CognoRise
INFOTECH

COGNORISE INFOTECH

Learn -- Innovate -- Thrive

JAVA DEVELOPMENT INTERNSHIP

ABOUT US

CognoRise Infotech thrives as a dynamic and varied community that unites individuals with akin aims and eventual achievements. Our primary concentration resides in establishing prospects that encompass a multitude of domains, encompassing the enhancement of leadership acumen, knowledge acquisition, engagement amongst students, and the cultivation of mutual interests.

We hold a steadfast belief in the potency of leadership and its capacity to propel constructive transformation. Hence, we furnish platforms and reservoirs of support for members within our community to cultivate their leadership proficiencies. By means of mentorship initiatives, interactive workshops, and synergistic undertakings, we endow individuals with the authority to assume leadership positions and engender meaningful alterations in their respective domains.

INSTRUCTIONS

- Update your linkedIn profiles.
- For the internship ,you will need to complete any 3 task at your convenience for successful completion of the internship.
- Create a new Github repository and name it CognoRise Infotech and share the link of the Github repo in the submission form(it will be shared later through email).
- You can refer to online resources such as Google Search and documentation.

SUBMISSION

- A TASK SUBMISSION FORM will be shared later through email . Till then please continue your task.
- Create a demo video of your task, preferably screen recorded.
- The video can be hosted on LinkedIn for proof of your work and build credibility among your peers .
- You can tag CognoRise Infotech on LinkedIn in such posts.
- Please add #cognorise in each of your task video postings on LinkedIn, Additionally, you can also add hashtags such as #internship #webdevelopment. for more reach and visibility.

TASK 1
NUMBER GUESSING GAME

- Generate a random number using the Random class.
- Prompt the user for input using Scanner.
- Compare the user's guess with the generated number.
- Provide feedback on whether the guess is too high, too low, or correct using conditional statements (if).
- Use a loop (e.g., while) to allow multiple attempts.
- Track the number of attempts and stop the game after a predefined limit.

TASK 2
LIBRARY MANAGEMENT SYSTEM

- Create a Book class with attributes like title, author, and status (checked out or available).
- Implement a LibraryCatalog class to manage books.
- Add methods to add books, search by title or author, check out, and return books.
- Use ArrayList to store and manage books.

TASK 3
CALCULATOR APP

- Prompt the user for two numbers and an operator using Scanner.
- Perform the corresponding arithmetic operation based on the operator (+, -, *, /).
- Display the result.
- Use try and catch to handle division by zero or other exceptions.

TASK 4
HANGMAN GAME

- Create an array or list of words to choose from.
- Select a random word using the Random class.
- Use a loop to iterate through each turn.
- Prompt the user for a letter using Scanner.
- Check if the letter is in the word and update the display.
- Display the hangman figure based on incorrect guesses.

TASK 5
STUDENT COURSE REGISTRATION SYSTEM

- Course Database: Store course information, including course code, title,description, capacity, and schedule.
- Student Database: Store student information, including student ID, name, and registered courses.
- Course Listing: Display available courses with details and available slots.
- Student Registration: Allow students to register for courses from the available options.
- Course Removal: Enable students to drop courses they have registered for.

TASK 6
EMAIL APP

- Include the JavaMail API library in your project. You can download it from the official Oracle website or use a build tool like Maven or Gradle to manage dependencies.
- Set up the properties for the SMTP server, including host, port, and authentication details.
- Create a Session object method and pass the properties along with an Authenticator if authentication is required.
- Create a MimeMessage object and set the necessary details such as sender, recipient, subject, and content.
- Use the Transport class to send the email
- Receiving Emails:
- Set up the properties for the IMAP server, including host, port, and authentication details.
- Create a Session object method and pass the properties along with an Authenticator if authentication is required.
- Use the Store class to connect to the email server.
- Use the Folder class to open the inbox or any other folder where you want to retrieve emails.
- Use methods provided by the Folder and Message classes to retrieve and process emails.

TASK 7
SCIENTIFIC CALCULATOR

- Design the calculator's user interface with buttons for digits, arithmetic operations, and scientific functions. Include a display area to show the input and results.
- implement event handlers to handle button clicks. Attach action listeners to the buttons to capture user input.
- Implement methods to handle basic arithmetic operations such as addition, subtraction, multiplication, and division.
- For scientific functions, implement methods for functions like square root, exponentiation, trigonometric functions, etc.
- Update the display area with user input and calculation results.
- Implement functionality for the clear button to clear the input and results, and the delete button to delete the last entered digit.

TASK 8
BRICK BREAKER GAME

- Define the main components of the game, including the paddle, ball, bricks, and the game board. Decide on the dimensions, colors, and positions of these components.
- Create a class for the paddle, which should include methods to move the paddle left and right.
- Create a class for the ball, including methods to move the ball and handle collisions with walls, the paddle, and bricks.
- Create a class for the bricks. Determine the number of rows and columns of bricks and their positions. Implement methods to draw and remove bricks when hit by the ball.
- Implement logic to capture user input for paddle movement.
- Implement collision detection logic to check for collisions between the ball, paddle, and bricks. Update the ball's direction and remove bricks when collisions occur.
- Implement logic to determine when the game is over (e.g., when the ball goes below the paddle) and when the player wins (e.g., when all bricks are destroyed).

INTERNSHIP AIM
LEARNING AND ADVANCEMENT

- The purpose of this internship is to learn and grow
- Our internship is designed to facilitate your learning journey and foster growth.The choice to seek guidance is entirely yours.
- The provided tasks may appear simple or complex.
- We anticipate your approach to be marked by professional commitment, dedicating the attention each task merits.

CONNECT WITH US

- MAIL US: cognoriseinfotech@gmail.com
-
- TELEGRAM: https://t.me/CognoRise_InfoTech
-
- LINKEDIN: https://www.linkedin.com/company/cognoriseinfotech/