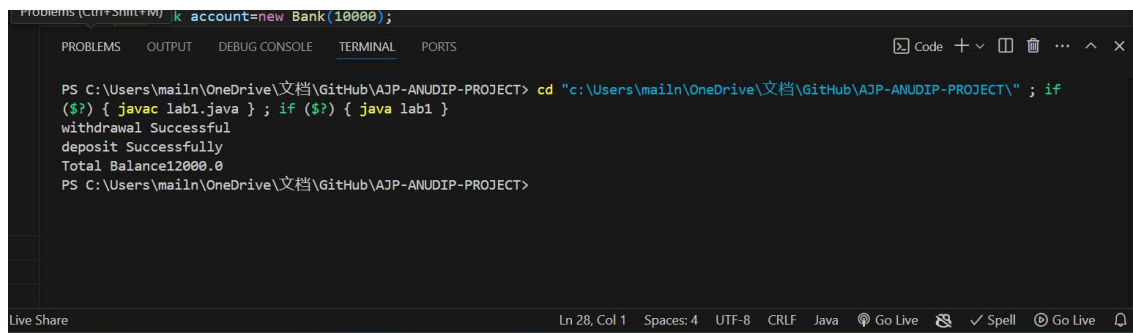


1. Create a Bank class and declare an instance variable named amount of type double. Create parameterized constructor to initialize variable "amount" with value 10000. Create two methods withdraw(double withdrawalAmount) and deposit(double depositAmount). Calculate withdrawal based on some condition (using ternary operator) like If amount is sufficient then "withdraw successful" message will be printed on the console and amount should be updated after withdraw. Later on, deposit 5000 in the account balance. At the end display total balance on the console.

```
import java.util.*;
class Bank{
    private double amount;
    public Bank(double amount){
        this.amount = amount;
    }
    public void withdraw(double withdrawalAmount){
        String message =(withdrawalAmount<=amount) ? "withdrawal Successful" : "Insufficient
Balance";
        System.out.println(message);
        if(withdrawalAmount<=amount){
            amount -= withdrawalAmount;
        }
    }
    public void deposit(double depositAmount){
        amount +=depositAmount;
        System.out.println("deposit Successfully");
    }
    public void displayBalance(){
        System.out.println("Total Balance" + amount);
    }
}
public class lab1{
    public static void main(String args[]){
        Bank account=new Bank(10000);
        account.withdraw(3000);
        account.deposit(5000);
        account.displayBalance();
    }
}
```

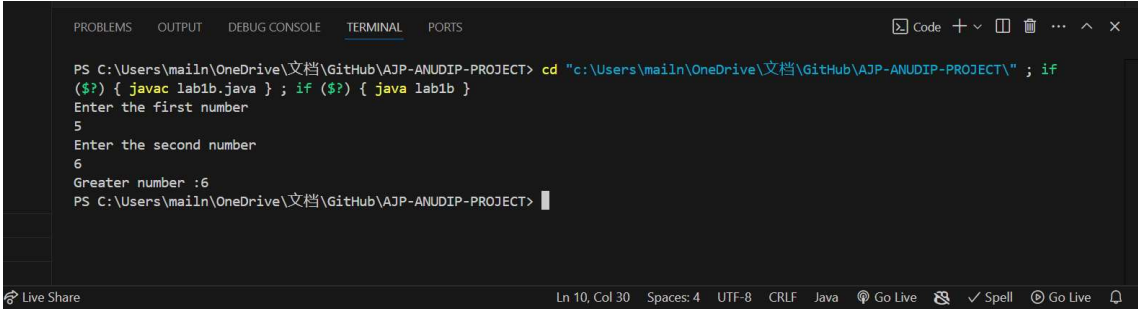


```
Problems (Ctrl+Shift+F) 1k account=new Bank(10000);
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\main\OneDrive\文档\GitHub\AJP-ANUDIP-PROJECT> cd "c:\Users\main\OneDrive\文档\GitHub\AJP-ANUDIP-PROJECT" ; if
($?) { javac lab1.java } ; if ($?) { java lab1 }
withdrawal Successful
deposit Successfully
Total Balance12000.0
PS C:\Users\main\OneDrive\文档\GitHub\AJP-ANUDIP-PROJECT>
```

Live Share Ln 28, Col 1 Spaces: 4 UTF-8 CRLF Java Go Live Spell Go Live

2. Write a program to input two numbers and find the maximum between two numbers using the conditional/ternary operator.

```
import java.util.Scanner;
public class lab1b{
    public static void main(String args[]){
        Scanner sc=new Scanner (System.in);
        System.out.println("Enter the first number");
        int num1 = sc.nextInt();
        System.out.println("Enter the second number");
        int num2 = sc.nextInt();
        int message=(num1>num2)?num1:num2;
        System.out.println("Greater number :" +message);
    }
}
```



The screenshot shows a terminal window with the following content:

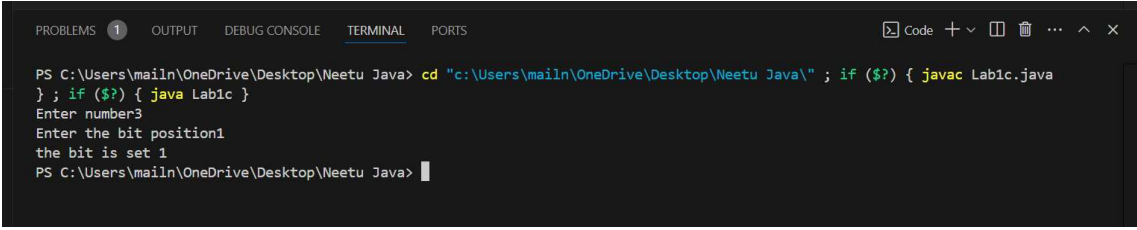
```
PS C:\Users\mailn\OneDrive\文档\GitHub\AJP-ANUDIP-PROJECT> cd "c:\Users\mailn\OneDrive\文档\GitHub\AJP-ANUDIP-PROJECT\" ; if
($?) { javac lab1b.java } ; if ($?) { java lab1b }
Enter the first number
5
Enter the second number
6
Greater number :6
PS C:\Users\mailn\OneDrive\文档\GitHub\AJP-ANUDIP-PROJECT>
```

The terminal window has tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The status bar at the bottom shows 'Ln 10, Col 30', 'Spaces: 4', 'UTF-8', 'CRLF', 'Java', 'Go Live', 'Spell', and 'Go Live'.

3. Write a program to declare two variables num and n and take an input during compilation time to check whether the nth bit of the given number is set (1) or not (0).

```
import java.util.Scanner;
public class Lab1c{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        System.out.print("Enter number");
        int number=sc.nextInt();

        System.out.print("Enter the bit position");
        int n = sc.nextInt();
        boolean isCheck = (number &(1<<n))!=0;
        if(isCheck){
            System.out.println("the bit is set 1");
        }
        else{
            System.out.println("the bit is set 0");
        }
    }
}
```



```
PS C:\Users\mailn\OneDrive\Desktop\Neetu Java> cd "c:\Users\mailn\OneDrive\Desktop\Neetu Java\" ; if ($?) { javac Lab1c.java } ; if ($?) { java Lab1c }
Enter number3
Enter the bit position1
the bit is set 1
PS C:\Users\mailn\OneDrive\Desktop\Neetu Java>
```

