JAVA DEVELOPMENT

SKYBUG
Technologies

SKYBUG
Technologies

About Us

SKYBUG Technologies, a leading IT service provider, is dedicated to nurturing the IT talent of tomorrow. Through our diverse range of internship opportunities, we empower students to explore various domains within the IT sector. With a focus on practical experience, expert mentorship, and innovation, we offer a dynamic platform for learning and growth. Join us at SkyBug Technologies and shape your IT future with hands-on experience and a supportive community of like-minded individuals.

INSTRUCTIONS

- For the Java Development Internship, you will need to complete **at least 3 tasks** for Successful Completion of the Internship.

- Maintain a Separate GitHub repository (name as **SKYBUG**) for all the tasks and share the link of the GitHub repo in the task submission form (it will be given letter through email).

- You can refer to online resources such as Google search and read tutorials. Watch videos(for help)

SUBMISSION

- A **Task Submission Form** will be Shared Later through email. Till then Continue your Task

- A video need to be created to showcase your work, a demo of your efforts.

- For the Java Development internship, you will need to complete **at least 3 tasks** for successful completion of the Internship.

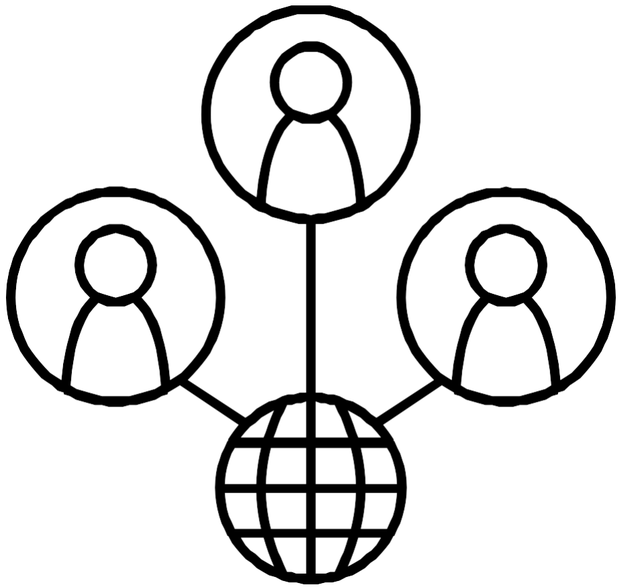
- The Video can be Hosted on LinkedIn for proof of your work and to build credibility among your peers. You can tag **@SKYBUG TECHNOLOGIES** in such posts.

- Please add **#SKYBUG** in each of your task video posting on LinkedIn, Additionally, You can also add hashtags such as #internship #Java Development for more reach and visibility.

About Internship

COMPLETION CERTIFICATE

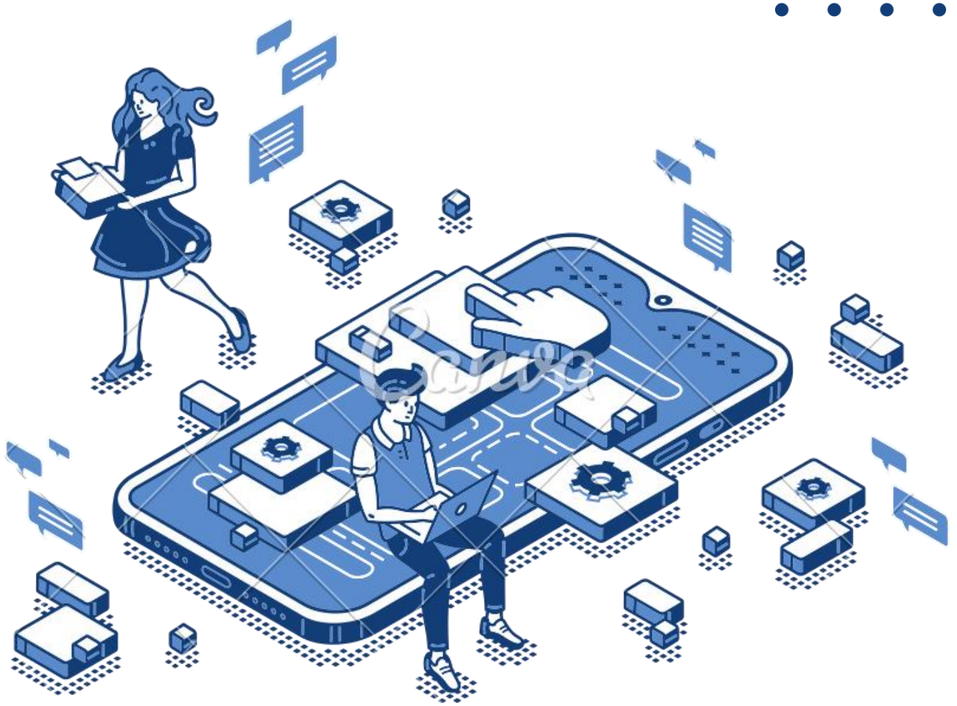
PLACEMENT SUPPORT



NETWORK OPPORTUNITY

SKYBUG Technologies

JAVA DEVELOPMENT



For the Java Development internship, you will need to complete at least 3 tasks for successful completion of the internship.

TASK1

NUMBER GAME

1. Generate a random number within a specified range, such as 1 to 100.

2. Prompt the user to enter their guess for the generated number.

3. Compare the user's guess with the generated number and provide feedback on whether the guess is correct, too high, or too low.

4. Repeat steps 2 and 3 until the user guesses the correct number. You can incorporate additional details as follows:

5. Limit the number of attempts the user has to guess the number.

6. Add the option for multiple rounds, allowing the user to play again.

7. Display the user's score, which can be based on the number of attempts taken or rounds won.

TASK2

QUIZ APLICATION WITH TIMER

1. Quiz Questions and Options: Store quiz questions along with multiple-choice options and correct answers.

2. Timer: Implement a timer for each question to limit the time to answer.

3. Question Display: Present one question at a time with multiple-choice options.

4. Answer Submission: Allow users to select an option and submit their answer within the given time**.**

5. Score Calculation: Keep track of the user's score based on correct answers.

Result Screen: Display the final score and a summary of correct/incorrect answers.

TASK3

STUDENT GRADE CALCULATOR

Input: Take marks obtained (out of 100) in each subject.

Calculate Total Marks: Sum up the marks obtained in all subjects.

Calculate Average Percentage: Divide the total marks by the total number of subjects to get the average percentage.

Grade Calculation: Assign grades based on the average percentage achieved.

Display Results: Show the total marks, average percentage, and the corresponding grade to the user

TASK4

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.

TASK5

ATM INTERFACE

1.Create a class to represent the ATM machine.

2. Design the user interface for the ATM, including options such as withdrawing, depositing, and checking the balance.

3. Implement methods for each option, such as withdraw(amount), deposit(amount), and checkBalance().

4. Create a class to represent the user's bank account, which stores the account balance.

5. Connect the ATM class with the user's bank account class to access and modify the account balance.

6. Validate user input to ensure it is within acceptable limits (e.g., sufficient balance for withdrawals).

7. Display appropriate messages to the user based on their chosen options and the success or failure of their transactions.

ASK US FOR HELP!

THE PURPOSE OF THIS INTERNSHIP IS TO LEARN AND GROW.

We have no desire to dictate to you. It is entirely up to you whether you seek guidance or not.

The given tasks may seem very easy or very difficult. We expect you to approach the tasks with professional diligence and given them the attension they deserve.

Get Social with us!!

www.skybugtech.in

skybugtechnologies@gmail.com

SkyBug Technologies