

}

}

$$x = \frac{34}{10} = 3$$

$$d = 3$$

$$\text{sum} = 9 + 3 = 12$$

$$x = \frac{3}{10} = 0.3$$

- 38) define a method to return of the sum of square of digits
- 39) Define a method to return product of digits?
- 40) Define a method to print how many even & odd digits present in the number?
- 41) Define a method to return how many prime digits present in the number?
- 42) Define a method to return the reverse of the number?

38) Define a method to return the sum of square of digits?

```
import java.util.Scanner;
class Square
{
    PSUM()
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        s.o.p("enter the number:");
        s.o.p(sum);
        int s = getValues(n);
        s.o.p(sum); s.o.p("values of " + n + " is " + s);
    }
}
```

static ~~void~~ ^{int} getValues(int n)

```
{
    int sum = 0, a = 1;
    while (a <= n)
    {
        sum = sum + (a * a);
        a++;
    }
    return sum;
}
```

```
int sum = 0;
while (n != 0)
{
    int d = n / 10;
    sum = sum + (d * d);
    n = n / 10;
}
return sum;
```

O/P
n = 345
sum = <u>50</u>
i.e) $3^2 + 4^2 + 5^2$ = 50
(3*3) + (4*4) + (5*5)

39) Define a method to return product of digits? (528)

class Product

```
{
    PSUM()
    {
        Scanner sc = new Scanner(System.in);
        s.o.p("enter a number:");
        int n = sc.nextInt();
        int product = findProduct(n);
        s.o.p("Product of digit: " + product);
    }
}
```

```
int n = sc.nextInt();
```

```
int product = findProduct(n);
```

```
s.o.p("Product of digit: " + product);
```

5 x 2 x 8 = 80

Static int find Product (int n)

```

{
    int product = 1;
    while (n > 0) (n != 0)
    {
        product = n * 10;
        n = 10;
        int d = n / 10;
        product = product * d;
        n = n / 10;
    }
    return product;
}

```

o/p

enter : 528

ans : 80

i.e) $5 \times 2 \times 8 = 80$

40) Define a method to print how many even & odd digits present in the number?

class address

{ psvm ()

{ scanner sc = new scanner (system.in);

s.o.p ("enter the number:");

int n = sc.nextInt();

printCount (n);

} static void printCount (int n)

{ int ecount = 0, dcount = 0;

~~while (n != 0)~~

do { int d = n / 10;

if (d % 2 == 0)

ecount ++;

else dcount ++;

n = n / 10; } while (n != 0);

s.o.p (" The total no. even digit is : " + ecount);

s.o.p (" The total no. even digit is : " + dcount);

}

41) Define a method to return how many prime digits present in the number?

class Prime

{
 psvm()

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

 s.o.p("Enter the values");

 int num = sc.nextInt();

 int p = method(num); s.o.p("The prime digit " + num + " is " + p);

 } static int method(int num)

 {
 int count = 0;

 while (num > 0)

 {
 int no = num / 10

 if (no == 1 || no == 3 || no == 5 || no == 7 || no == 2) {
 num = num / 10;

 count++;

 return num;

42) Define a method to return the reverse of the number?

import java.util.Scanner;

class Reverse

{
 psvm()

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

 s.o.p("Enter the number:");

 int n = sc.nextInt();

 int rd = reverseDigit(n);

 s.o.p("Reverse digit " + n + " is " + rd);

 } static int reverseDigit(int n)

 {
 int r = 0;

 while (n != 0)


```

    {
        int d = n % 10;
        r = r * 10 + d;
        n = n / 10;
    }
    return r;
}

```

7/7/22

43) Define a method to return true if the number is palindrome otherwise false?

class Palindrome

```

{
    Psum()
    {
        Scanner sc = new Scanner(System.in);
        sop("enter the number:");
        int n = sc.nextInt();
        boolean rs = isPalindrome(n);
        if (rs == true)
            sop("no. is palindrome");
        else
            sop("no is not palindrome");
    }
}

```

static boolean isPalindrome (int n)

```

{
    int rev = 0, temp = n;
    do {
        int d = n % 10;
        rev = rev * 10 + d;
    }
}

```

```

    n = n / 10;
    } while (n != 0);
    return rev == temp;
}
}

```

4) Define a method to return true if the number is strong number otherwise false?

class strongNumbers

{
 psum()

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

 int sop("Enter the number:");

 int n = nextInt();

 boolean sh = isStrong(n);

 if (sum == true)

 sop("The number of " + n + " is palindrome");

 else
 sop("The number of " + n + " is not palindrome");

}
 static boolean isStrong(int n)

{
 int sum = 0, temp = n;

 while (n != 0)

{
 int d = n % 10;

 sum = sum + fact(d);

 n = n / 10;

 }

 return sum == temp;

~~static boolean isStrong (int d)~~

static int fact (int d)

```
{
    int fact = 1;
    while (d > 1)
    {
        fact = fact * d;
        d--;
    }
    return fact;
}
```

45) Define a method return "true" if no is armstrong no, otherwise return false?

class Armstrong

{ static boolean isArmstrong (int x)

{ int sum = 0, temp = x;

int dc = countDigit (x);

do

{ int d = x % 10;

sum = sum + power (d, dc);

x = x / 10;

} while (x != 0);

return sum == temp;

}

```

static int countDigit (int n)
{
    int count = 0;
    do
    {
        count ++;
        n = n/10;
    } while (n != 0);
    return count;
}

```

```

static int power (int n, int p)
{
    int pw = 1;
    while (p > 0)
    {
        pw = pw * n;
        p--;
    }
    return pw;
}

```

```

psvm ()
{
    Scanner s = new Scanner (System.in);
    sop
}

```


4b) Define a method to return true if the number is disarium
false if the number is happy. But

57) Define a method to return true the number is happy number

48) Define a method to return ~~xylem~~ the number is xylem

otherwise return phloem.

49) Define a method to return average of digits
11.1 To return the diff. b/w bi

49) Define a method to return the diff. b/w biggest & smallest digit.

50) Define a method to return the diff. b/w biggest & smallest digit.