Generic	Write a program that takes 3 numbers as input and finds largest number among them
Generic	1. loops: Print pattern numbers from the N number- pattern - 3,6,9,12,15,18... etc 2. Conditions - Get a number and identify the given number is +ve/-ve/0 3. Conditions - Get a number and identify the number is Even or Odd 4. Write a program that takes two numbers and an operator (+,-,/,*) as inputs and perform corresponding arithmetic operation
Variables/Operators	1. Write a Java program that calculates the area and circumference of a circle given its radius (as input - hardcoded) using appropriate variables and mathematical operators. 2. Create a program that takes two integers as input and swaps their values without using a third variable. Display the original and swapped values.
Simple If	Given current time, write a program to print if it is day time or night. You can consider time greater than 18 as night
If – else	check voter's age eligibilty and declaring age as a constant (18)
if – else ladder	Given an integer,n, perform the following conditional actions: If n is odd, print "Team" If n is even and in the inclusive range of 2 to 5, print "Australia" If n is even and in the inclusive range of 6 to 20, print "Allianz" If n is even and greater than 20 , print "Testing"
Nested If	Given a number. 1. check if the number is lesser than 100. 2. check if the number is greater than 50. 3. If the number is greater than 50 and lesser than 100, then print the message that the entered number is greater than 50 and lesser than 100. 4. If the number is not greater than 50 but is lesser than 100, then print the message that the entered number is lesser than 100
Switch Case	1. Use a switch statement to display the name of the day of the week based on the input (1 for Monday, 2 for Tuesday, etc.). 2. Create a simple calculator program that takes two numbers and an operator (+, -, *, or /) as input. Use a switch statement to perform the corresponding arithmetic operation and display the result.

While loop	1. Print Numbers: Write a program that uses a while loop to print numbers from 1 to 10. 2. Print Even Numbers: Write a program that uses a while loop to print even numbers from 2 to 20. 3. Print Odd Numbers: Write a program that uses a while loop to print odd numbers from 1 to 15. 4. Print Multiplication Table: Write a program that takes an integer as input and uses a while loop to print its multiplication table. 5. Calculate Factorial: Write a program that takes an integer as input and uses a while loop to calculate and print its factorial. 6. Print Reverse: Write a program that takes a string as input and uses a while loop to print it in reverse. 7. Sum of Digits: Write a program that takes an integer as input and uses a while loop to calculate and print the sum of its digits. 8. Power of a Number: Write a program that takes two integers, base and exponent, as input and uses a while loop to calculate and print the result of base raised to the power of exponent.
Do While	1. Print Numbers: Write a program that uses a do-while to print numbers from 1 to 10. 2. Print Even Numbers: Write a program that uses a do-while to print even numbers from 2 to 20. 3. Print Odd Numbers: Write a program that uses a do-while to print odd numbers from 1 to 15. 4. Calculate Sum: Write a program that uses a do-while to continuously take input from the user until the user enters a negative number. Calculate and print the sum of all the positive numbers entered. 5. Guessing Game: Write a program that generates a random number between 1 and 100 and asks the user to guess the number using a do-while. Provide hints whether the guess is too high or too low until the correct number is guessed. 6. Factorial Calculation: Write a program that takes an integer as input and uses a do-while to calculate and print its factorial. 7. Number Reversal: Write a program that takes an integer as input and uses a do-while to reverse its digits and print the result. 8. Menu-Driven Calculator: Write a program that displays a menu with options for addition, subtraction, multiplication, and division. Use a do-while to repeatedly ask the user for choices and perform the selected operation until the user chooses to exit.
for loop	1. Print Numbers: Write a program that uses a for loop to print numbers from 1 to 10. 2. Print Even Numbers: Write a program that uses a for loop to print even numbers from 2 to 20. 3. Print Odd Numbers: Write a program that uses a for loop to print odd numbers from 1 to 15. 4. Print Multiplication Table: Write a program that takes an integer as input and uses a for loop to print its multiplication table. 5. Calculate Factorial: Write a program that takes an integer as input and uses a for loop to calculate and print its factorial. 6. Print Reverse: Write a program that takes a string as input and uses a for loop to print it in reverse. 7. Sum of Digits: Write a program that takes an integer as input and uses a for loop to calculate and print the sum of its digits. 8. Power of a Number: Write a program that takes two integers, base and exponent, as input and uses a for loop to calculate and print the result of base raised to the power of exponent.

Arrays	1. program that finds the maximum and minimum values in an array of integers. 2. program that calculates the sum and average of elements in an array of numbers. 3. Write a Java program to reverse the elements of an array. 4. Write a Java program that finds and prints the duplicate elements in an array. 5. program that finds and prints the common elements between two arrays. 6. program that finds the second largest element in an array of integers. 7. Write a Java program to add two matrices of the same size. Sample int[][] matrix1 = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9} }; int[][] matrix2 = { {9, 8, 7}, {6, 5, 4}, {3, 2, 1} }; 8. program to find the sum of the diagonal elements of a square matrix. 9. program to check if a given matrix is symmetric.
Enhanced For loop	1. program to find the sum of all elements in an integer array using the enhanced for loop 2. program to reverse an array of strings using the enhanced for loop. 3. generate a multiplication table using a 2D integer array and the enhanced for loop. 4. find the sum of all elements in a 2D integer array using the enhanced for loop.
String manipulations	1. Write a program that takes a string as input and counts the number of vowels and consonants in it. 2. Write a program that checks whether a given string is a palindrome (reads the same forwards and backwards). 3. Write a program that counts the number of words in a given sentence. 4. Write a program that removes duplicate characters from a given string. 5. Write a program that counts the occurrences of a specific character in a given string. 6. Write a program that checks whether two given strings are anagrams (contain the same characters in a different order). 7. Write a program that takes a sentence as input and capitalizes the first letter of each word. 8. Write a program that counts the occurrences of a given substring in a given string. 9. Write a program that replaces all occurrences of a given substring with another substring in a given string.
StringBuilder	1. Write a Java program that reverses a given string. 2. Write a Java program that removes duplicate characters from a given string. 3. Write a Java program that takes a sentence as input and capitalizes the first letter of each word.

StringBuffer	1. Write a Java program that takes a string as input and uses StringBuffer to reverse the string. 2. program that takes a string as input and uses StringBuffer to remove duplicate characters from the string. 3. program that takes two strings as input and uses StringBuffer to concatenate them. 4. program that checks if a given string is a palindrome (reads the same forwards and backwards) using StringBuffer. 5. Write a Java program that takes a string as input and uses StringBuffer to count the number of vowels and consonants in the string.
Useful Classes	1. Write a Java program that takes a sentence as input and uses the StringTokenizer class to tokenize the sentence into individual words. 2. Write a Java program that takes a date in the format "dd-MM-yyyy" as input and converts it to a Date object using the SimpleDateFormat class. 3. Write a Java program that calculates the age of a person based on their birthdate. Use the Date class and Calendar class to perform the calculations. 4. Write a Java program that takes a Date object and formats it to display in the format "yyyy-MM-dd HH:mm:ss" using the SimpleDateFormat class. 5. Write a Java program that calculates the number of days between two given dates using the Date class and Calendar class. 6. Write a Java program that displays the current date and time using the Date class and SimpleDateFormat class.
Class and Object basics	1. Create a class Car with attributes make, model, and year. Create two objects of the Car class and demonstrate their usage. 2. Create a class Student with attributes name, age, and studentId. Create two objects of the Student class and demonstrate their usage. Print out the information of each student object.
Variables	1. Write a Java program that calculates the area of a rectangle. Create a method that takes the length and width as parameters, calculates the area using a local variable, and returns the result. 2. Create a class Constants with final variables for common mathematical constants like pi and e. Use these constants in calculations to demonstrate their usage. 3. Create a class Person with instance variables name and age. Initialize these variables. Create two Person objects and demonstrate their attributes.
Methods	1. Create a class Rectangle with instance methods setDimensions(int length, int width) and calculateArea() that calculates the area of a rectangle using its length and width. Create an object of the Rectangle class, set dimensions, and calculate the area. 2. Create a class ShapeAreaCalculator with overloaded methods to calculate the area of different shapes (circle, rectangle, triangle) using appropriate parameters. Demonstrate the use of these methods. 3. Create a class TemperatureConverter with static methods celsiusToFahrenheit(double celsius) and fahrenheitToCelsius(double fahrenheit) to convert temperatures. Demonstrate the use of these methods by converting temperatures.

Access Modifiers	1. Default Access Modifier: Create a class Person with instance variables name and age. Write methods to set and get these variables. Create an object of the Person class and demonstrate accessing the instance variables using methods. 2. Create a class BankAccount with the following private instance variables: accountNumber, balance, and ownerName. i. Write a private method generateAccountNumber() that generates a random account number for each new account. ii. Write a public method openAccount(String ownerName, double initialBalance) that initializes the ownerName, sets the balance to initialBalance, and calls the private generateAccountNumber() method to assign an account number. iii. Write a public method getAccountInfo() that displays the account details (account number, owner name, and balance). Create a Main class to demonstrate the BankAccount class. Open two bank accounts and display their information using the getAccountInfo() method.