TORCH BEARER CONVENT SCHOOL

(Sr. Secondary School Affiliated to CBSE)

PROJECT FILE

Subject**: Information Technology**

Class: **XII**

Academic Session: **2025–26**

Project Title:

“A Comprehensive Study of SQL Queries, Java Programs, and Web-Based Applications”

Submitted By:

Name**: Rohan Naagar**
Class: **XII – Non-Medical**
Roll No.: **11**
School Name**: Torch Bearer Convent School**

Submitted To:

Subject Teacher's Name**: Mr. Raju Gupta**
Designation: **Teacher(Information Technology)**

Board:

CENTRAL BOARD OF SECONDARY EDUCATION (CBSE) New Delhi, India

Java

MySQL Developement Services

IRCTC

Digital India
Power To Empower

INDEX

Class 12 Project File – Academic Session 2025–26

◆ SQL QUERIES

◆ JAVA PROGRAMS

S. No.	Topic
1.	Print the string "Hello World"
2.	Display characters A to Z using for loop
3.	Check if a number is Positive, Negative, or Zero using if...else
4.	Find the largest among three numbers using if...else
5.	Check whether a number is even or odd using if...else
6.	Generate multiplication table (1 to 10) using for loop
7.	Find the area of a rectangle by taking length and breadth as input
8.	Print the sum of the first 10 natural numbers
9.	Generate Fibonacci series based on user input
10.	Print the factorial of a given number
11.	Calculate the sum of digits of a number
12.	Check whether a number is prime
13.	Reverse a number
14.	Check whether a string is a palindrome
15.	Count the number of vowels in a string

◆ OPERATING WEB-BASED APPLICATIONS

S. No.	Topic
1.	Case Study – Online Bill Calculator / Book Rail Ticket
2.	Case Study – Online Quiz: Internet and Phone Safety for School Children
3.	Case Study – Online Game: Hangman

CERTIFICATE

This is to certify that **Rohan Naagar**, a student of **Class XII – Non-Medical** , has successfully completed the project work titled:

“A Comprehensive Study of SQL Queries, Java Programs, and Web-Based Applications”

as per the guidelines issued by the **Central Board of Secondary Education (CBSE)**, for the academic session **2025–26**, under the subject **Information Technology (code-802)**.

The project submitted is an original work undertaken by the student and has been completed under my supervision and guidance.

I wish him/her all the best for future academic endeavors.

Signature of the Subject Teacher
Name: _____
Designation: _____
Date: _____

Signature of the Principal
Name: _____
School Seal:

Date: _____

Student Details:
Name: **Rohan Naagar**
Class & Section: **XII – Non-Medical**
Roll Number: **11**
Subject: **Information Technology**

Scan the QR for profile

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to all those who helped me complete this project successfully.

First and foremost, I am thankful to the **Almighty** for providing me the strength and determination to undertake and complete this project.

I extend my heartfelt thanks to my **subject teacher, Mr. Raju Gupta**, for their constant support, valuable guidance, and encouragement throughout the course of this project. Their insightful suggestions and constructive feedback played a vital role in shaping the final outcome.

I am also grateful to our **Principal, Mrs. Anjali Kaushik** , for providing the necessary infrastructure and academic environment to carry out this work.

I would also like to thank my **parents and friends** for their moral support, patience, and cooperation during the preparation of this project.

Finally, I acknowledge that this project is the result of dedication, discipline, and collaborative effort, and I am proud to submit it as a part of my Class 12 curriculum.

Name: Rohan Naagar
Class: XII – Non-Medical
Roll Number: 11
Subject: Information Technology

SQL QUERIES

---

1. Write down syntax and MySQL program to create a database STUDENT and its OUTPUT.

AIM

To write a MySQL code to create a database STUDENT, listing out its **SYNTAX**, **PROGRAM**, and the resulting **OUTPUT**.

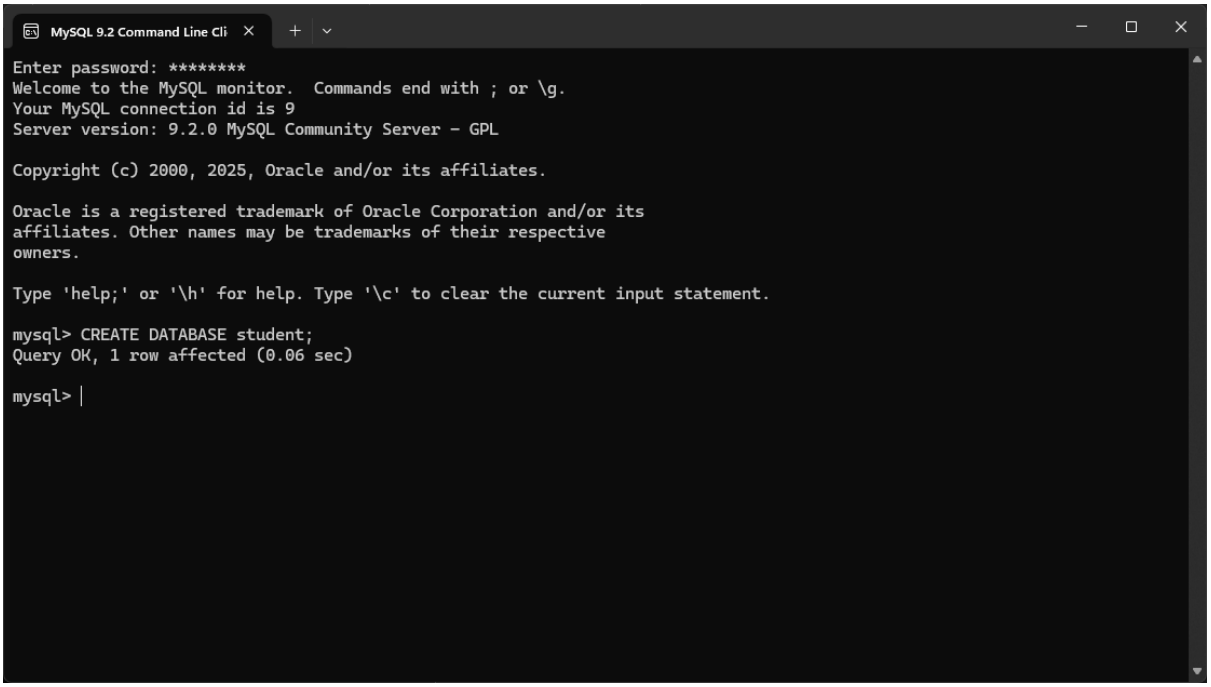
SYNTAX

CREATE DATABASE dbname;

PROGRAM

mysql> CREATE DATABASE student;

OUTPUT



---

2. Write down syntax and MySQL program to delete a database STUDENT and its OUTPUT.

AIM

To write a MySQL code to delete a database STUDENT, listing out its **SYNTAX**, **PROGRAM**, and the resulting **OUTPUT**.

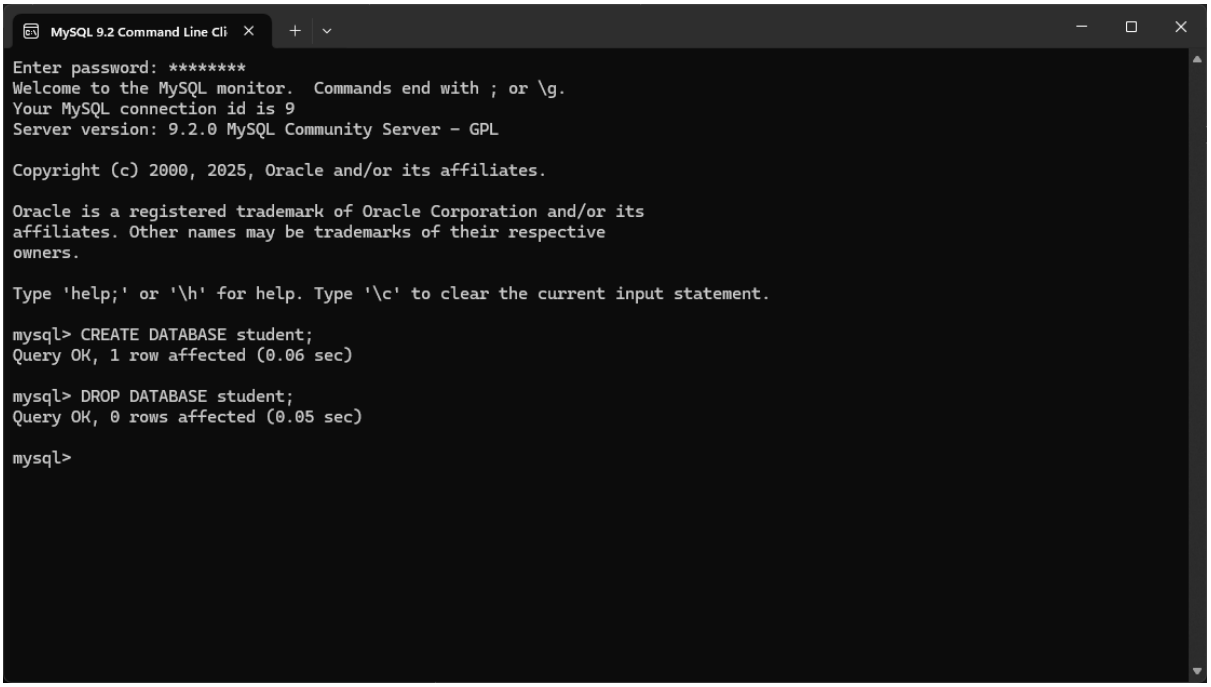
SYNTAX

DROP DATABASE dbname;

PROGRAM

mysql> DROP DATABASE student;

OUTPUT



---

3. Write down syntax and MySQL program to create a table STUDENT and its OUTPUT.

AIM

To write a MySQL code to create a table STUDENT with fields and constraints, listing out the **SYNTAX**, **PROGRAM**, and the **OUTPUT**.

SYNTAX

CREATE TABLE table_name (

column1 datatype constraint,

column2 datatype constraint,

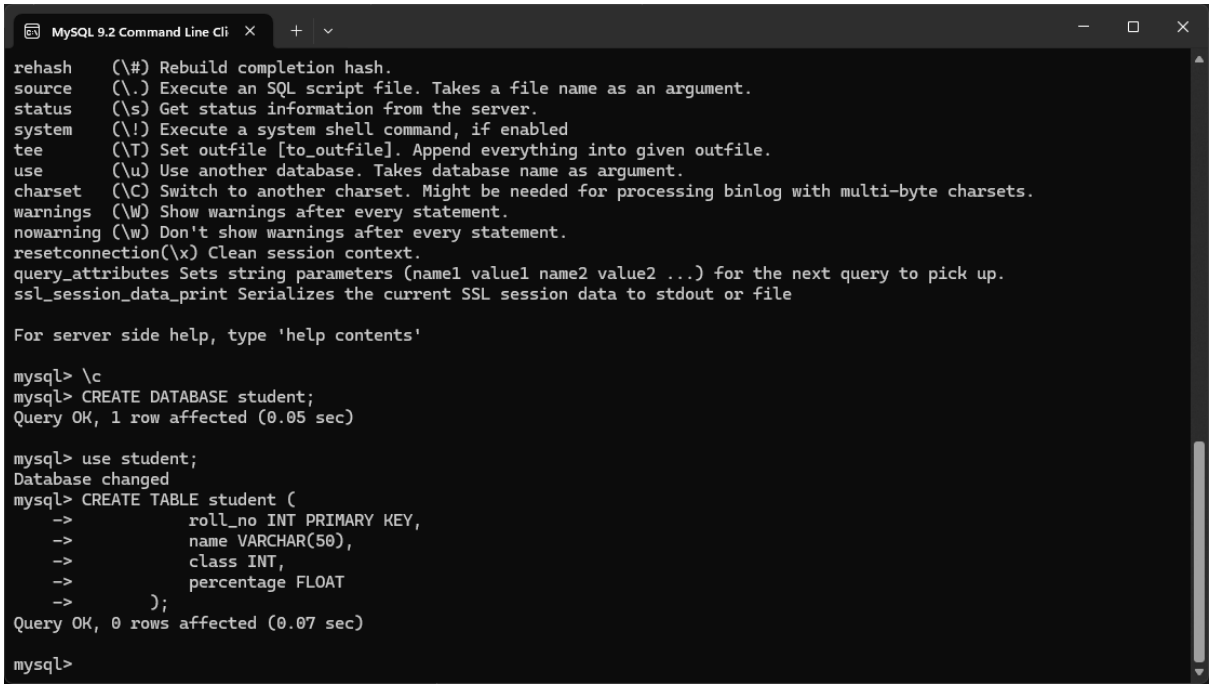
...

);

PROGRAM

```
mysql> CREATE TABLE student (
        roll_no INT PRIMARY KEY,
        name VARCHAR(50), class INT , percentage FLOAT  );
```

OUTPUT



---

4. Write down syntax and MySQL program to show the databases and tables created in MySQL and its OUTPUT.

AIM

To write MySQL commands that display the **list of databases** and **tables** created, along with the **SYNTAX**, **PROGRAM**, and **OUTPUT**.

SYNTAX

SHOW DATABASES;

SHOW TABLES;

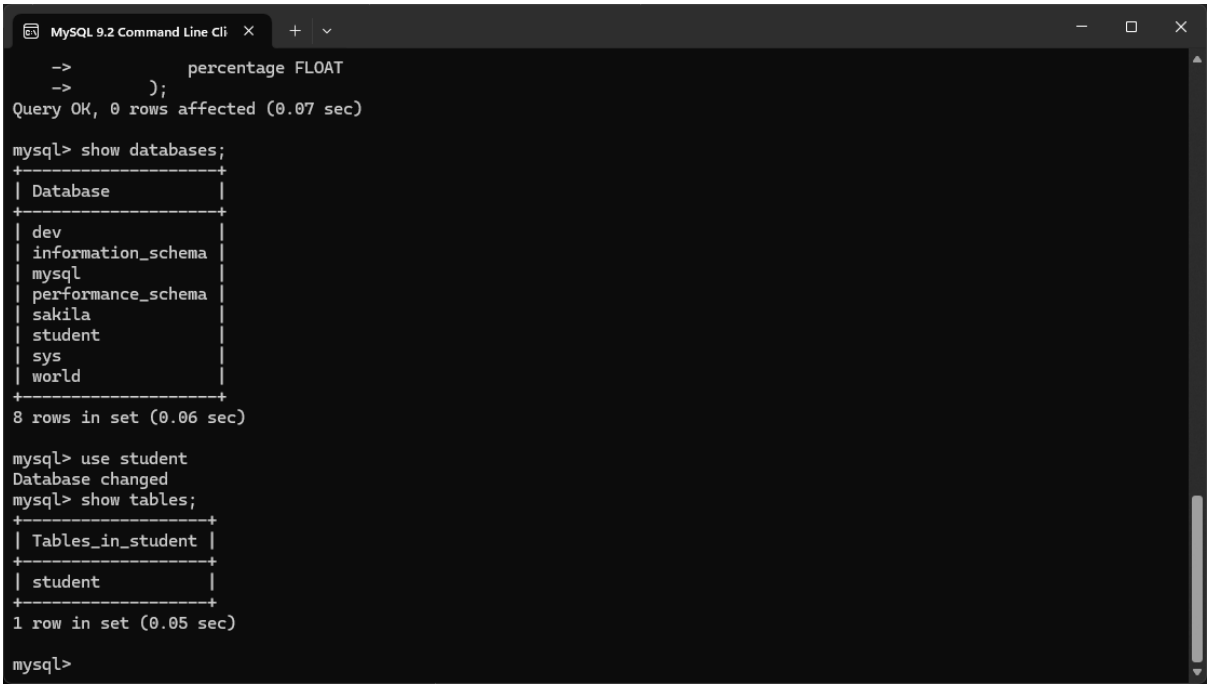
PROGRAM

mysql> SHOW DATABASES;

mysql> USE student;

mysql> SHOW TABLES;

OUTPUT



---

5. Write down syntax and MySQL program to list field names with their data types and constraints for a table, and its OUTPUT.

AIM

To write MySQL code that displays the **structure of a table** STUDENT, showing each **field name**, its **data type**, and any **constraints** applied.

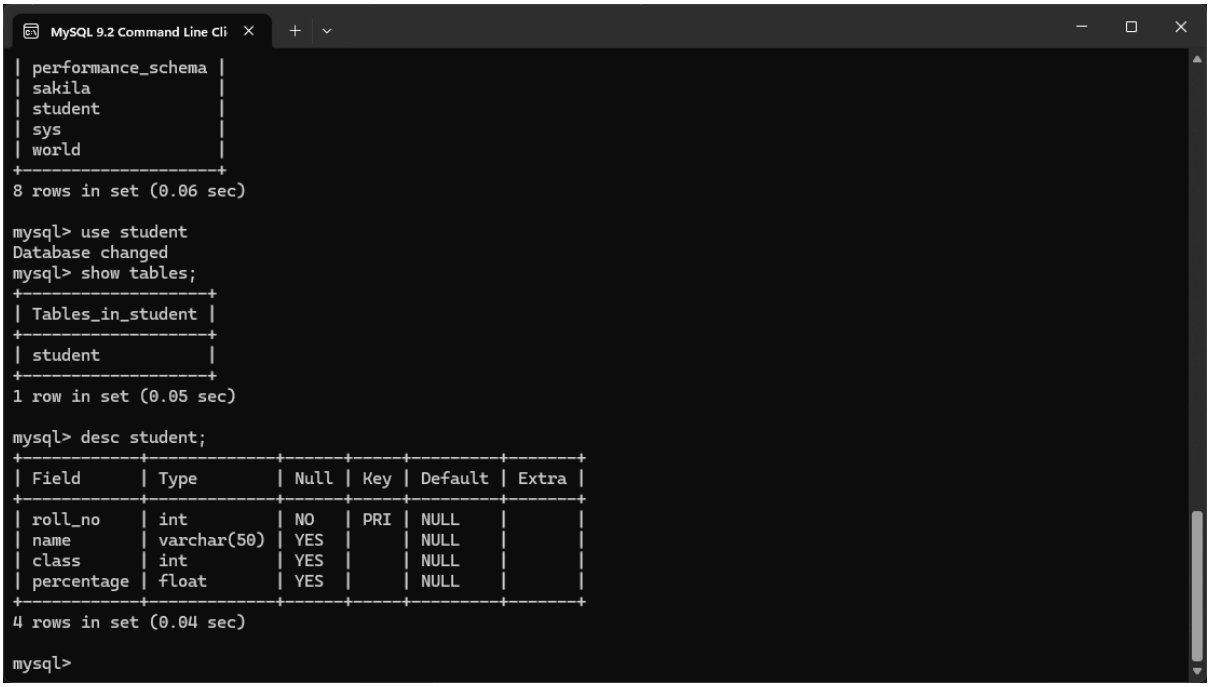
SYNTAX

DESC table_name;

PROGRAM

mysql> DESC student;

OUTPUT



---

6. Write down syntax and MySQL program to set a default value for a field in a table and its OUTPUT.

AIM

To write MySQL code that sets a **default value** for a field in the table STUDENT, showing **SYNTAX**, **PROGRAM**, and **OUTPUT**.

SYNTAX

ALTER TABLE table_name

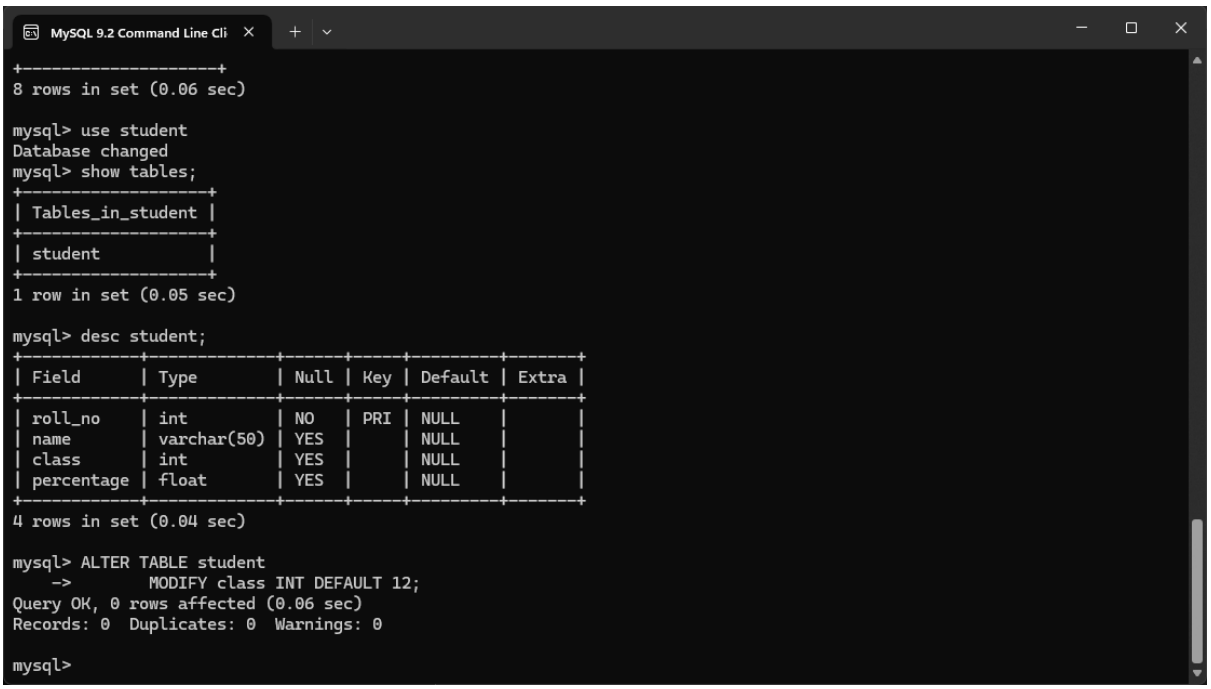
MODIFY column_name datatype DEFAULT default_value;

PROGRAM

mysql> ALTER TABLE student

MODIFY class INT DEFAULT 12;

OUTPUT



---

7. Write down syntax and MySQL program to insert values into a table and its OUTPUT.

AIM

To write MySQL code to **insert records** into the STUDENT table, showing **SYNTAX**, **PROGRAM**, and **OUTPUT**.

SYNTAX

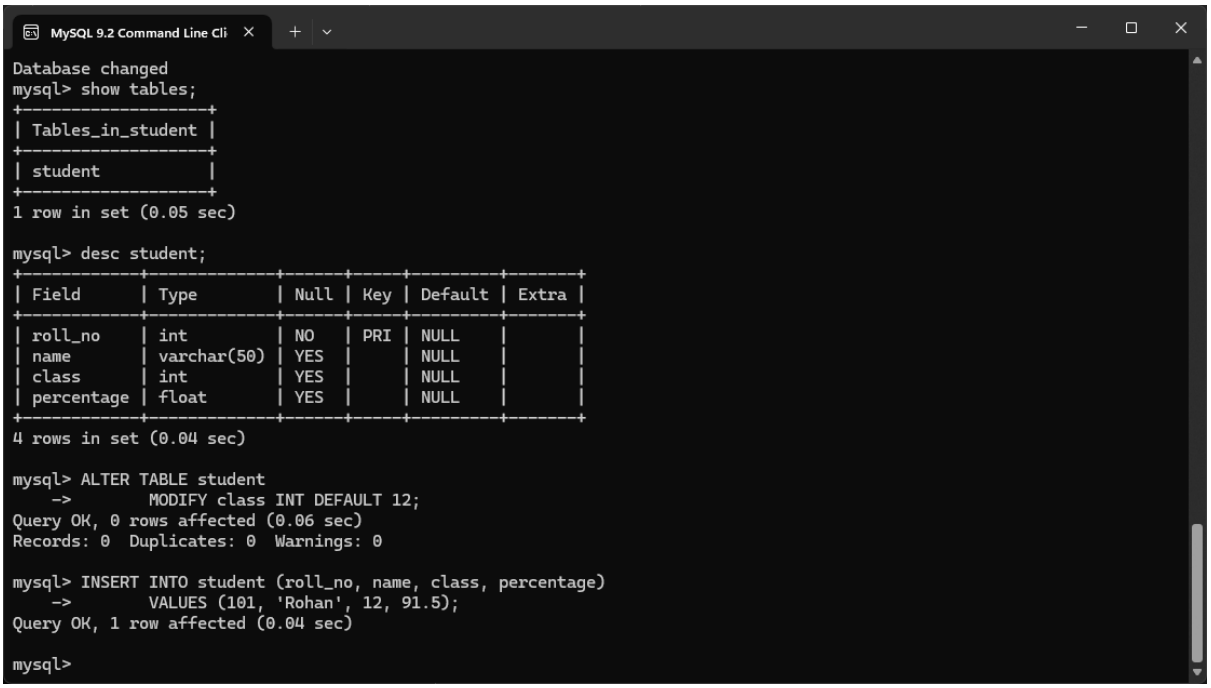
INSERT INTO table_name (column1, column2, ...)

VALUES (value1, value2, ...);

PROGRAM

```
mysql> INSERT INTO student (roll_no, name, class, percentage)
    VALUES (101, 'Rohan', 12, 91.5);
```

OUTPUT



---

8. Write down syntax and MySQL program to display the details of a field in a table based on a given condition and its OUTPUT.

AIM

To write a MySQL command to **retrieve specific records** from the STUDENT table using a **conditional query**, showing **SYNTAX**, **PROGRAM**, and **OUTPUT**.

SYNTAX

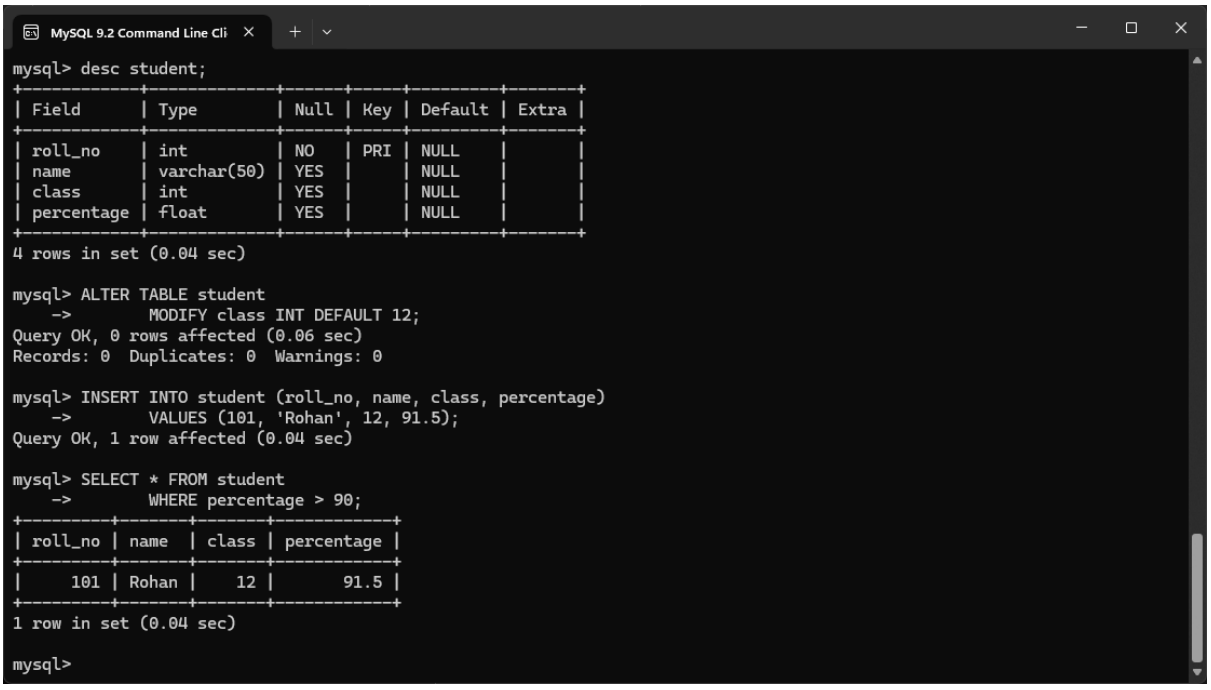
SELECT * FROM table_name

WHERE condition;

PROGRAM

```
mysql> SELECT * FROM student
    WHERE percentage > 90;
```

OUTPUT



---

9. Write down syntax and MySQL program to update the details of a field in a table based on a given condition and its OUTPUT.

AIM

To write a MySQL command to **update data** in the STUDENT table based on a specified condition, showing **SYNTAX**, **PROGRAM**, and **OUTPUT**.

SYNTAX

UPDATE table_name

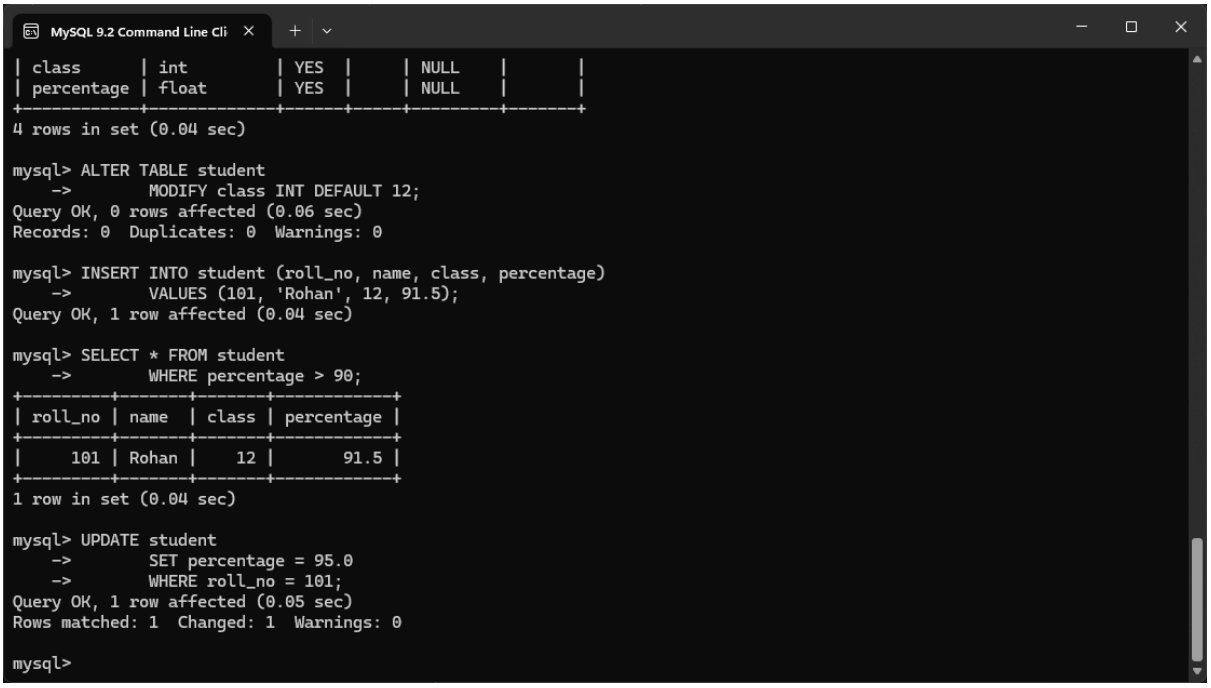
SET column_name = new_value

WHERE condition;

PROGRAM

```
mysql> UPDATE student
    SET percentage = 95.0
    WHERE roll_no = 101;
```

OUTPUT



---

10. Write down syntax and MySQL program to use aggregate functions in MySQL and its OUTPUT.

AIM

To demonstrate the use of **aggregate functions** in MySQL such as COUNT(), AVG(), MAX(), MIN(), and SUM() on the STUDENT table, showing **SYNTAX**, **PROGRAM**, and **OUTPUT**.

SYNTAX

SELECT AGGREGATE_FUNCTION(column_name)

FROM table_name;

PROGRAM

mysql> SELECT COUNT(*) FROM student;

mysql> SELECT AVG(percentage) FROM student;

mysql> SELECT MAX(percentage) FROM student;

mysql> SELECT MIN(percentage) FROM student;

mysql> SELECT SUM(percentage) FROM student;

OUTPUT

---

11. Write SQL syntax and program to add a new column to a table.

AIM

To add a new column gender to an existing table student using ALTER TABLE.

SYNTAX

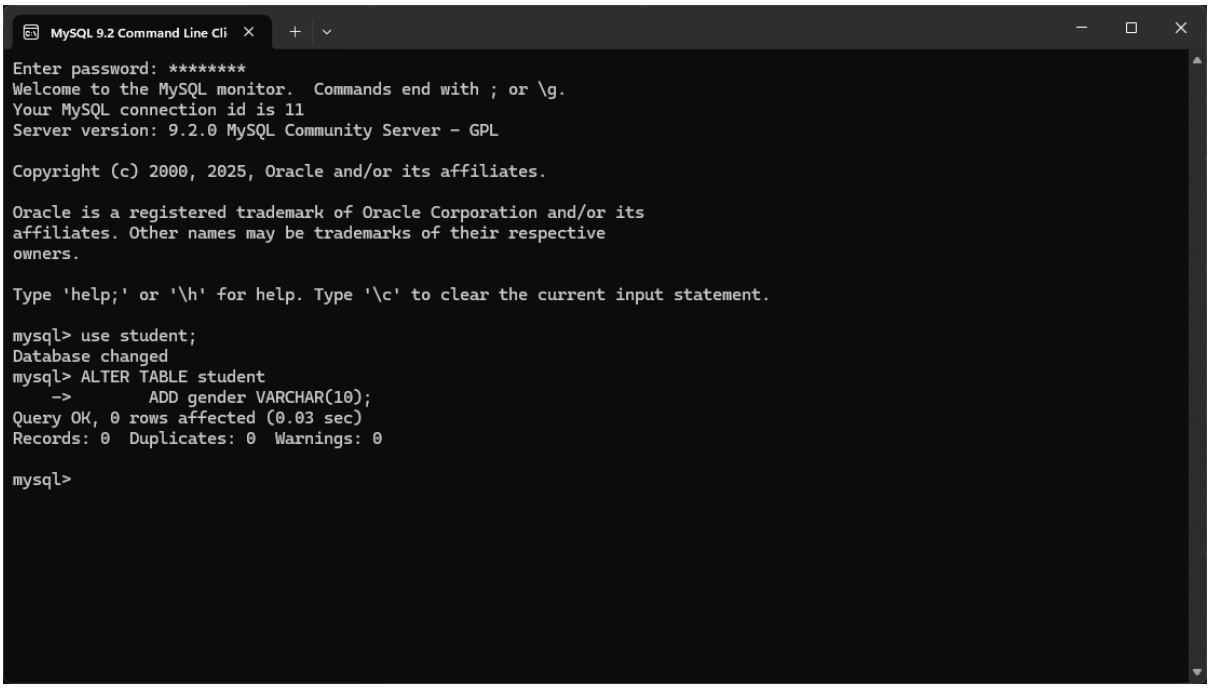
ALTER TABLE table_name

ADD column_name datatype;

PROGRAM

```
mysql> ALTER TABLE student
    ADD gender VARCHAR(10);
```

OUTPUT



---

12. Write SQL syntax and program to delete a specific record from a table.

AIM

To remove a particular record using the DELETE command based on a condition.

SYNTAX

DELETE FROM table_name

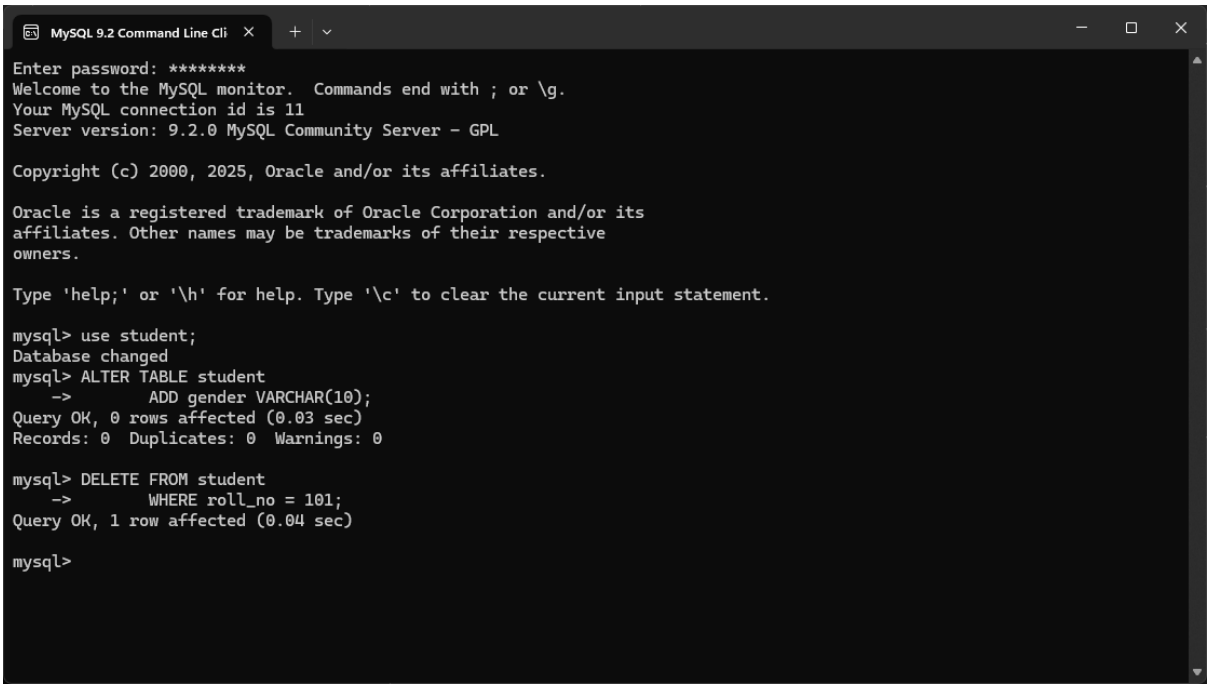
WHERE condition;

PROGRAM

mysql> DELETE FROM student

WHERE roll_no = 101;

OUTPUT



---

13. Write SQL syntax and program to sort data in ascending order.

AIM

To sort the data of a table in **ascending order** using ORDER BY.

SYNTAX

SELECT * FROM table_name

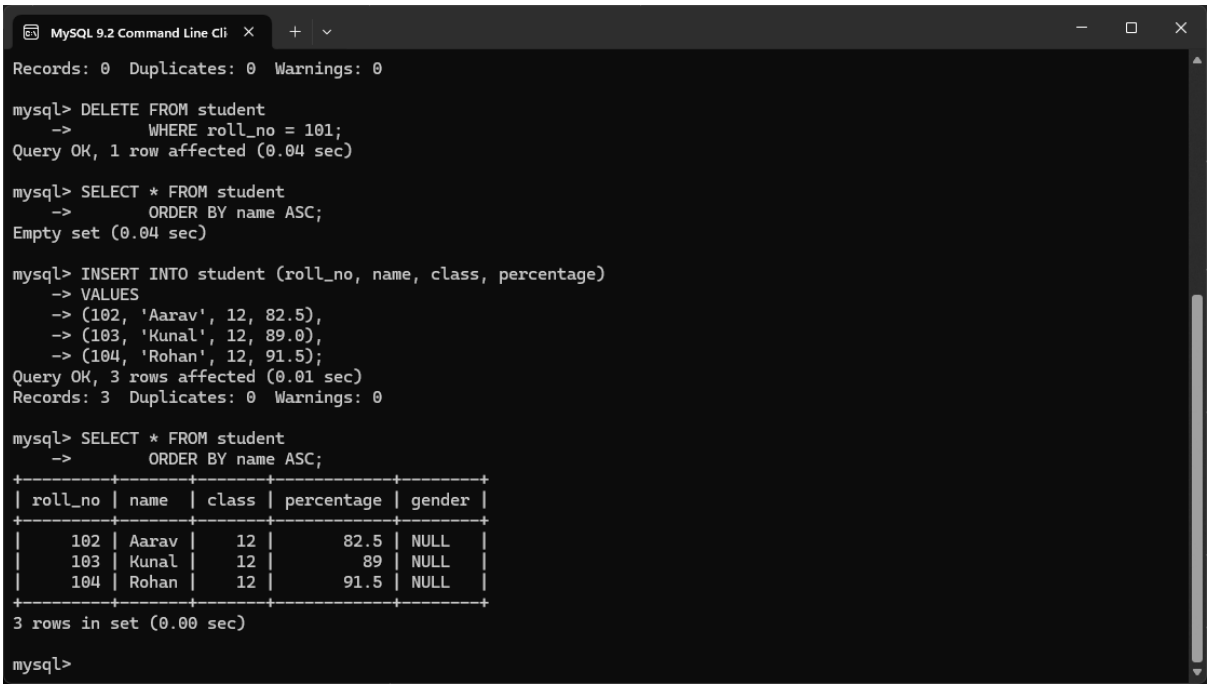
ORDER BY column_name ASC;

PROGRAM

mysql> SELECT * FROM student

ORDER BY name ASC;

OUTPUT



---

14. Write SQL syntax and program to count the number of students.

AIM

To count the total number of records in the student table using the COUNT() aggregate function.

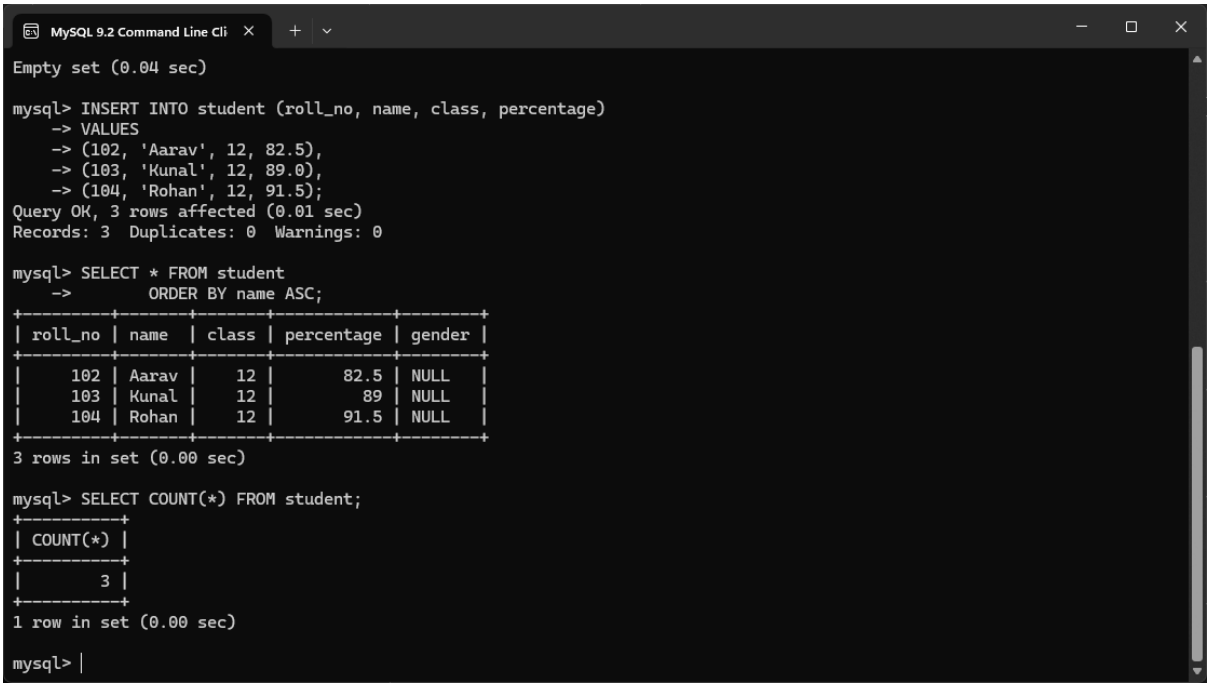
SYNTAX

SELECT COUNT(*) FROM table_name;

PROGRAM

mysql> SELECT COUNT(*) FROM student;

OUTPUT



---

15. Write SQL syntax and program to rename a column in a table.

AIM

To rename the column name to full_name in the student table using ALTER TABLE.

SYNTAX

ALTER TABLE table_name

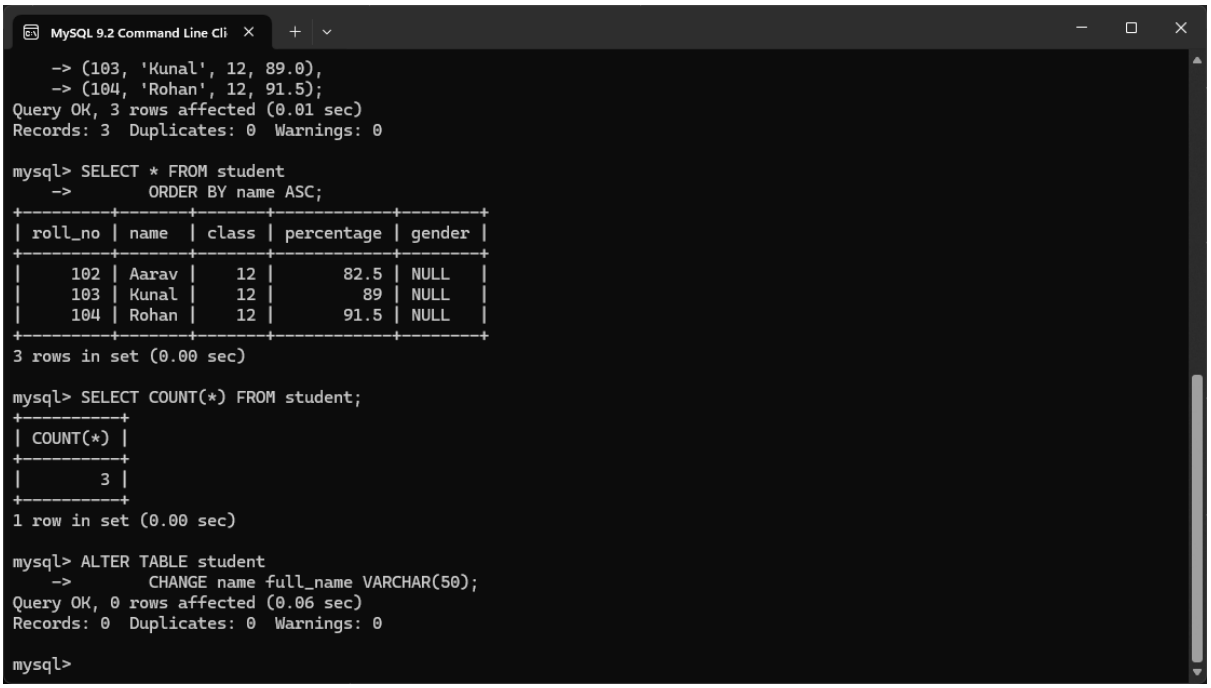
CHANGE old_column_name new_column_name datatype;

PROGRAM

mysql> ALTER TABLE student

CHANGE name full_name VARCHAR(50);

OUTPUT



JAVA PROGRAMS

---

1. Write a Java program to print “Hello World”.

AIM

To write a simple Java program that displays the message **"Hello World"** on the screen.

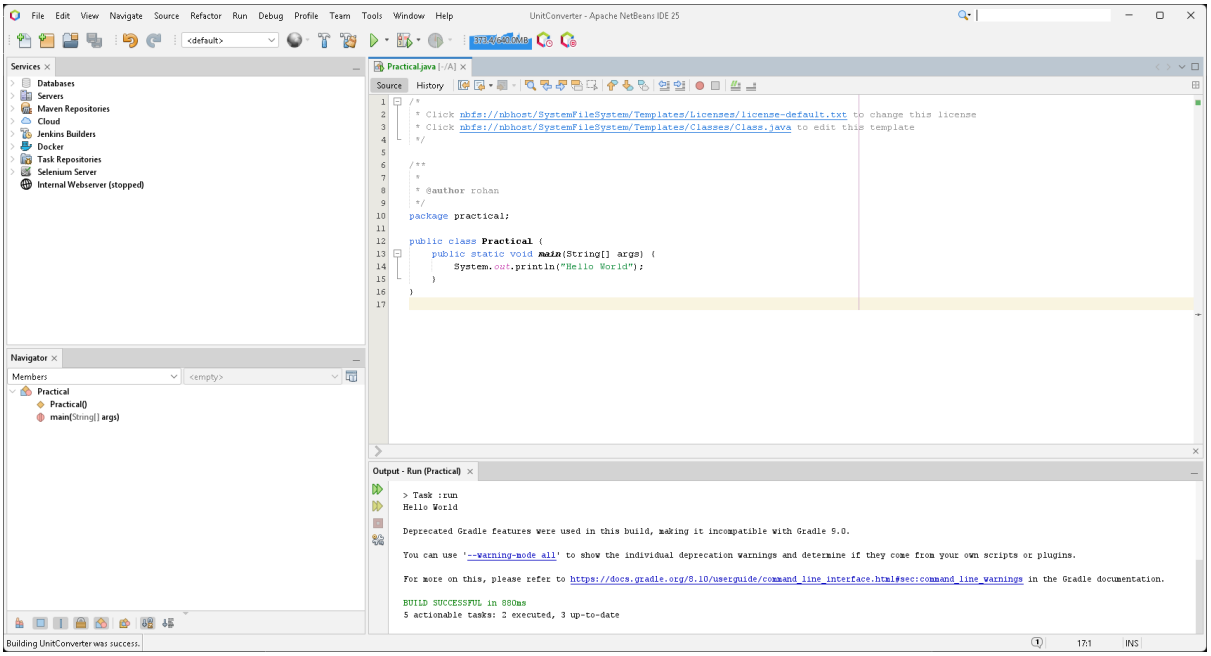
SYNTAX

```
public class ClassName {
   public static void main(String[] args) {
      // Statement(s)
   }
}
```

PROGRAM

```
public class Practical {
   public static void main(String[] args) {
      System.out.println("Hello World");
   }
}
```

OUTPUT



---

2. Write a Java program to display characters A to Z using a for loop.

AIM

To write a Java program that prints the **uppercase English alphabets (A to Z)** using a for loop.

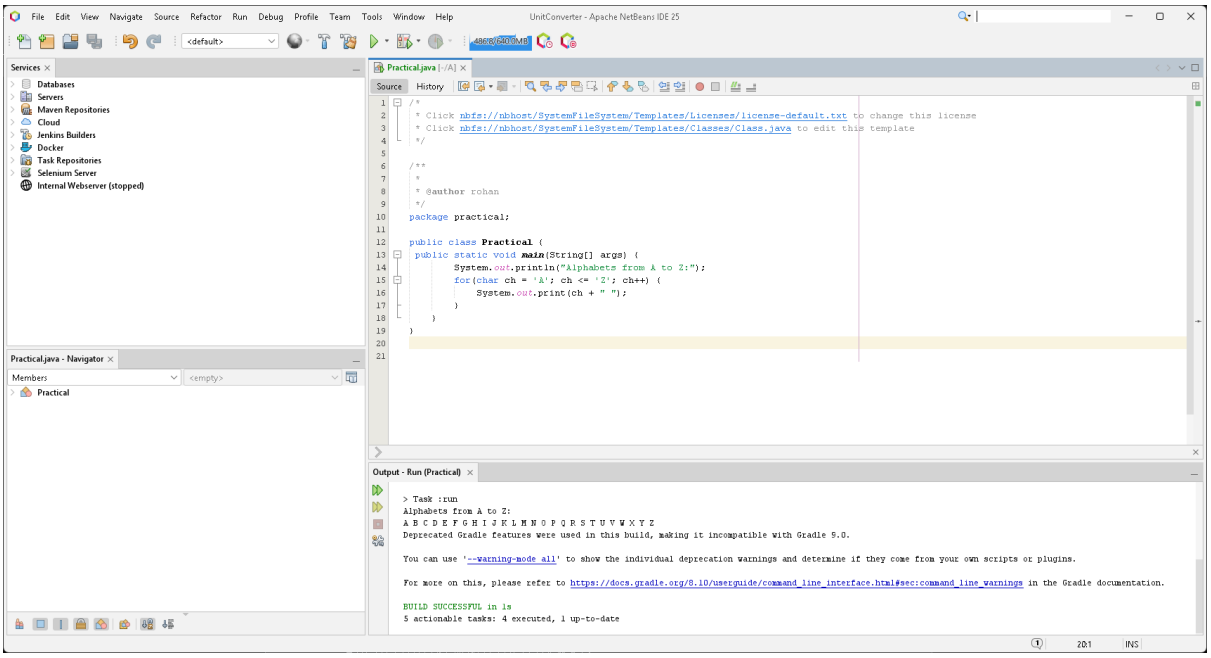
SYNTAX

```
for(char ch = 'A'; ch <= 'Z'; ch++) {
   // print character
}
```

PROGRAM

```
public class Practical {
   public static void main(String[] args) {
      System.out.println("Alphabets from A to Z:");
      for(char ch = 'A'; ch <= 'Z'; ch++) {
         System.out.print(ch + " ");
      }
   }
}
```

OUTPUT



---

3. Write a Java program to check whether a number is Positive, Negative, or Zero using if...else.

AIM

To write a Java program that determines whether the input number is **positive**, **negative**, or **zero** using **conditional statements**.

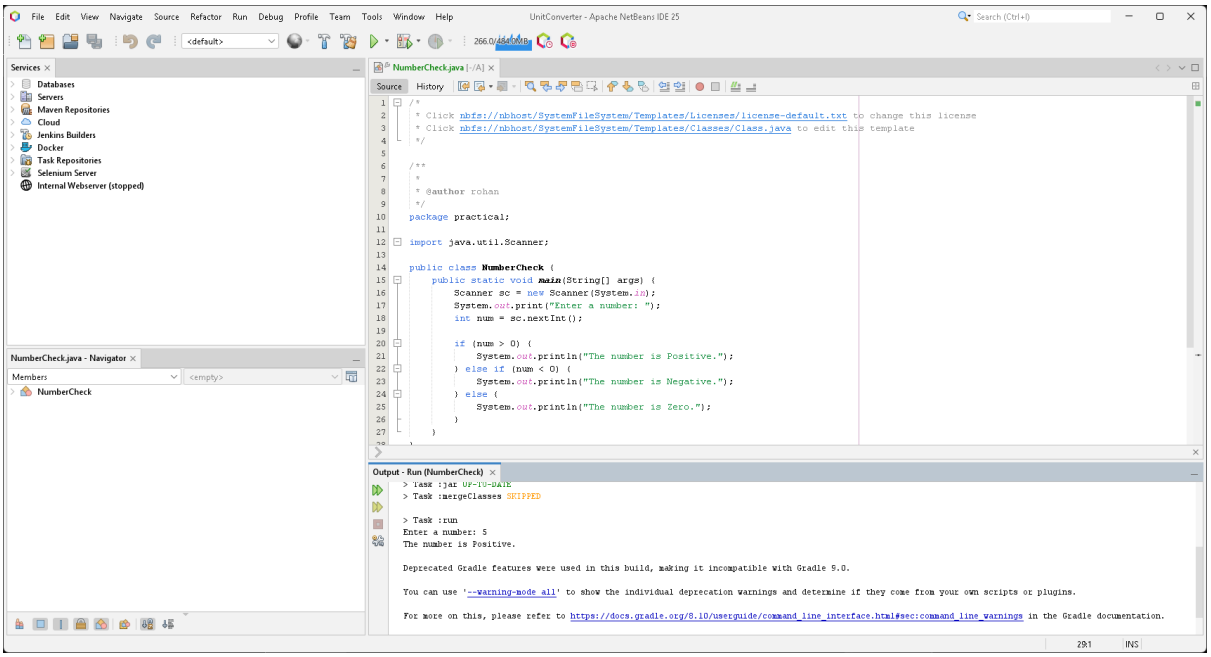
SYNTAX

```
if (number > 0)
   // positive
else if (number < 0)
   // negative
else
   // zero
```

PROGRAM

```
import java.util.Scanner;
public class NumberCheck {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = sc.nextInt();
      if (num > 0) {
         System.out.println("The number is Positive.");
      } else if (num < 0) {
         System.out.println("The number is Negative.");
      } else {
         System.out.println("The number is Zero.");
      }
   }
}
```

OUTPUT ***(Sample Input: 5)***



---

4. Write a Java program to find the largest among three numbers using if...else.

AIM

To write a Java program that accepts three numbers as input and determines the **largest** among them using **nested if...else statements**.

SYNTAX

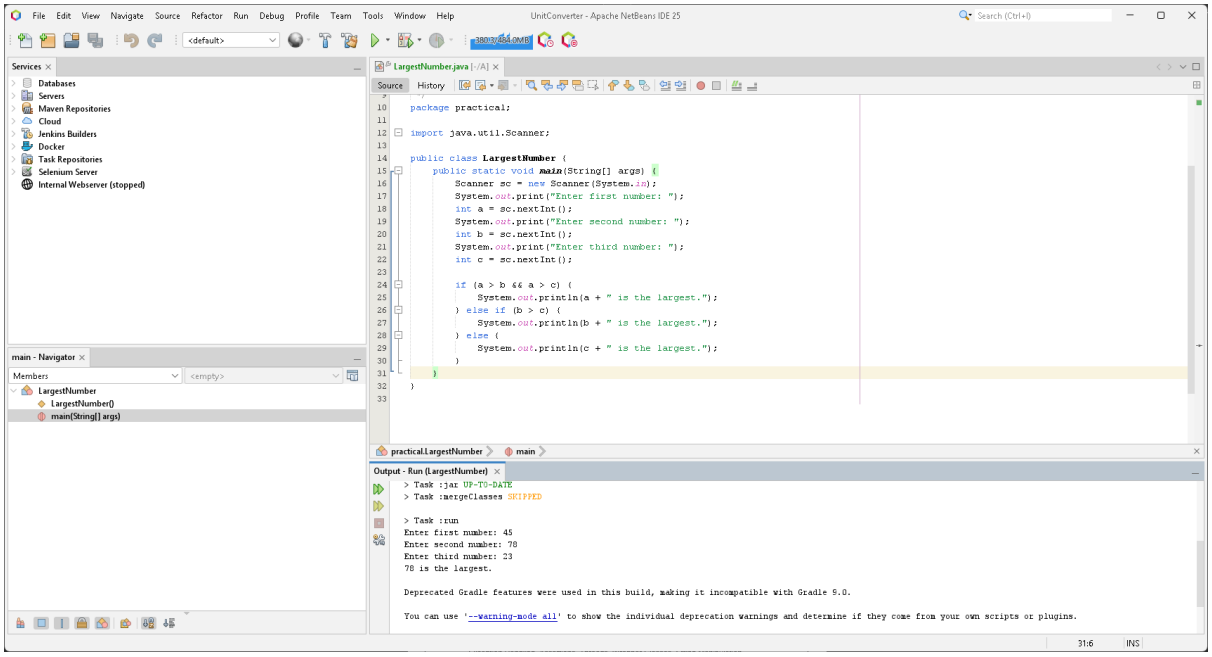
```
if (a > b && a > c)
   // a is largest
else if (b > c)
   // b is largest
else
   // c is largest
```

PROGRAM

```
import java.util.Scanner;


public class LargestNumber {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter first number: ");
      int a = sc.nextInt();
```

```
        System.out.print("Enter second number: ");
        int b = sc.nextInt();
        System.out.print("Enter third number: ");
        int c = sc.nextInt();

        if (a > b && a > c) {
            System.out.println(a + " is the largest.");
        } else if (b > c) {
            System.out.println(b + " is the largest.");
        } else {
            System.out.println(c + " is the largest.");
        }
    }
}
```

OUTPUT *(Sample Input: 45, 78, 23)*



---

5. Write a Java program to check whether a number is Even or Odd using if...else.

AIM

To write a Java program that determines whether an **input number** is **even** or **odd** using the modulo (%) operator and if...else condition.

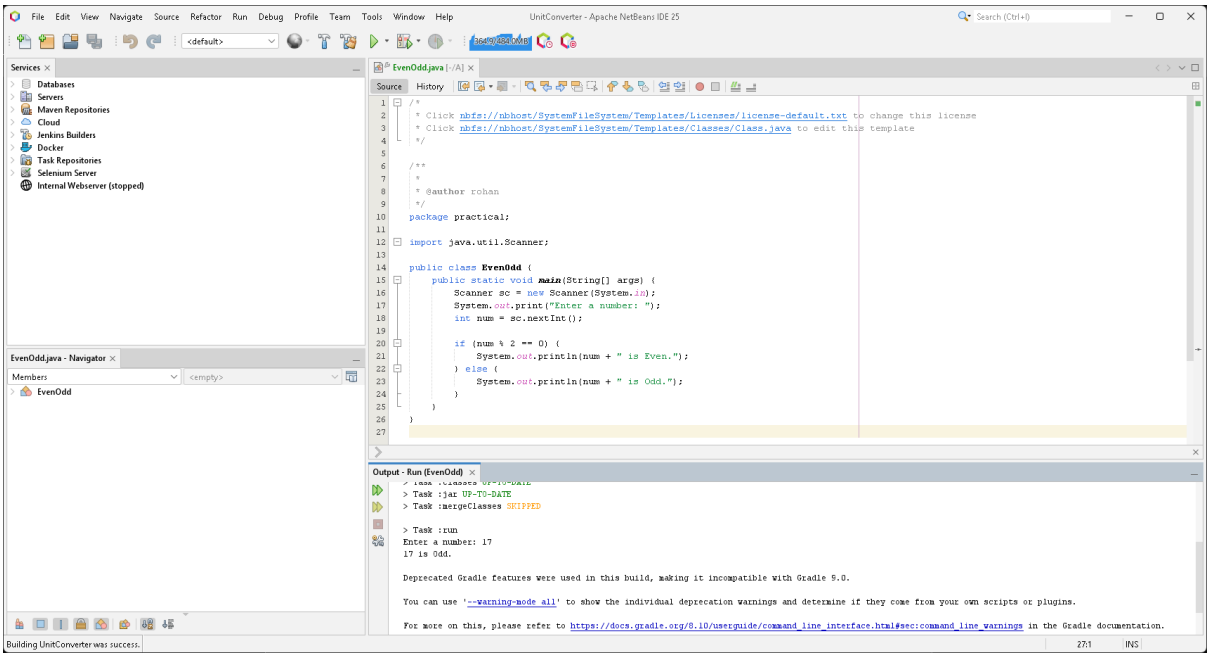
SYNTAX

```
if (number % 2 == 0)
   // Even
else
   // Odd
```

PROGRAM

```
import java.util.Scanner;
public class EvenOdd {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = sc.nextInt();
      if (num % 2 == 0) {
         System.out.println(num + " is Even.");
      } else {
         System.out.println(num + " is Odd.");
      }
   }
}
```

OUTPUT *(Sample Input: 17)*



---

6. Write a Java program to generate a multiplication table using a for loop.

AIM

To write a Java program that **prints the multiplication table** of a given number (from 1 to 10) using a for loop.

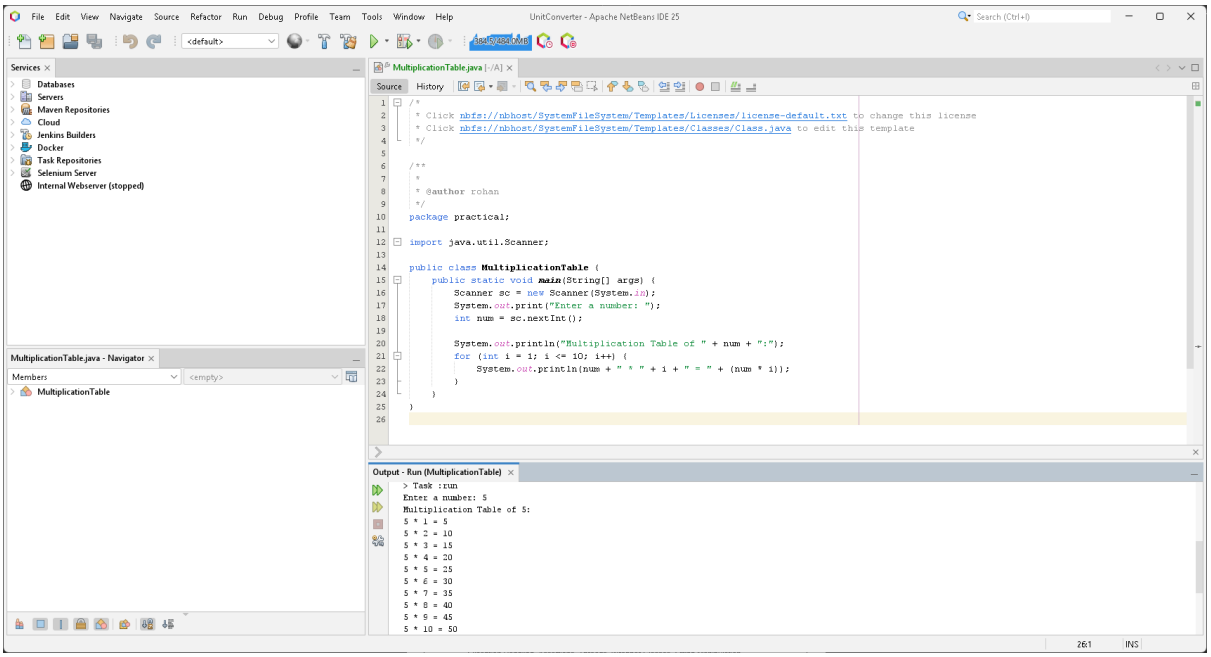
SYNTAX

```
for (int i = 1; i <= 10; i++) {
   System.out.println(number + " * " + i + " = " + (number * i));
}
```

PROGRAM

```
import java.util.Scanner;


public class MultiplicationTable {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = sc.nextInt();


      System.out.println("Multiplication Table of " + num + ":");
      for (int i = 1; i <= 10; i++) {
         System.out.println(num + " * " + i + " = " + (num * i));
      }
   }
}
```

OUTPUT *(Sample Input: 5)*



---

7. Write a Java program to find the area of a rectangle by taking length and breadth as input.

AIM

To write a Java program that calculates the **area of a rectangle** using the formula Area = length × breadth based on **user input**.

SYNTAX

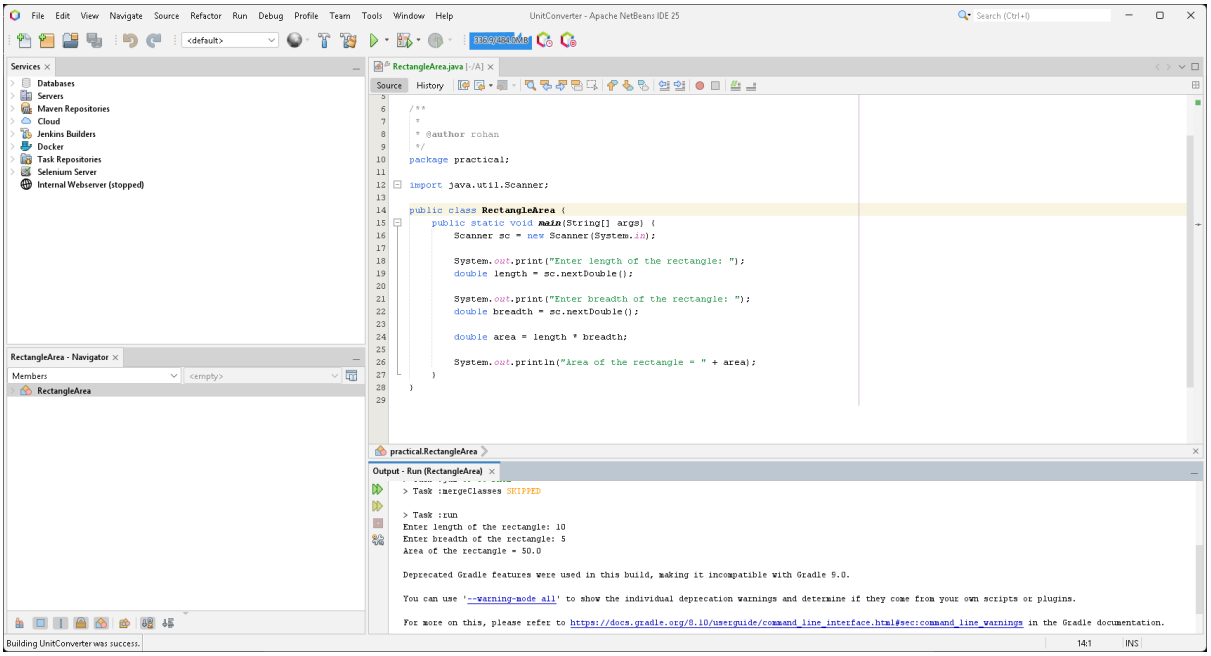
area = length * breadth;

PROGRAM

```
import java.util.Scanner;


public class RectangleArea {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);


      System.out.print("Enter length of the rectangle: ");
      double length = sc.nextDouble();


      System.out.print("Enter breadth of the rectangle: ");
      double breadth = sc.nextDouble();


      double area = length * breadth;
```

```
        System.out.println("Area of the rectangle = " + area);
    }
}
```

OUTPUT *(Sample Input: length = 10, breadth = 5)*



---

8. Write a Java program to print the sum of the first 10 natural numbers.

AIM

To write a Java program that computes and displays the **sum of the first 10 natural numbers** using a for loop.

SYNTAX

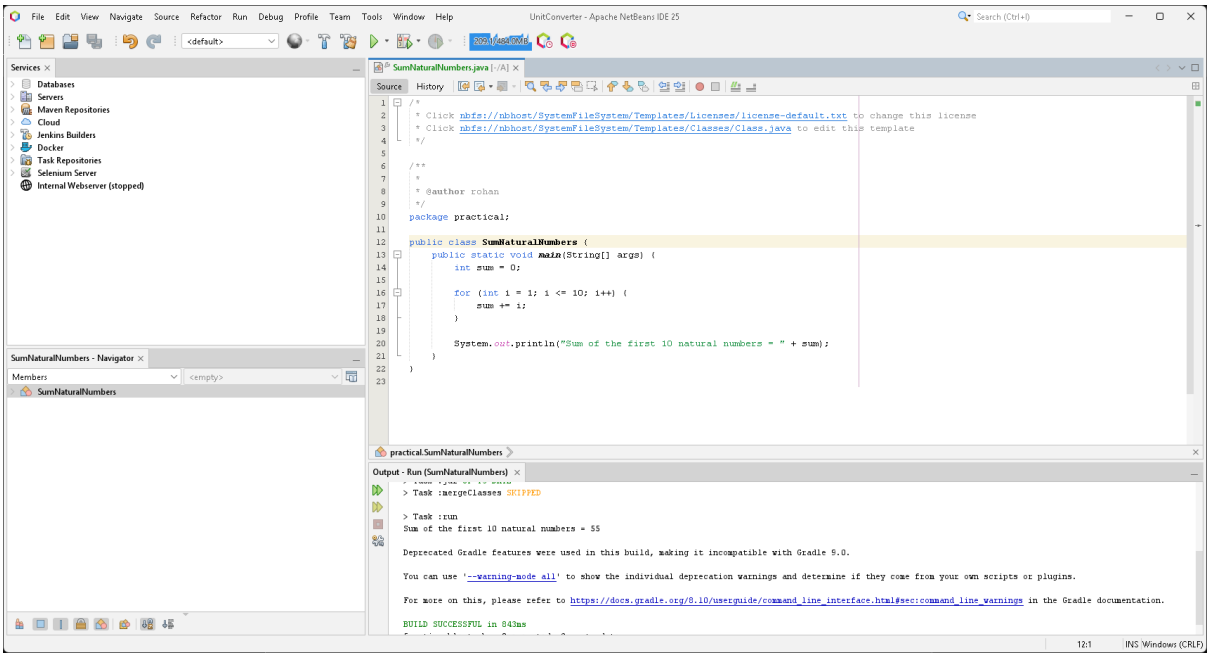
```
for (int i = 1; i <= 10; i++) {
    sum += i;
}
```

PROGRAM

```
public class SumNaturalNumbers {
    public static void main(String[] args) {
        int sum = 0;

        for (int i = 1; i <= 10; i++) {
```

```
        sum += i;
    }

    System.out.println("Sum of the first 10 natural numbers = " + sum);
  }
}
```

OUTPUT



---

9. Write a Java program to generate Fibonacci series based on user input.

AIM

To write a Java program that **generates the Fibonacci series** up to a specified number of terms provided by the user.

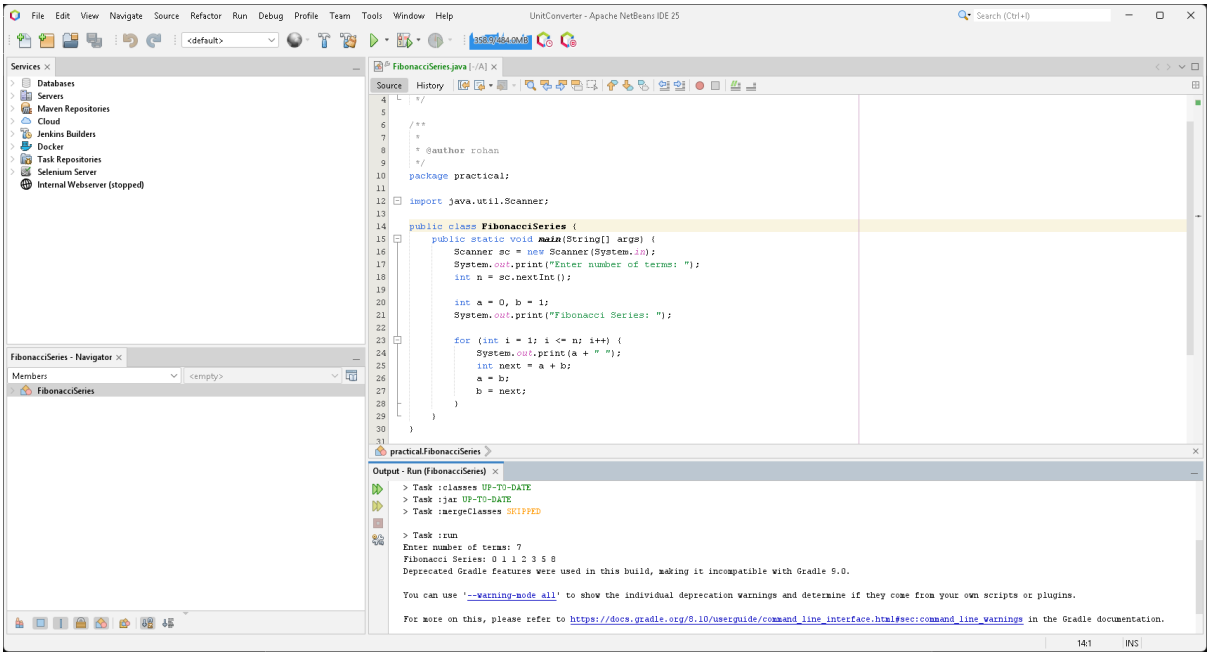
SYNTAX

```
int a = 0, b = 1;
for (int i = 1; i <= n; i++) {
  // print a
  int next = a + b;
  a = b;
  b = next;
}
```

PROGRAM

```
import java.util.Scanner;


public class FibonacciSeries {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter number of terms: ");
      int n = sc.nextInt();

      int a = 0, b = 1;
      System.out.print("Fibonacci Series: ");

      for (int i = 1; i <= n; i++) {
         System.out.print(a + " ");
         int next = a + b;
         a = b;
         b = next;
      }
   }
}
```

OUTPUT *(Sample Input: 7)*



10. Write a Java program to print the factorial of a given number.

AIM

To write a Java program that calculates the **factorial** of a number entered by the user using a for loop.

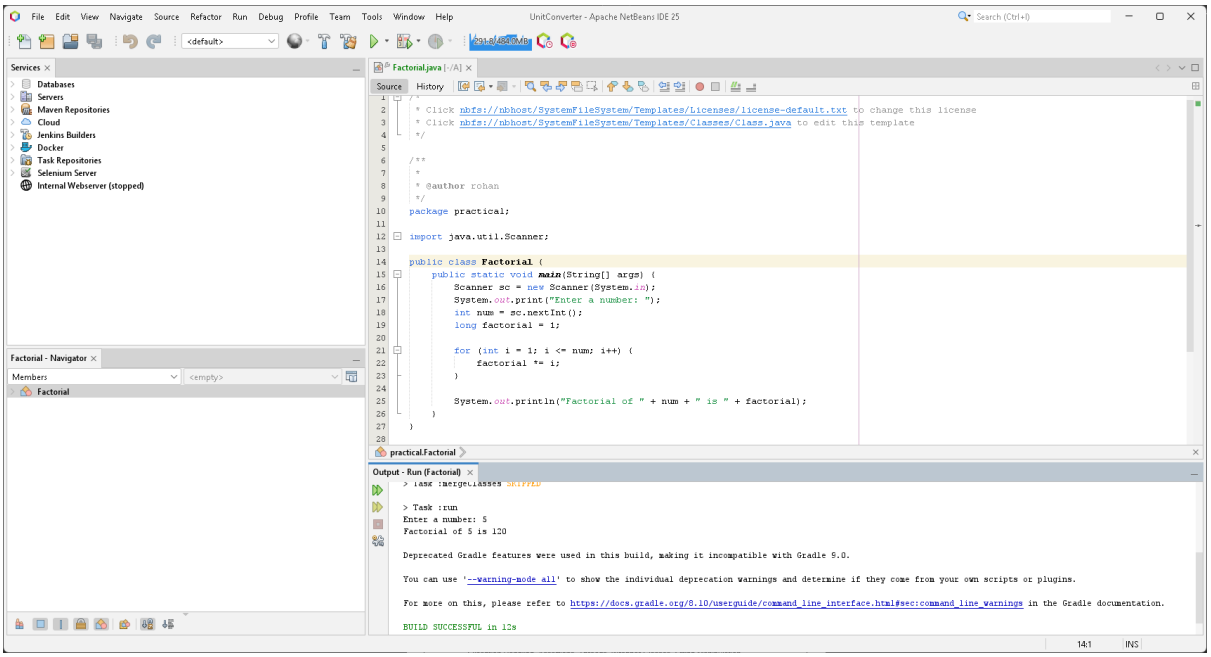
SYNTAX

```
factorial = 1;
for (int i = 1; i <= number; i++) {
   factorial *= i;
}
```

PROGRAM

```
import java.util.Scanner;

public class Factorial {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = sc.nextInt();
      long factorial = 1;

      for (int i = 1; i <= num; i++) {
         factorial *= i;
      }

      System.out.println("Factorial of " + num + " is " + factorial);
   }
}
```

OUTPUT *(Sample Input: 5)*



---

11. Write a Java program to calculate the sum of digits of a number.

AIM

To find the sum of individual digits of a number using a loop.

SYNTAX

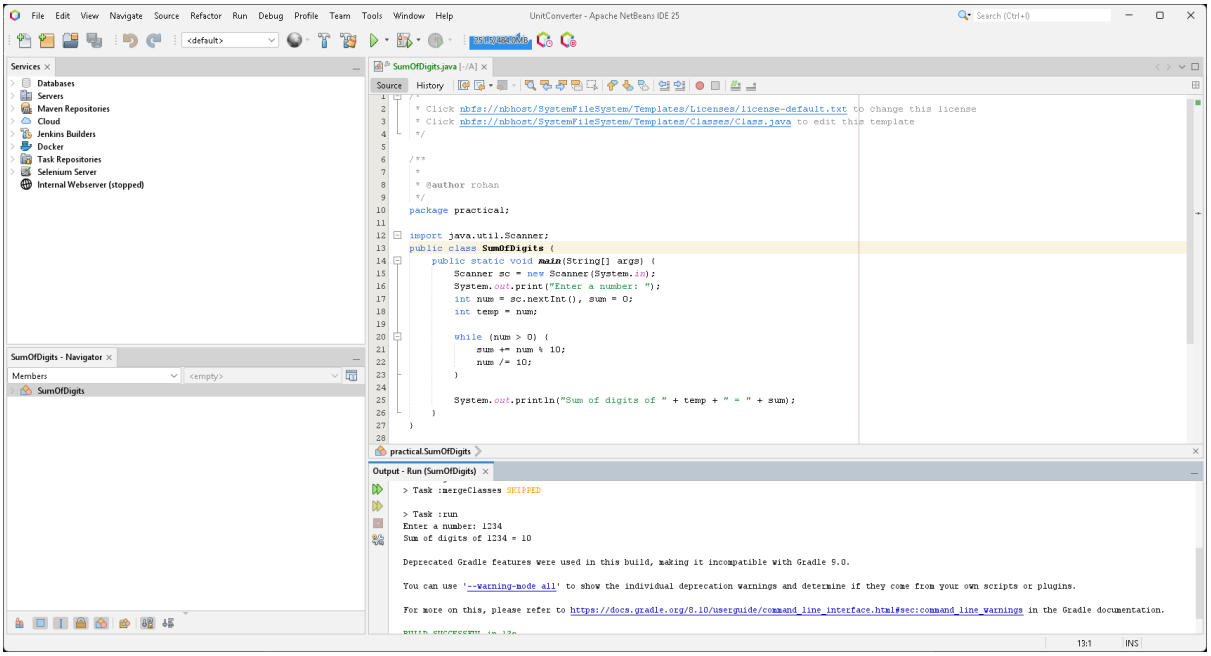
```
while (num > 0) {
   sum += num % 10;
   num /= 10;
}
```

PROGRAM

```
import java.util.Scanner;
public class SumOfDigits {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int num = sc.nextInt(), sum = 0;
      int temp = num;

      while (num > 0) {
         sum += num % 10;
         num /= 10;
      }
```

```
        System.out.println("Sum of digits of " + temp + " = " + sum);
    }
}
```

OUTPUT



---

12. Write a Java program to check whether a number is prime.

AIM

To determine whether a number is prime using a loop and conditions.

SYNTAX

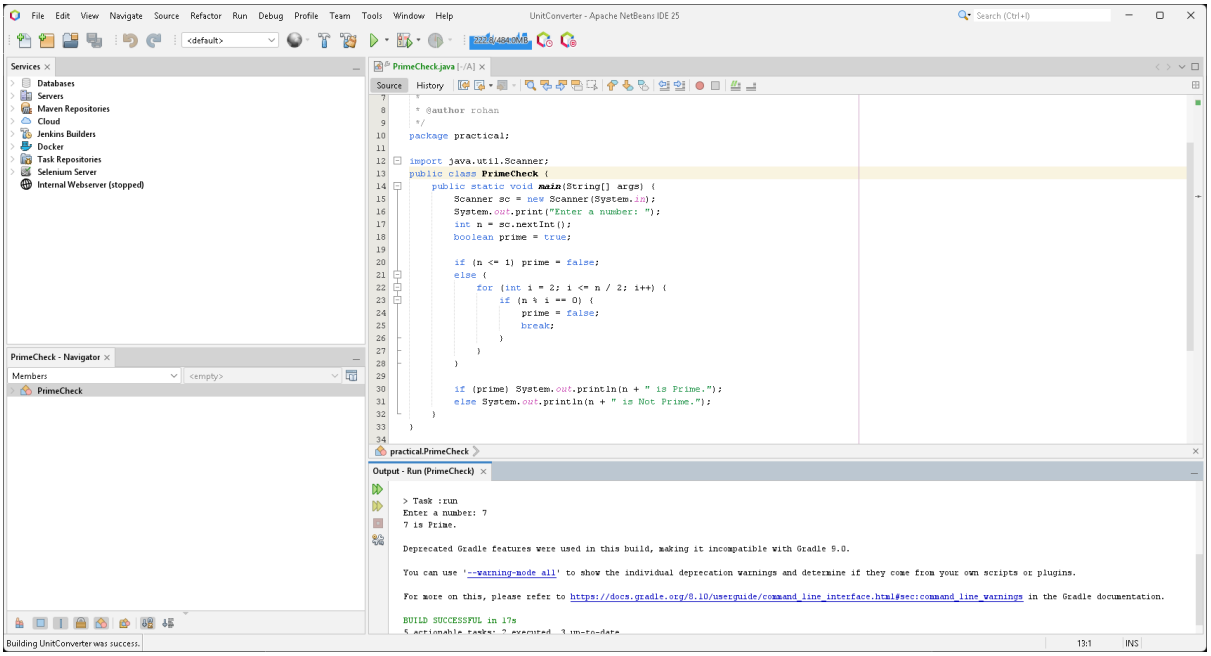
for (i = 2; i <= n/2; i++) { if (n % i == 0) not prime; }

PROGRAM

```
import java.util.Scanner;
public class PrimeCheck {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int n = sc.nextInt();
        boolean prime = true;

        if (n <= 1) prime = false;
        else {
```

```
        for (int i = 2; i <= n / 2; i++) {
            if (n % i == 0) {
                prime = false;
                break;
            }
        }
    }

    if (prime) System.out.println(n + " is Prime.");
    else System.out.println(n + " is Not Prime.");
  }
}
```

OUTPUT



---

13. Write a Java program to reverse a number.

AIM

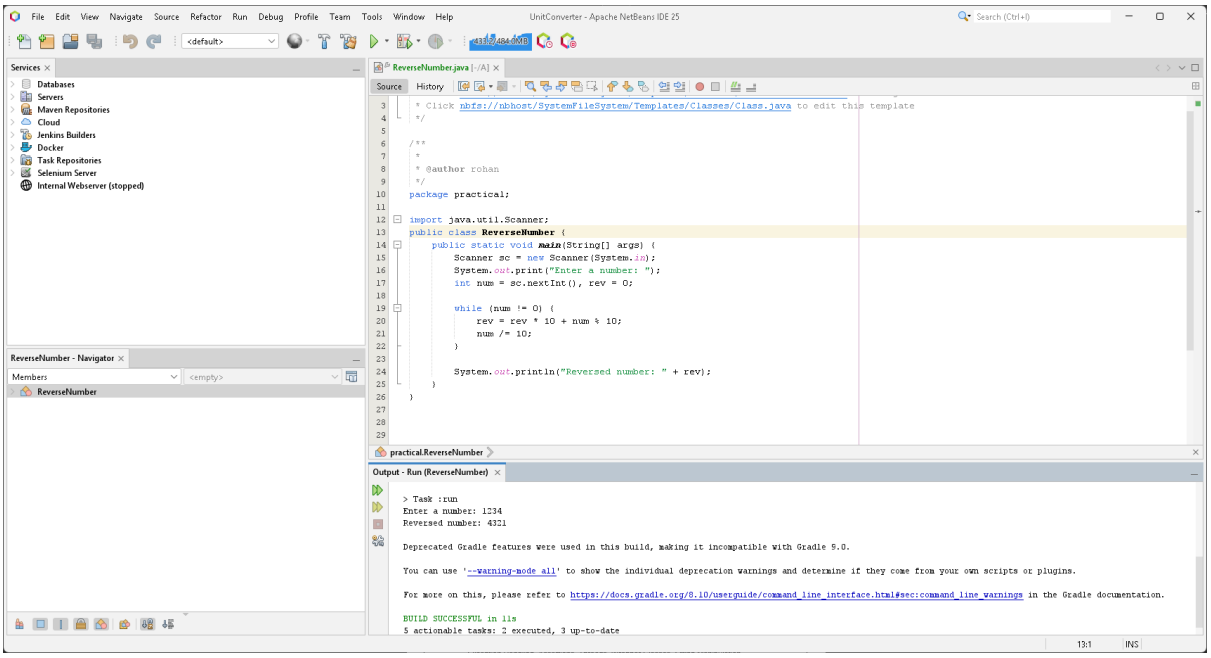
To reverse the digits of a number using a loop.

PROGRAM

```
import java.util.Scanner;

public class ReverseNumber {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
```

```
        System.out.print("Enter a number: ");
        int num = sc.nextInt(), rev = 0;

        while (num != 0) {
            rev = rev * 10 + num % 10;
            num /= 10;
        }

        System.out.println("Reversed number: " + rev);
    }
}
```

OUTPUT



---

14. Write a Java program to check whether a string is a palindrome.

AIM

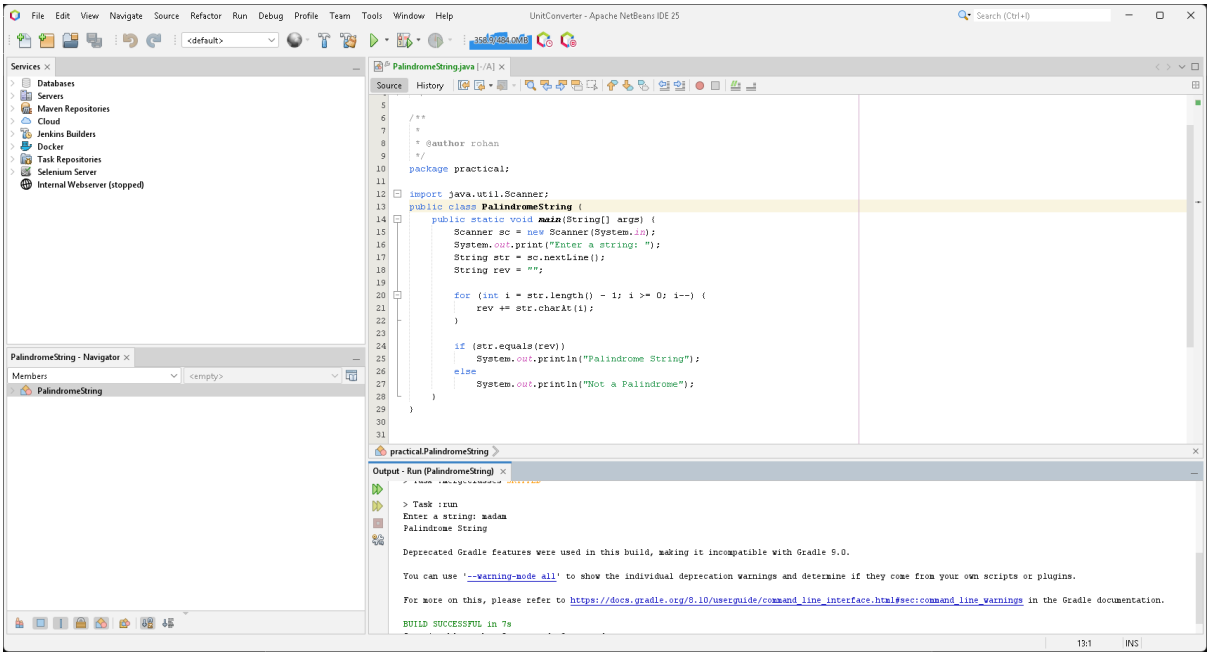
To verify if a string is the same forward and backward.

PROGRAM

```
import java.util.Scanner;

public class PalindromeString {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
```

```
        String str = sc.nextLine();
        String rev = "";

        for (int i = str.length() - 1; i >= 0; i--) {
            rev += str.charAt(i);
        }

        if (str.equals(rev))
            System.out.println("Palindrome String");
        else
            System.out.println("Not a Palindrome");
    }
}
```

OUTPUT



---

15. Write a Java program to count vowels in a string.

AIM

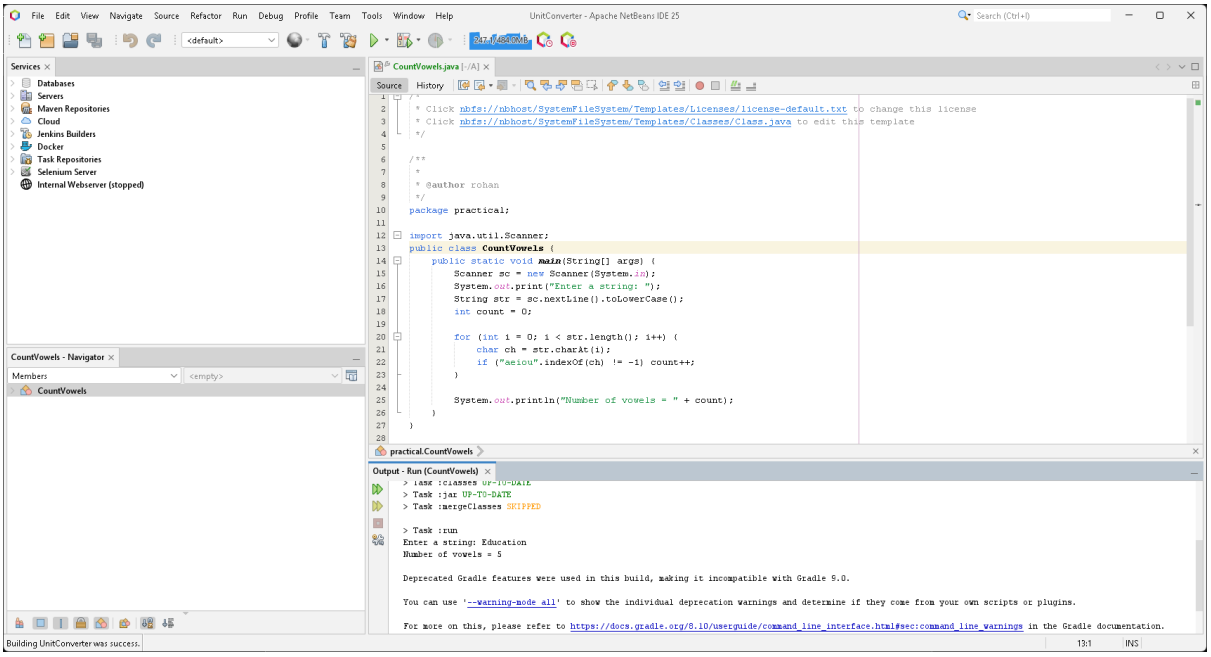
To count the total number of vowels in a given string.

PROGRAM

```
import java.util.Scanner;

public class CountVowels {

    public static void main(String[] args) {
```

```
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = sc.nextLine().toLowerCase();
        int count = 0;

        for (int i = 0; i < str.length(); i++) {
            char ch = str.charAt(i);
            if ("aeiou".indexOf(ch) != -1) count++;
        }

        System.out.println("Number of vowels = " + count);
    }
}
```

OUTPUT



OPERATING WEB BASED APPLICATION

Case Study 1: Online Bill Calculator / Book Rail Ticket

AIM

To explore and demonstrate the use of **web-based utility applications** such as **Online Electricity Bill Calculators** or **Online Railway Ticket Booking Systems**, enabling users to calculate fares, charges, or book services online.

DESCRIPTION

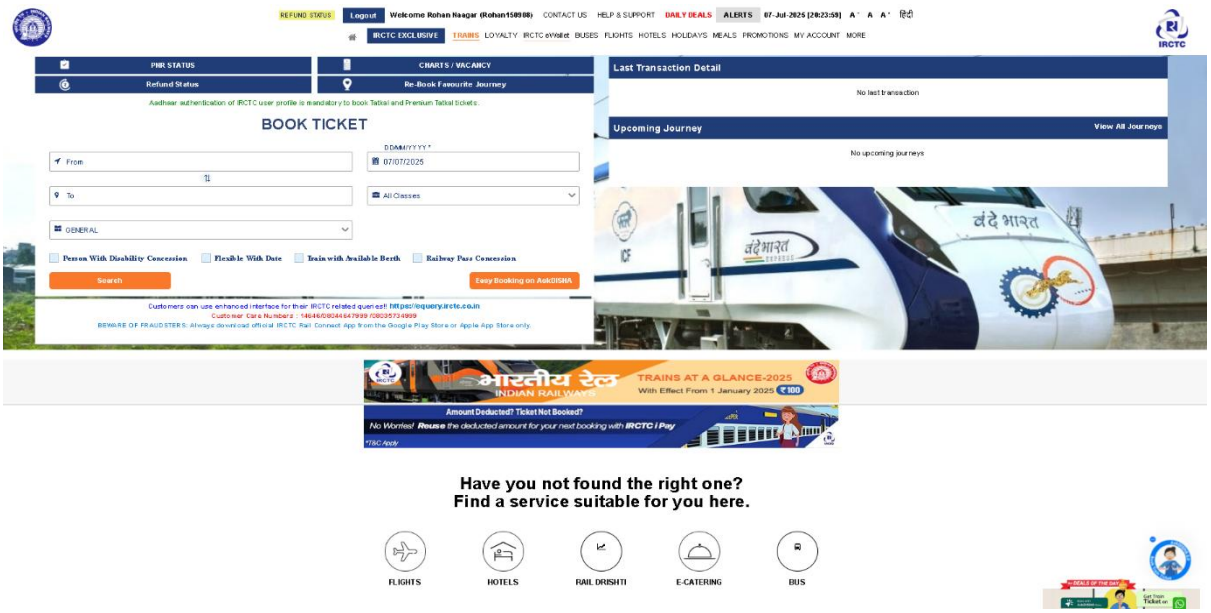
Web-based billing and booking systems are used by government and private service providers. These platforms allow users to:

- Input parameters such as units consumed, class of travel, distance, etc.
- Automatically calculate charges/tickets in real-time.
- Receive confirmation through web forms, email/SMS, or e-payment integration.

TOOLS USED

- **Web Browser** (Chrome / Firefox)
- **Official Web Portals** (IRCTC, State Electricity Boards, etc.)
- **Internet Connectivity**

OUTPUT



Conclusion

Web-based utilities simplify real-world services, save time, ensure accuracy, and reduce human error.

Case Study 2: Online Quiz – Internet and Phone Safety for School Children

---

AIM

To use an **online quiz platform** to educate school children about **internet safety and phone security**, and to test their awareness.

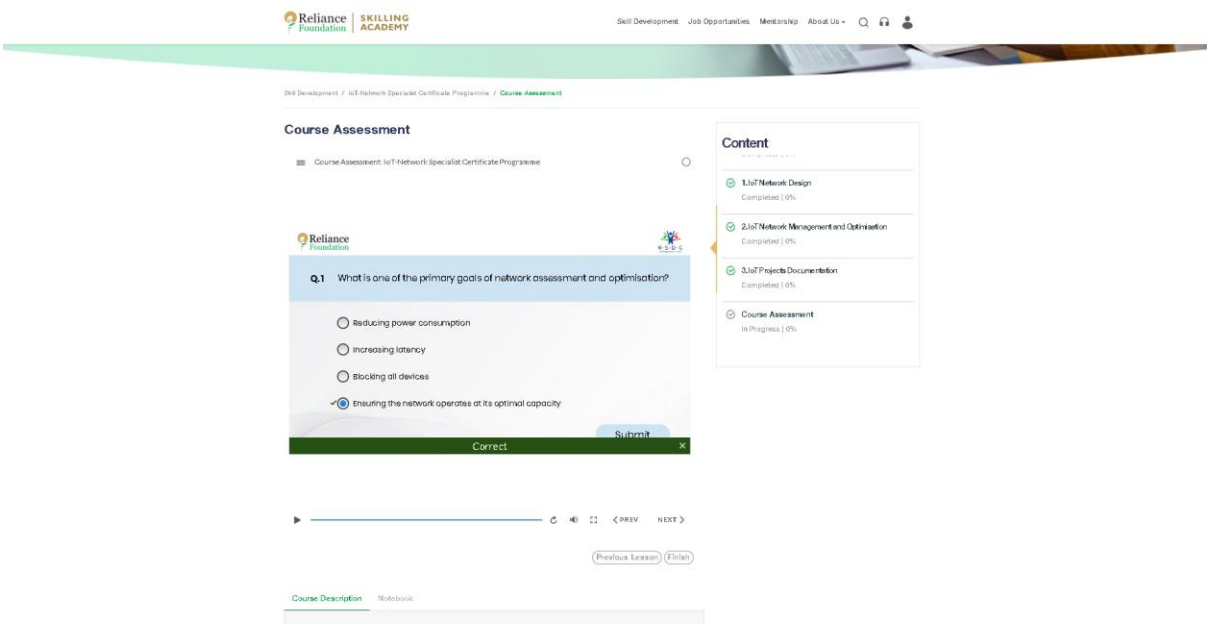
DESCRIPTION

- The quiz includes topics like:
    - Strong password creation
    - Avoiding suspicious links
    - Not sharing OTPs
    - Cyberbullying awareness
- Students access the quiz via platforms such as **Google Forms**, **Kahoot**, or **School LMS portals**.

TOOLS USED

- Online Quiz Tools (Google Forms, Kahoot, etc.)
- Web Browser
- Internet Access

OUTPUT



Conclusion

The quiz helps improve awareness about cyber safety and responsible internet usage among students.

Case Study 3: Online Game – Hangman

AIM

To use a **web-based game** to understand the logic and programming principles behind classic word games like **Hangman**.

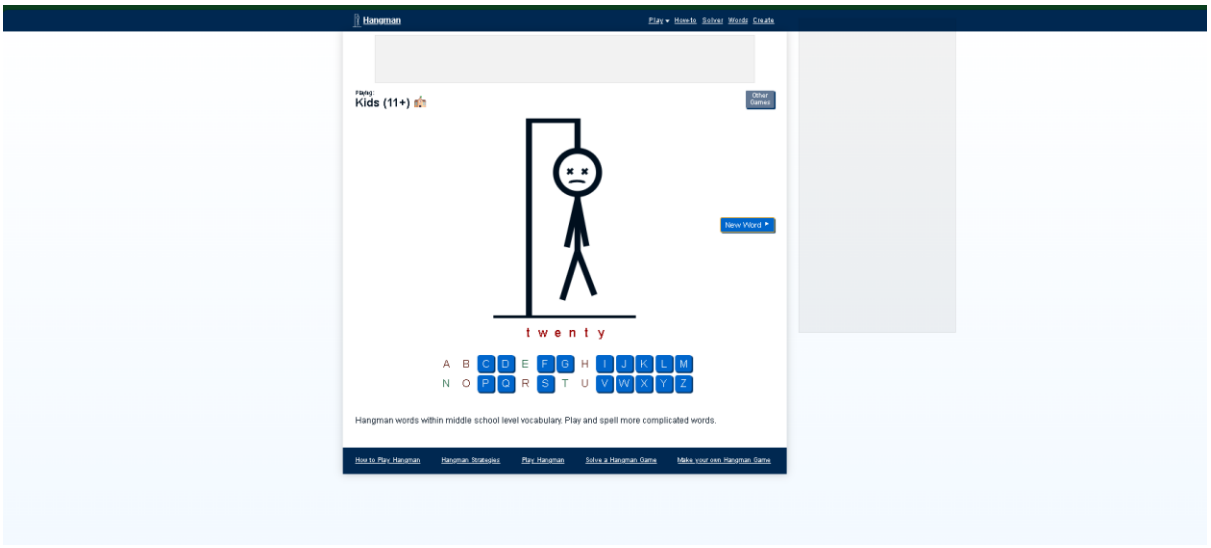
DESCRIPTION

- The game involves guessing a word one letter at a time.
- Each incorrect guess reveals a part of the hangman graphic.
- Educational value: vocabulary building, logical reasoning.

TOOLS USED

- Online platforms like https://www.hangmanwords.com/
- JavaScript or Python (for DIY versions)
- Web Browser

OUTPUT



Conclusion

Online games like Hangman are not only entertaining but also reinforce logical and language skills using simple web technologies.

BIBLIOGRAPHY

The following sources were referred to during the preparation of this project file:

1. **Textbooks:**
    - NCERT Computer Science Class XII
    - CBSE Informatics Practices Curriculum Guide (2025–26)
2. **Online Resources:**
    - https://www.w3schools.com/sql/
    - https://www.javatpoint.com/java-programs
    - https://docs.oracle.com/javase/
    - https://www.mysql.com/
    - https://www.geeksforgeeks.org/
    - https://www.chatgpt.com/
3. **Software & Tools:**
    - MySQL Server & Workbench
    - Java JDK (version 17 or higher)
    - Visual Studio Code / Notepad++
    - Web Browsers (Chrome, Firefox)
4. **Teacher & Peer Support:**
    - Classroom Lectures
    - Practical Lab Sessions
    - Guidance from subject teacher **Mr. Raju Gupta**

APPENDIX / ANNEXURE

Supporting content and additional information attached with this project:

1. **Screenshots:**
    - MySQL Command Prompt Outputs
    - Java Program Output Snapshots
    - Web Application Interfaces (Online Quiz, Bill Calculator, Hangman Game)
2. **Source Code:**
    - Complete Java source code (.java files)
    - SQL script files used in practicals
3. **Test Data & Results:**
    - Sample input/output used for program validation
    - Data sets used in SQL queries (e.g., student table entries)
4. **Certificates & Documentation:**
    - Project Certificate Page
    - Acknowledgement Page
    - CBSE Practical Guidelines (if attached) 802-IT.pdf
