Bharatiya Vidya Bhavan's Atmakuri Rama Rao School

Road No. 45 (End), Jubilee Hills, Hyderabad - 500 033

Note:

Dear students, please ensure that the program is written on the right-hand (ruled) page of your record, and the corresponding output is written on the left-hand (unruled) page.

Java Programs

1. **Write a program to check whether a given number is odd or even.**

```
public class Odd Even
{
  public static void main (String [ ] args)
  {
        int number = 15;
        if (number % 2 == 0)
         {
            System.out.println("The given number "+ number + " is Even");
        }
        else
        {
           System.out.println("The given number "+ number + " is Odd");
        }
   }
}
```

Output:

The given number 15 is Odd

2. **Write a program to check whether the given year is leap year or not.**

```
public class Leap
{
   public static void main (String args[])
   {
     int year = 2023;
     if ((year % 4 == 0 && year % 100 != 0) || ( year % 400 == 0))
     {
       System.out.println("The given year "+ year + " is Leap Year");
     }
     else
     {
        System.out.println("The given year "+ year + " is not Leap year");
     }
   }
}
```

Output:

The given year 2023 is not Leap year

3. **Write a program to perform arithmetic operations using switch statement.**

```
public class Calculator
{
   public static void main (String args[ ])
   {
      int a = 10, b = 5;
      char ch= '+';
      switch (ch)
      {
         case '+':
            System.out.println("Result: " + (a + b));
            break;
         case '-':
            System.out.println("Result: " + (a - b));
            break;
         case '*':
            System.out.println("Result: " + (a * b));
            break;
         case '/':
            System.out.println("Result: " + (a / b));
             break;
         Case '%':
            System.out.println("Result: " + (a % b));
            break;
         default:
            System.out.println("Invalid operator");
      }
     }
}
```

Output:

Result: 15

4. **Write a program to check the given number is +ve or -Ve.**

```
class Demo
{
    public static void main (String[ ] args)
   {
      int number = 12;
      if (number > 0)
        System.out.println("Given number is +ve");
      else
       System.out.println(" Given number is _ve");
  }
}
```

Output: Given number is +ve

5. **Write a program to print even numbers from 1 to 100.**

```
public class EvenNumbers
 {
   public static void main(String[] args)
 {
      System.out.println("Even numbers from 1 to 100 are:");

      for (int i = 2; i <= 100; i++)
```

```
        {
              System.out.println(i);
               i=i+1;
        }
    }
}
```

Output:

```
2
4
6
8
10
12
14
16
18
.
.
.
.
.
.
.
100
```

6. **Write a program to print 5 multiples.**

```
public class Test
 {
   public static void main(String[] args)
 {
      System.out.println("Five multiples are :");

      for (int i = 1; i <= 10; i++)
      {
            System.out.println( "5" + "X" + i + "=" + 5*i );
      }
   }
}
```

Output:

```
Five multiples are :
5X1=5
5X2=10
5X3=15
5X4=20
5X5=25
5X6=30
5X7=35
5X8=40
5X9=45
5X10=50
```

7. **Write a program to print even numbers from 1 to 100 using while loop.**

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

```
 {
       int i = 2;
       while (i <= 100)
         {
          System.out.println(i);
          i += 2;
       }
   }
}
```

Output:

```
2
4
6
8
10
12
14
16
18
.
.
.
.
.
.
.
.
.
100
```

8. **Write a program to print square of a number using do – while loop.**

```
class suoare
{
   public static void main(String[] args)
 {
       int i = 1;
       System.out.println("The square of numbers from 1 to 100 are  :");
     do
    {
          System.out.println(i * i);
          i= i+ 1;
       }while (i < 10);
   }
}
```

Output:

```
The square of numbers from 1 to 100 are:
1
4
9
16
25
36
```

```
49
64
81
100
```

9. **Write a program to print sum of array elements.**

```
class Array
{
   public static void main(String[] args)
 {
       int[] arr = {10,20,30,40,50};
       int sum=0;
       for (int i=0; i< arr.length; i++)
       {
          sum= sum + arr[i];
       }
       System.out.println("The sum of array elements are : "+ sum);
   }
}
```

Output:

The sum of array elements are : 150

10. **Write a program to calculate area of rectangle using methods in java.**

```
class Rectangle
{
   static int area(int length, int breadth)
   {
      int result = length * breadth;
      return result;
   }
   public static void main(String[] args)
   {
       int l= 10;
      int b= 20;
      int area1 = area(l,b);
      System.out.println("The area of rectangle : "+ area1);
   }
}
```

Output:

The area of rectangle: 200

11. **Write a program to demonstrate string functions in java.**

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

```
        {
            String Text ="Hello";
            String Text1 = "Good Morning";
            int length;
             System.out.println("The Length of Text1 : "+ Text1.length());
             System.out.println(" Upper case of text is :" + Text.toUpperCase());
            System.out.println(" Concatination of Text & Text1 is :" + Text.concat("
          ").concat(Text1));
        }
   }
```

Output:

```
The Length of Text1 : 12
 Upper case of text is :HELLO
 Concatination of Text & Text1 is: Hello Good Morning
```

12. Write a program to demonstrate exceptional handling.

```
class Demo
{
   static int devide(int length, int breadth)
   {
      int result = length / breadth;
      return result;
   }
   public static void main(String[] args)
    {
     try
     {
      int l= 10;
      int b= 0;
      int area1 = devide(l,b);
      System.out.println("The area of rectangle : "+ area1);
      } catch (Exception e)
         {
            System.out.println(e.getMessage());
         }
   }
}
```

Output:

```
/ by zero
```

13. **Write a program to display numbers in an array.**

```
public class SumOfArray
 {
   public static void main(String[] args)
 {
       int[] numbers = {10, 20, 30, 40, 50};
       for (int i = 0; i < numbers.length; i++)
       {
          System.out.println(numbers[i]);
       }
   }
}
```

Output :

```
10
20
30
40
50
```

14. **Write** a program for string handling functions.

```
public class stringhandling
{
   public static void main(String[] args)
   {
      String Name = " BVBARRS";
      System.out.println(" Replace BVBARRS with School = " + Name.replace("BVBARRS",
"SCHOOL"));
       System.out.println(" String in Lower case = " + Name.toLowerCase());
       System.out.println(" Character at = " + Name.charAt(5));
       System.out.println(" String in Lower case = " + Name.indexOf("A"));
   }
}
```

Output:

```
Replace BVBARRS with School =  SCHOOL
 String in Lower case =  bvbarrs
 Character at = R
 String in Lower case = 4
```

15. Write a program to reverse a number.

```
class ReverseNumber
{
   public static void main(String[] args)
   {
       int num = 123;
       int reverse = 0;
       while (num != 0)
       {
           int digit = num % 10;
```

```
            reverse = reverse * 10 + digit;
            num = num / 10;
        }
        System.out.println("Reversed Number: " + reverse);
    }
}
```

Output:
Reversed Number: 321

SQL Queries

1. Create following table using SQL statements.

Table Name1 : Client

Field name	Data Type	Constraints
Clientno	Char(6)	Primary Key
Name	Varchar(20)	Not Null
City	Varchar(20)	Not Null
Pin	Number	
Mobile	Number	

```
CREATE TABLE Client (
                        CNo integer primary key,
                        Name varchar(15) not null,
                       City varchar(10) not null,
                        Pin intege,
                        Mobile char(10));
```

- **Query 1: Insert the following data into table.**

Output:

CNo	**Name**	**City**	**Pin**	**Mobile**
101	**Manu**	**Hyderabad**	**500033**	**123456789**
102	**Prema**	**Bangalore**	**500013**	**234567891**
103	**Priti**	**Delhi**	**212234**	**345678912**
104	**Hema**	**Bombay**	**234432**	**456789123**
105	**Seema**	**Hyderabad**	**567123**	**567891234**

insert into Client values (101, " Manu", "Hyderabad" , 500033, "123456789");

insert into Client values (102, " Prema", "Bangalore" , 500013, "234567891");

insert into Client values (103, " Priti ", " Delhi " , 212234, "345678912");

insert into Client values (104, " Hema ", " Bombay " , 234432, "456789123");

insert into Client values (105, " Seema ", " Hyderabad " , 567123, "567891234");

- **Query 2: Display the details of clients whose city is Hyderabad.**

Select * from Client where city like 'Hyderabad';

Output:

CNo	Name	City	Pin	Mobile
101	Manu	Hyderabad	500033	123456789

- **Query 3: write a query to count number of clients.**

  select count(*) from Client;

Output:

count(*)
5

Query 4: write q query to display all the details

select * from Client;

Output:

CNo	Name	City	Pin	Mobile
101	Manu	Hyderabad	500033	123456789
102	Prema	Bangalore	500013	234567891
103	Priti	Delhi	212234	345678912
104	Hema	Bombay	234432	456789123
105	Seema	Hyderabad	567123	567891234

Query 5: write a query to display Names of clients in Ascending order.

select * from Client order by Name asc;

Output:

CNo	Name	City	Pin	Mobile
104	Hema	Bombay	234432	456789123
101	Manu	Hyderabad	500033	123456789
102	Prema	Bangalore	500013	234567891
103	Priti	Delhi	212234	345678912
105	Seema	Hyderabad	567123	567891234

2. Create following table using SQL statements where TID is unique and not null, TName must have a value, Boarding, Destination and Cost.

Table Name: Train

```
CREATE TABLE Train (
                    TID integer primary key,
                    TName varchar(15) not null,
                    Boarding varchar(20),
                    Destination varchar(20),
                    Cost Integer);
```

- **Query 1: Insert the following data into table.**

Insert into Train values (305, “VandeBharat”, “New Delhi”, “Katra”, 2500);

Insert into Train values (206, “Shatabdi”, “Mumbai”, “Delhi”, 3500);

Insert into Train values (307, “Superfast”, “Lucknow”, “Chennai”, 4500);

Insert into Train values (208, “Holiday Express”, “Lucknow”, “Delhi”, 2000);

Output:

TID	TName	Boarding	Destination	Cost
305	VandeBharat	New Delhi	Katra	2500
206	Shatabdi	Mumbai	Delhi	3500
307	Superfast	Lucknow	Chennai	4500
208	Holiday Express	Lucknow	Delhi	2000

- **Query 2 : Display the details of all the passengers whose train fair (Cost) is more than Rs. 3000/-**

```
select * from Train where Cost > 3000;
```

Output:

TID	TName	Boarding	Destination	Cost
206	Shatabdi	Mumbai	Delhi	3500
307	Superfast	Lucknow	Chennai	4500

Query 3: Write a query to remove the destination column from the Train table

```
ALTER TABLE Train DROP COLUMN Destination;
```

Query 4: display all the records from Train table.

```
select * from Train1;
```

Output:

TID	TName	Boarding	Cost
305	VandeBharat	New Delhi	2500
206	Shatabdi	Mumbai	3500
307	Superfast	Lucknow	4500
208	Holiday Express	Lucknow	2000

- **Query 5**: find the total amount collected by the customers.

```
select Sum(Cost) from Train1;
```

Sum(Cost)
12500

3. Create a table using SQL commands:

Table Name : Passenger

PID	TID	Name	DOT
P108	307	Gagandeep Singh	14-10-2022
P209	206	Rahat Ali	01-01-2023
P110	305	Cristopher	10-12-2022
P677	208	Diksha Luther	11-09-2019
P555	307	Nira Kadam	06-11-2020
P211	208	Ritam Ahuja	17-12-2021

- **Query 1: Create a table with above data:**

```
Create  table Passenger (  PID char(5) Primary key,
                           TID integer,
                      Name Varchar(20),
                      DOT date);
```

- **Query 2: Insert the following data into table.**

```
Insert into Passenger values ( “P108”,307, “Gagandeep Singh”, 2022-10-14 );
Insert into Passenger values (“P209”,  206, “Rahat Ali”, 2023-01-01 );
Insert into Passenger values (“P110”, 305, “Cristopher”, 2022-12-10);
Insert into Passenger values ( “ P677”, 208, “Diksha Luther”, 2019-09-11);
Insert into Passenger values ( “ P555”, 307, “Nira Kadam”, 2020-11-06);
```

- **Query 3: To display TID ,Name, and DOT of all passengers.**

```
select TID ,Name,DOT from Passenger;
```

Output:

TID	Name	DOT
307	Gagandeep Singh	1998
206	Rahat Ali	2021
305	Cristopher	2000
208	Diksha Luther	1999
307	Nira Kadam	2003

- **Query 4: To display details of all TID between 206 and 306.**

  select * from Passenger where TID between 206 and 306;

Output:

PID	TID	Name	DOT
P209	206	Rahat Ali	2021
P110	305	Cristopher	2000
P677	208	Diksha Luther	1999

- **Query 5 : Write a query to delete passenger table;**

  Drop table Passenger;

4. Create a table using SQL

Table Name: Teacher

TID	TNAME	DOA	DEPT	PAY	
501	SUBHAV JAIN	2014-07-09	COMPUTER	450000	
203	SHUBH NANGIA	2009-11-23	CHEMISTRY	340000	
510	VARUN KAPOOR	2015-02-15	HINDI	320000	
307	KAMAL PATRA	2010-12-29	ACCOUNTS	310000	
490	SUJAN REDDY	2012-09-11	ENGLISH	270000	

Create table Teacher (TID number primary key, TName varchar(20) Not null, DOA Date, Dept varchar (20), Pay number);

Query 1: insert the above details:

Insert into Teacher values (501, "SUBHAV JAIN", 2014-07-09, "COMPUTER", 450000);

Insert into Teacher values (203, "**SHUBH NANGIA** ", **2009-11-23**, "**CHEMISTRY** ", **340000**);

Insert into Teacher values (**510**, "**VARUN KAPOOR** ", **2015-02-15**, "**HINDI** ", **320000**);

Insert into Teacher values (**307**, "**KAMAL PATRA** ", **2010-12-29**, "**ACCOUNTS** ", **310000**);

Query 2: To display the number of teachers and the corresponding department name, in each department.

Count (TNAME)	Dept
1	ACCOUNTS
1	CHEMISTRY
1	COMPUTER
1	HINDI

Query 3: To display the minimum PAY of teachers

SELECT min (PAY) FROM Teacher;

Min (PAY)
310000

Query 4: to find the sum of salary of teachers.

Select sum(Pay) from Teacher;

Sum (Pay)
1420000

Query 5: write a query to display names of teacher whose name starts with "S"

Select * from Teacher where TName Like "s%";

TID	TName	DOA	Dept	Pay
501	SUBHAV JAIN	1998	COMPUTER	450000
203	SHUBH NANGIA	1975	CHEMISTRY	340000

5. **Create a table using SQL commands.**

Table Name: Student

Create Table Student (ADMNO number primary key, NAME varchar(20) not null, GRADE number, DOB date, Marks number, Gender char(2));

Query 1: write a command to delete DOB from Student table;

Alter table Student drop DOB;

Query 2: create a query to add the following fields in to table.

ADMNO	NAME	GRADE	MARKS	GENDER
101	Rupal	12	90	F
102	Varun	12	86	M
103	Medha	12	95	F
104	Seema	12	76	F
105	Arun	12	66	M

```
Insert into Student values (101, 'Rupal', 12, 90, 'F');

Insert into Student values (102, 'Varun', 12, 86, 'M');

Insert into Student values (103, 'Medha', 12, 95, 'F');

Insert into Student values (104, 'Seema', 12, 76, 'F');

Insert into Student value s(103, 'Arun', 12, 66, 'M');
```

Query 3: create a query to sort student details based on marks.

```
select * from Student order by marks desc;
```

ADMNO	NAME	GRADE	Marks	Gender
103	Medha	12	95	F
101	Rupal	12	90	F
102	Varun	12	86	M
104	Seema	12	76	F

Query 4: create a query to display student details where ADMNO is 103.

```
Select * from Student where ADMNO=103;
```

ADMNO	NAME	GRADE	Marks	Gender
103	Medha	12	95	F

Query 5: create a query to change the marks of student whose ADMNO is 104.

```
Update Student set Marks = 98 where ADMNO = 104;

Select * from Student;
```

ADMNO	NAME	GRADE	Marks	Gender
101	Rupal	12	90	F
102	Varun	12	86	M
103	Medha	12	95	F
104	Seema	12	98	F