Experiment 1 - Write a program using command line argument in java

Echoing Command-Line Arguments:

```
class CommandArgsEcho
{
  public static void main(String args[])
  {
        for(String str : args)
        {
            System.out.println(str);
        }
    }
}
```

Output :

```
java CommandArgsEcho Hello World from main
Hello
World
from
main
```

Parsing Numeric Command-Line arguments

```
class CmdArgsParseNum
{
    public static void main(String args[])
    {
        int sum = 0;
        for(int i = 0; i < args.length; i++)
        {
            sum += Integer.parseInt(args[i]);
        }
        System.out.println("The Sum of all the integers is: " + sum);
    }
}
```

Output :

```
java CmdArgsParseNum 1 2 3 4 5
The Sum of all the integers is: 15
```

Experiment 2

Programs on Basic programming constructs like branching and looping

WAP to print the roots of quadratic equation.

```
// a. WAP to print the roots of quadratic equation
import java.util.Scanner;
class Roots{
  public static void main(String args[]) {
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter the coefficients of the quadratic equation");
    int a, b, c;
    a = sc.nextInt();
    b = sc.nextInt();
    c = sc.nextInt();
    int D = (b*b - 4*a*c);
    double rootD = Math.sqrt(D);
    boolean is0 = (D == 0);
    System.out.println(( is0 ? ("The root of the equation is " + -b / (2*a)) :
                               ("The roots are:  " +
                               ((-b + rootD) / (2*a)) + "and "+
                               ((-b - rootD) / (2*a)) )));
  }
}
```

Output :

```
Enter the coefficients of the quadratic equation
1
-2
1
The root of the equation 1
```

WAP to check if entered number is a prime number.

```
import java.util.Scanner;
class TestPrime{
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        int num, i;
        System.out.println("Enter a number");
        num = sc.nextInt();
        for( i = 2; i <= num / 2; i++)
        {
            if(num % i == 0)
            {
                System.out.println("The Number is not prime");
                break;
            }
        }
        if(i == num / 2 + 1)
        {
            System.out.println("The number is prime");
        }
    }
}
```

Output :

```
Enter a number
35
The Number is not prime
```

Study of different operators in java

WAP to compare two numbers

```
import java.util.Scanner;
class TestCompare
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        int num1, num2;
        System.out.println("Enter two numbers: ");
        num1 = sc.nextInt();
        num2 = sc.nextInt();
        System.out.println("The greater number is: " + ((num1 > num2) ? num1: num2));
    }
}
```

Output :

```
Enter two numbers:
45
65
The greater number is: 65
```

WAP to print truth table for java logical operators

```
class LogicalOperators
{
    public static void main(String args[])
    {
        boolean a = false;
        boolean b = false;
        System.out.println("A | B | !A  | A && B  | A || B ");
        for(int i = 0; i < 4; i++)
        {
            if( i == 2){a = !a;}
            System.out.printf("%d | %d |  %d  |    %d    |    %d  \n",
                              a ? 1 : 0, b ? 1 : 0, (!a ? 1 : 0),
                              (a && b ? 1 : 0), (a || b ? 1 : 0));
            b = !b;
        }
    }
}
```

Output :

```
A | B | !A  | A && B  | A || B
0 | 0 |  1  |    0    |    0
0 | 1 |  1  |    0    |    1
1 | 0 |  0  |    0    |    1
1 | 1 |  0  |    1    |    1
```

WAP to read the number & shift left & right by 3 bits

```
import java.util.Scanner;
class BitShift
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter an integer: ");
        int num = sc.nextInt();
        System.out.println("The number left shifted thrice: " + (num << 3));
        System.out.println("The number right shifted thrice: " + (num >> 3));
    }
}
```

Output :

```
Enter an integer: 8
The number left shifted thrice: 64
The number right shifted thrice: 1
```

Programs using accepting input through keyboard

Print the Fibonacci series upto the nth term taking the value of n from the user.

```java
import java.util.Scanner;
class Fibonacci
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number of terms to print: ");
        int n = sc.nextInt();
        int a = 0;
        int b = 1;
        for( int i = 0; i < n  ; i++)
        {
            System.out.print(a + "  ");
            b = b + a;
            a = b - a;
        }
    }
}
```

Output :

```
Enter the number of terms to print: 10
0  1  1  2  3  5  8  13  21  34
```

WAP to reverse the given no.

```java
import java.util.Scanner;
class Reverse
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        int rem, rev = 0;
        while(num > 0)
        {
            rem = num%10;
            rev = rev*10 + rem;
            num = num/10;
        }
        System.out.println("The Reverse of the given number is: " + rev);
    }
}
```

Output :

```
Enter a number: 1234
The Reverse of the given number is: 4321
```

WAP to calculate area & circumference of circle

```java
import java.util.Scanner;
class AreaCircumference
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the radius of the Circle: ");
        int radius = sc.nextInt();
        double area = Math.PI*Math.pow(radius, 2);
        double circumference = 2*Math.PI*radius;
        System.out.println("The circumference is " + circumference +
                           " and the Area is "+ area);
    }
}
```

Output :

```
Enter the radius of the Circle: 4
The circumference is 25.132741228718345 and the Area is 50.26548245743669
```

WAP to swap given two strings

```java
import java.util.Scanner;
class Swap
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str1 = sc.nextLine();
        System.out.print("Enter another string: ");
        String str2 = sc.nextLine();
        System.out.println("Strings Before Swapping: " + "str1: " +
                            str1+ " str2: "+ str2);
        String temp = str1;
        str1 = str2;
        str2 = temp;
        //Write about copying references and making a deep copy
        System.out.println("Strings After Swapping: " + "str1: " +
                            str1+ " str2: "+ str2);
    }
}
```

Output :

```
Enter a string: Hi
Enter another string: There
Strings Before Swapping: str1: Hi str2: There
Strings After Swapping: str1: There str2: Hi
```

WAP to convert temperature from Fahrenheit to Celsius

```java
import java.util.Scanner;
class FarenheitToCelsius
{
    public static int toCelsius(int farenheit)
    {
        return (farenheit - 32) * 5 / 9;
    }
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the temperature in Farenheit: ");
        int fahrenheit = sc.nextInt();
        int celsius = toCelsius(fahrenheit);
        System.out.print("The temperature in Celsius: " + celsius);
    }
}
```

Output :

```
Enter the temperature in Farenheit: 32
The temperature in Celsius: 0
```

WAP to find a square, square root, and Cube of a given no. using abstraction

```java
// WAP to find a square, square root, and Cube of a given no. using abstraction
import java.util.Scanner;
class MathOperations
{
    public static double cube(double num)
    {
        return num*num*num;
    }

    public static double square(double num)
    {
        return num*num;
    }

    public static double sqrt(double num)
    {
        double i;
        for(i = 0; !(i*i > num); i = i + 0.01);
        return i;
    }
    public static void main(String args[])
    {
        int i = 10;
        i = (int) cube(i);
        int j = 100;
        j = (int) sqrt(j);
        System.out.println(j);
        System.out.println(i);
    }
}
```

Output :

```
Enter a number: 100
Enter the operation: 1. Square 2. Square Root 3. Cube 4. Exit: 1
10000
Enter the operation: 1. Square 2. Square Root 3. Cube 4. Exit: 2
10
Enter the operation: 1. Square 2. Square Root 3. Cube 4. Exit: 3
1000000
Enter the operation: 1. Square 2. Square Root 3. Cube 4. Exit: 4
```

Experiment4:

Code:

```java
import java.util.Scanner;
/*
The Account class containing the following:
Data:
          name of the depositor - name
          account number - accNumber
          type of account - accType
          balance amount in the account - balance
Methods:
          1.to assign initial values - createAccount
          2.to deposit an amount - deposit
          3.to withdraw an amount after checking balance - withdraw
          4.to display the name & balance - accDetails
*/
class Account
{
  String name;
  String accNumber;
  String accType;
  int balance;
  void createAccount(String name, String accNumber, String accType)
  {
    this.name = name;
    this.accNumber = accNumber;
    this.accType = accType;
    this.balance = 0;
  }
  void deposit(int value)
  {
    balance = balance + value;
  }
  void withdraw(int value)
  {
    if(value > balance)
    {
      System.out.println("Insufficient balance");
    }
    else
    {
      balance = balance - value;
    }
  }
  void accDetails()
  {
    System.out.println("Account Holder: " + name);
    System.out.println("Balance: " + balance);
  }
}
// Demonstrating the Account Class
class BankAccount
{
  public static void main(String args[])
  {
    Scanner sc = new Scanner(System.in);
    Account acc1 = new Account();
    int choice, amount;
    boolean exit = false;
    while(!exit)
    {
      System.out.print("Select an option:\n1. Create an account\n 2. Deposit\n"
                "3. Withdraw\n4. Account Details\n5. Exit\n  -->");
      choice = sc.nextInt();
      switch(choice)
      {
        case 1:
          sc.nextLine();
          System.out.print("Enter the account holder's name: ");
          String name = sc.nextLine();
          System.out.print("Enter the account number: ");
          String accNumber = sc.next();
          System.out.print("Enter the account type: ");
          String type = sc.next();
          acc1.createAccount(name, accNumber, type);
          break;
        case 2:
          System.out.print("Enter the amount to deposit: ");
          amount = sc.nextInt();
          acc1.deposit(amount);
          break;
        case 3:
          System.out.print("Enter the amount to withdraw: ");
          amount = sc.nextInt();
          acc1.withdraw(amount);
          break;
        case 4:
          acc1.accDetails();
          break;
        case 5:
          exit = true;
          break;
        default:
          System.out.println("Enter a valid option");
      }
    }
  }
}
```

Output:

Name: Aum Kulkarni

Roll No: 36

Div: D6AD

Experiment No: 6

Experiment Title: Program on Packages:
Write a Program to demonstrate user defined Packages

Directory Structure:

```
D:\JAVACODE\EXPERIMENTS\EXPERIMENT6
├───package1
└───package2
    └───packageA
```

Code:

```java
// package1/Class1.java
package package1;
public class Class1
{
  public Class1()
  {
    System.out.println("Hello From Class1 from package1");
  }
}
```

```java
// package1/Class2.java
package package1;
public class Class2
{
  public Class2()
  {
    System.out.println("Hello From Class2 from package1");
  }
}
```

```java
// package1/Class3.java
package package1;
public class Class3
{
  public Class3()
  {
    System.out.println("Hello From Class3 from package1");
  }
}
```

```java
// package2/ClassA.java
package package2;
public class ClassA
{
  public ClassA()
  {
    System.out.println("Hello From ClassA from package2");
  }
}
```

```java
// package2/package3/Class1.java
package package2.packageA;
public class Class1
{
  public Class1()
  {
    System.out.println("Hello From Class1 from package3");
  }
}
```

```java
// SampleClass.java
import package1.*;
import package2.ClassA;
import package2.packageA.*;
class SampleClass
{
  public static void main(String args[])
  {
    package1.Class1 c1 = new package1.Class1();
    Class2 c2 = new Class2();
    Class3 c3 = new Class3();
    ClassA cA = new ClassA();
    package2.packageA.Class1 c31 = new package2.packageA.Class1();
  }
}
```

```java
    // SampleClass.java
    import package1.*;
    import package2.ClassA;
    import package2.packageA.*;
    class SampleClass
    {
        public static void main(String args[])
        {
            package1.Class1 c1 = new package1.Class1();
            Class2 c2 = new Class2();
            Class3 c3 = new Class3();
            ClassA cA = new ClassA();
            package2.packageA.Class1 c31 = new package2.packageA.Class1();
        }
    }
```

Output:

```
Hello From Class1 from package1
Hello From Class2 from package1
Hello From Class3 from package1
Hello From ClassA from package2
Hello From Class1 from package3
```