Our team of skilled professionals is committed to helping businesses unlock their full potential by providing innovative, scalable, and reliable solutions.

INSTRUCTIONS

- Update your LinkedIn profile with your achievements, including the offer letter and completion certificate. Mention and tag @Codveda in your posts.
- Use hashtags like #CodvedaJourney, #CodvedaExperience, and #FutureWithCodveda to showcase your progress and experiences.
- Share your project completion updates on LinkedIn, accompanied by a video explanation and the GitHub project repository link.
- You will be provided with four tasks. Select and complete any three tasks within your domain to fulfill the internship requirements.
- Submit your completed tasks via the Codveda submission form. Ensure all tasks are submitted within the allocated 15-day period.

SUBMISSION

- Create a professional video showcasing your internship projects and achievements.
- Host the video on LinkedIn to provide proof of your work and establish credibility among your peers. Consider tagging Codveda Technology in your posts to ensure they are notified of your work using hashtags like #CodvedaAchievements and #CodvedaProjects.
- A SUBMISSION FORM will be shared later. Till then, please continue your tasks and maintain a separate file for each level.
- When posting the video on LinkedIn, include engaging content that highlights your contributions and skills. Tailor the post to your specific internship domain to maximize impact and visibility.

Level 1 (Basic)

Task 1: Simple Calculator:

- Description: Develop a basic calculator that can perform four primary arithmetic operations: addition, subtraction, multiplication, and division.

Objectives:

- Create functions for each operation.
- Take two inputs from the user and allow them to select the desired operation.
- Handle division by zero with appropriate error messages.

Level 1 (Basic)

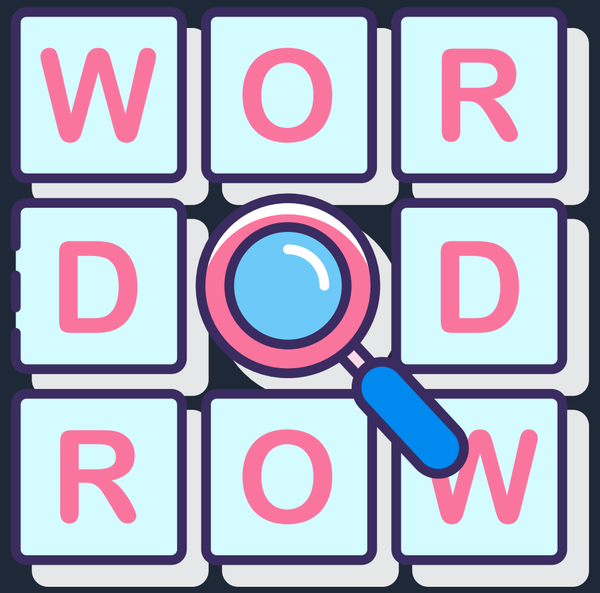
Task 2: Number Guessing Game

- Description: Write a program that randomly generates a number between 1 and 100. The user has to guess the number, and the program will give feedback if the guess is too high or too low.

Objectives:

- Use the random module to generate a random number.
- Give the user multiple attempts to guess the number.
- Provide appropriate feedback (e.g., "Too high" or "Too low").
- Exit the game if the user guesses correctly or after a maximum number of attempts.

Level 1 (Basic)



Task 3: Word Counter

- Description: Create a Python program that reads a text file and counts the number of words in it.

Objectives:

- Read the content of a file.
- Split the content into words and count them.
- Handle exceptions, such as file not found.

Level 2 (Intermediate)

Task 1: To-Do List Application

- Description: Build a simple command-line to-do list application. Users should be able to add, delete, mark as done, and list tasks.

Objectives:

- Implement the ability to add, view, and delete tasks.
- Store the tasks in a file (either CSV or JSON format).
- Mark tasks as completed.
- Implement basic error handling (e.g., trying to delete a task that doesn’t exist).

Level 2 (Intermediate)

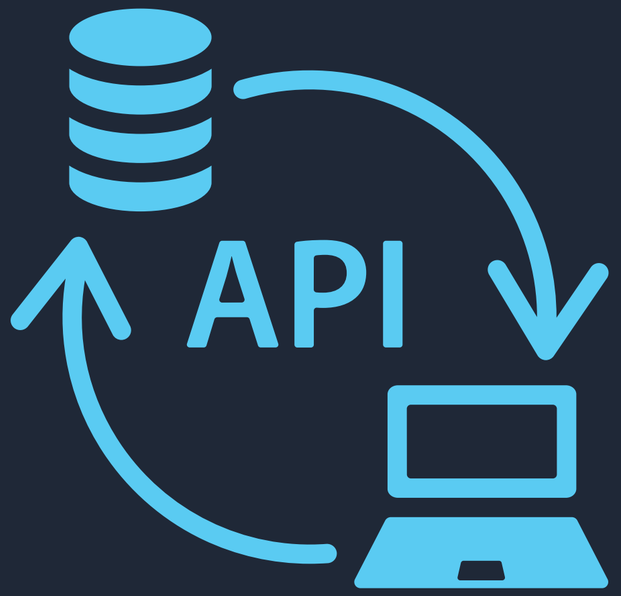
Task 2: Data Scraper

- Description: Develop a web scraper to extract specific data from a website (e.g., news headlines, product prices).

Objectives:

- Use the requests library to retrieve web page content.
- Parse the HTML using BeautifulSoup.
- Extract specific data, such as article titles or product details.
- Save the scraped data into a CSV file.

Level 2 (Intermediate)



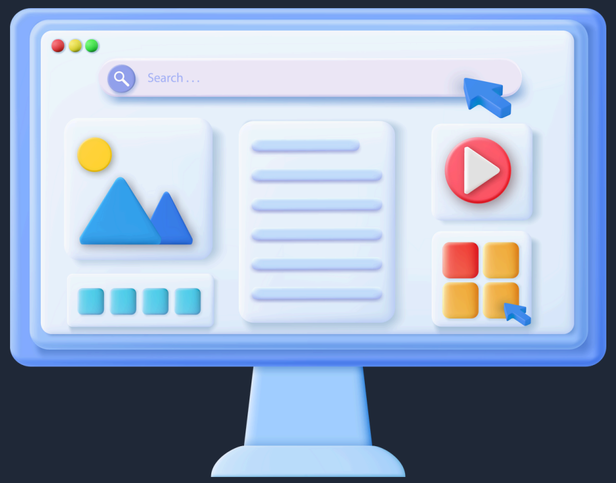
Task 3: API Integration

- Description: Write a Python script that interacts with an external API to fetch and display data (e.g., weather, cryptocurrency prices).

Objectives:

- Use the requests library to make GET requests to an API.
- Parse and display the fetched data in a user-friendly format.
- Handle errors, such as failed requests or invalid responses.

Level 3 (Advanced)



Task 1: Django Web Application with Authentication

- Description: Build a fully functional web application using Django that includes user authentication (login, registration, and password reset). The application can be a blog, task manager, or e-commerce site.

Objectives:

- Implement user registration, login, and logout functionality.
- Secure user passwords using Django’s built-in authentication system.
- Create user roles (e.g., admin, regular user) with different permissions.
- Integrate password reset functionality via email.

Level 3 (Advanced)

Task 2:

Basic File Encryption/Decryption

- Description: Create a Python script that encrypts and decrypts text files using a simple encryption algorithm (e.g., Caesar cipher or Fernet encryption).

Objectives:

- Allow the user to input a file for encryption or decryption.
- Encrypt the file content and save it as a new file.
- Provide functionality to decrypt the file back to its original form.

Level 3 (Advanced)

Task 3: N-Queens Problem

- Description: Solve the classic N-Queens problem where the goal is to place N queens on an N x N chessboard such that no two queens threaten each other.

Objectives:

- Represent the chessboard as a 2D array.
- Use backtracking to place queens one by one in safe positions.
- Ensure that no two queens are on the same row, column, or diagonal.

Challenges: Designing an efficient backtracking solution and correctly handling constraints.

How to Contact Us?

For additional information, kindly get in touch with our team.

@codveda

support@technofyz.com

www.codveda.com