Likhith Gowda K

An ambitious and modest B.E(CSE) graduate with 8.5 CGPA, proficient in Java, Python, HTML, SQL, CSS and JavaScript with good communication skills and problem-solving skills. I am seeking a position with an organization that values continued learning and growth in emerging technologies, and provides opportunities for mentorship and career advancement.

9606849258 Klgowda2000@gmail.com Bangalore

EDUCATION

R R Institute Of Technology **2019-2023**

B.Tech. Computer Science

CGPA – 8.6

Sri Chaitanya PU College **2017-2019**

Department of Pre-University Education(Karnataka), PCMB

PERCENTAGE – 86.33

St Claret High School **2016-2017**

Secondary School Certificate(SSLC/Tenth)

PERCENTAGE – 97.76

INTERNSHIP - 1

KodNest Technologies Pvt Ltd **Jan,2023 – May,2023**

Role: Full Stack Intern

- Trained in Java SE, Java EE, SQL, HTML, CSS, Java Script and Basics of Data Structures
- Executed Projects – Full Stack Web Application Shop Nest

INTERNSHIP - 2

Karunadu Technologies Pvt Ltd **Aug,2022 – Sep,2022**

Role: Machine Learning Intern

- Trained in Python, PyQt5 and Machine Learning algorithms
- Executed Projects – White Wine Quality Prediction

PROJECT - 1

Shop Nest – E- Commerce Full Stack Web Application

Technologies Used: Oracle, Java SE, Java EE(Servlets & JSP's)

- The aim of an e-commerce website application is to facilitate online buying and selling of products or services.
- It includes features such as product catalog, shopping cart, payment gateway integration, order management, user accounts, and customer

PROJECT - 2

Blood Bank Management System

Technologies Used: Python, HTML, Javascript, CSS, Bootstrap,Flask

- Blood Bank System is a browser based solution that is designed to store, process, retrieve and analyze information concerned with the administrative and clinical aspects of a blood bank.
- It includes features such as requesting blood of particular blood group by the hospital, maintaining the donor information and blood availability status in blood bank.

TECHNICAL & CAREER SKILLS

- Java SE
- Java EE
- Html
- CSS
- Java Script
- SQL
- Critical Thinking
- Problem Solving
- Teamwork & Collaboration
- Research & Analysis
- Python

CERTIFICATIONS

- Certified Full Stack Developer from KodNest
- Certified in Programming, Data Structures and Algorithms Using Python from IIT - Madras – NPTEL
- Certified Software Developer from NCVET

ACHIEVEMENTS

- Earned a bronze-level Java badge on HackerRank
- Earned a silver-level 30 Days of Code on HackerRank
- Have a typing speed of 35 words per minute

HOBBIES

- Watching Movies
- Yoga
- Listening to Music

DECLARATION

I solemnly declare that the information furnished above is free from errors to the best of my knowledge and belief.

Date:

Place:

Signature

BITWISE OPERATORS

Bitwise operators are the operators which are used to perform operations on individual bits of binary number. These operators are commonly used in low-level or machine-level programming.

There are six types of bitwise operators. They are :-

- AND (&)
- OR (|)
- XOR (^)
- NOT (~)
- LEFT SHIFT (<<)
- RIGHT SHIFT(>>)

AND Bitwise Operator (&)

AND bitwise operator is a binary operator which performs AND operation on the corresponding bits of the two binary numbers.

In AND operation the result is 1 only if the two corresponding binary bits are 1, otherwise it is 0.

Eg: 10&6

The binary representation of 10 is 00001010
The binary representation of 6 is 00000110

By performing AND operation on both the binary numbers gives the result:

```
0 0 0 0 1 0 1 0 (10)
0 0 0 0 0 1 1 0 (6)
-----------------
0 0 0 0 0 0 1 0 (2)
```

The result of 10&6 is 2

OR Bitwise Operator (|)

OR bitwise operator is a binary operator which performs OR operation on the corresponding bits of the two binary numbers.

In OR operation the result is 0 only if the two corresponding binary bits are 0, otherwise it is 1.

Eg: 10|6

The binary representation of 10 is 00001010
The binary representation of 6 is 00000110

By performing OR operation on both the binary numbers gives the result:

```
0 0 0 0 1 0 1 0 (10)
0 0 0 0 0 1 1 0 (6)
-----------------
0 0 0 0 1 1 1 0 (14)
```

The result of 10|6 is 14

XOR Bitwise Operator (^)

XOR bitwise operator is a binary operator which performs XOR operation on the corresponding bits of the two binary numbers.

In XOR operation the result is 0 only if the two corresponding binary bits are same, otherwise it is 1.

Eg: 10^6

The binary representation of 10 is 00001010
The binary representation of 6 is 00000110

By performing XOR operation on both the binary numbers gives the result:

```
0 0 0 0 1 0 1 0 (10)
0 0 0 0 0 1 1 0 (6)
-----------------
0 0 0 0 1 1 0 0 (12)
```

The result of 10^6 is 12

NOT Bitwise Operator (~)

NOT bitwise operator is a unary operator which performs one's complement of the given binary number

In one's complement all the bits of the binary number is inverted i.e it inverts 0's to 1's and 1's to 0's.

Eg: ~6

The binary representation of 6 is 00000110

By performing one's complement on the given binary number, the result is :

```
0 0 0 0 0 1 1 0 (6)
-----------------
1 1 1 1 1 0 0 1 1
```

The leftmost bit is 1. So this is a negative binary number. In order to obtain the decimal value we need to perform 2's complement on the given number upon which we get the answer as -7

The result of ~6 is -7

Left Shift Bitwise Operator (<<)

Left Shift bitwise operator is a binary operator which shifts the binary bits present in the first operand to the left based upon the value given in the second operand.

Syntax: a<<b

a - is the first operand in which binary bits are shifted to the left
b – is the second operand which decides the number of left shifts

Eg: 10<<2

Here we need to shift the binary bits in 10 to the left by 2 times

The binary representation of 10 is 00001010

Original Number 0 0 0 0 1 0 1 0 (10)

First Left Shift 0 0 0 1 0 1 0 0 (20)

Second Left Shift 0 0 1 0 1 0 0 0 (40)

The result of 10<<2 is 40

Right Shift Bitwise Operator (>>)

Right Shift bitwise operator is a binary operator which shifts the binary bits present in the first operand to the right based upon the value given in the second operand.

Syntax: a>>b

a - is the first operand in which binary bits are shifted to the right

b – is the second operand which decides the number of right shifts

Eg: 10>>2

Here we need to shift the binary bits in 10 to the right by 2 times

The binary representation of 10 is 00001010

Original Number 0 0 0 0 1 0 1 0 (10)

First Right Shift 0 0 0 0 0 1 0 1 (5)

Second Right Shift 0 0 0 0 0 0 1 0 (2)

The result of 10>>2 is 2