Pre-Programming M-2 Batch

Q1] Extract all digit in given number

```
package preprogramingM1;

import java.util.Scanner;

public class Extract_DIgit {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter the number : ");

int num = sc.nextInt();

while (num > 0) {

int rem = num % 10;

System.out.println(rem);

num = num / 10;

}

}

}
```

Q2] WAP to count no of digit in given number

```
package preprogramingM1;

import java.util.Scanner;

public class Count_Digti {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the number.....");

int num = scanner.nextInt();

int c = 0;

while (num > 0) {

num = num / 10;

c++;

}

System.out.println("Total digit in given number is: " + c);

}

}
```

Q3]WAP to find the sum of digit in given number

```
package preprogramingM1;

import java.util.Scanner;

public class SumOfDigit {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the number.....");
```

```
int num = scanner.nextInt();
int sum = 0;
while (num > 0) {
int rem = num % 10;
sum = sum + rem;
num = num / 10;
}
System.out.println("Total digit in given number is: " + sum);
}
}
```

Q4] WAP to find sum of Odd digit in number

```
package preprogramingM1;
import java.util.Scanner;
public class SumOfOddDigit {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number.....");
int num = scanner.nextInt();
int sum = 0;
while (num > 0) {
int rem = num % 10;
if(rem % 2 != 0)
sum = sum + rem;
num = num / 10;
}
System.out.println("Total digit in given number is: " + sum);
}
}
```

Q5] WAP to find sum of all even digit in number

```
package preprogramingM1;
import java.util.Scanner;
public class SumOfEvenNumber {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number.....");
int num = scanner.nextInt();
int sum = 0;
while (num > 0) {
int rem = num % 10;
```

```
if (rem % 2 == 0)
sum = sum + rem;
num = num / 10;
}
System.out.println("sum of even number digit is : " + sum);
}
}
```

Q6] WAP to check number is Spy or not

Spy number means : sum of digit == product of digit

Ex] 123 => spy number

1+2+3 = 6 ==========➔ sum

1*2*3=6 ==========➔ product

```
package preprogramingM1;
import java.util.Scanner;
public class SpyNumber {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Enter the numbe:");
int num = sc.nextInt();
int sum = 0;
int pro = 1;
while (num > 0) {
int rem = num % 10;
sum = sum + rem;
pro = pro * rem;
num = num / 10;
}
if(sum == pro) {
System.out.println("Number is Spy number");
}else {
System.out.println("Number is not Spy number");
}
}
}
```

Q7] WAP to find Factorial of given number

```
package preprogramingM1;
import java.util.Scanner;
public class FactorialNumber {
public static void main(String[] args) {
```

```
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number");
int num = scanner.nextInt();
long fact = 1;
for (int i = 1; i <= num; i++) {
fact = fact * i;
}
System.out.println("Factorial of " + num + " is " + fact);
}
}
```

Q8] find the factorial of each digit in given number

```
package preprogramingM1;
import java.util.Scanner;
public class FactorialOfEachDigitInNuber {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the Number");
int num = scanner.nextInt();
int fact = 1;
while (num > 0) {
int rem = num % 10;
for (int i = 1; i <= rem; i++) {
fact = fact * i;
}
System.out.println("Factorial of " + rem + " is " + fact);
num = num / 10;
}
}
}
```

Q9] find the factor of given number

```
package preprogramingM1;
import java.util.Scanner;
public class FactorOfNumber {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number to find the factor");
int num = scanner.nextInt();
System.out.print("Factor of the " + num + " : ");
for (int i = 1; i <= num; i++) {
```

```
if (num % i == 0) {
System.out.print(i + " ");
}
}
}
}
```

Q10] Find the factor of given digit in number

```
package preprogramingM1;
import java.util.Scanner;
public class FactorofEachDigit {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number");
int num = scanner.nextInt();
while (num > 0) {
int rem = num % 10;
System.out.print("Factor of " + rem + " : ");
for (int i = 1; i <= rem; i++) {
if (rem % i == 0) {
System.out.print(i + " ");
}
}
num = num / 10;
System.out.println();
}
}
}
```

Q11] find the total factor of given digit

```
package preprogramingM1;
import java.util.Scanner;
public class TotalFactorOfGivenNumber {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number : ");
int num = scanner.nextInt();
int count = 0;
for (int i = 1; i <= num; i++) {
if (num % i == 0) {
count++;
```

```
}
}
System.out.println("Total Factor of the " + num + " is " + count);
}
}
```

Q12] find the number is prime or not

```
package preprogramingM1;
import java.util.Scanner;
public class PrimeNumber {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
int num = sc.nextInt();
int c = 1;
for (int i = 2; i <= num; i++) {
if (num % 2 == 0) {
c++;
}
}
if (c == 2) {
System.out.println("Number is Prime number");
} else {
System.out.println("Number is not prime number");
}
}
}
```

Q13] find the prime number between o to 100

```
package preprogramingM1;
public class PrimeNumberBetweenNumbers {
public static void main(String[] args) {
System.out.println("Prime number between 1 to 100");
for (int i = 1; i <= 100; i++) {
int c = 0;
for (int j = 1; j <= i; j++) {
if (i % j == 0) {
c++;
}
}
if (c == 2) {
System.out.println(i);
```

```
}
}
}
}
```

Q14] find the reverse number of given number

```
package preprogramingM1;
import java.util.Scanner;
public class ReverseNumber {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Enter the number");
int num = sc.nextInt();
int rev = 0;
int num1 = num;
while (num > 0) {
int rem = num % 10;
rev = rev * 10 + rem;
num = num / 10;
}
System.out.println("Reverse number of " + num1 + " is " + rev);
}
}
```

15] check the number is palindrome or not

```
package preprogramingM1;
import java.util.Scanner;
public class PalindromeNumber {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number : ");
int num = scanner.nextInt();
int num1 = num;
int rev = 0;
for (int i = num; i > 0; i = i / 10) {
int rem = num % 10;
rev = rev * 10 + rem;
num = num / 10;
}
if (rev == num1) {
System.out.println("Number is palindrome");
```

```
} else {
System.out.println("Number is not palindrome");
}
}
}
```

16] WAP program to check number is perfect or not

Perfect number = sum of factor of given number expect it is called as perfect number

Ex : 4 -> 1,2 1+2 = 3 =➔ not perfect number

```
package preprogramingM1;
import java.util.Scanner;
public class PerfectNumber {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number: ");
int num = scanner.nextInt();
int sum = 0;
for (int i = 1; i < num; i++) {
if (num % i == 0) {
sum = sum + i;
}
}
if (sum == num) {
System.out.println("perfect number.....");
} else {
System.out.println("not perfect Number.....");
}
}
}
```

16] WAP to find perfect number between 1 to 700

```
package preprogramingM1;
public class PefectNumberBetweenTwoNumber {
public static void main(String[] args) {
System.out.println("perfect number between 1 to 700 is :");
for (int num = 1; num < 700; num++) {
int sum = 0;
for (int i = 1; i < num; i++) {
if (num % i == 0) {
sum = sum + i;
}
```

```
}
if (num == sum) {
System.out.println(num);
}
}
}
}
```

17] WAP to given number is strong number or not

Factorial of each digit in the number is equal to number is called as strong number

Ex : 145 ==== 1 => 1

4 => 24 1+24+120 = 145 this is strong number

5=>120

```
package preprogramingM1;
import java.util.Scanner;
public class StrongNumber {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Enter the number");
long num = sc.nextLong();
long num1 = num;
long sum = 0;
while (num > 0) {
long rem = num % 10;
long fact = 1;
for (int i = 1; i <= rem; i++) {
fact = fact * i;
}
sum = sum + fact;
num = num / 10;
}
if (sum == num1) {
System.out.println("Strong number.....");
} else {
System.out.println("not strong number.....");
}
}
}
```

18] WAP to find power of given base value

```
package preprogramingM1;
```

```
import java.util.Scanner;

public class Power {

public static void main(String[] args) {

int base = 2;

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the number");

int num = scanner.nextInt();

int pow = 1;

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

pow = pow * 2;

}

System.out.println("Power of " + num + " is " + pow);

}

}
```

19] WAP to find GCD for given value :

GCD means

Example : n1 = 6 ----------→ 1,2,3,6

n2 = 12----------→ 1, 2, 3 ,4,6,12

GCD ===→ 1,2,3,6

```
package preprogramingM1;

import java.util.Scanner;

public class GcdNumbers {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the number : ");

int n1 = scanner.nextInt();

System.out.println("Enter the number : ");

int n2 = scanner.nextInt();

System.out.println("all GCD number of " + n1 + " and " + n2);

for (int i = 1; i <= n1 && i <= n2; i++) {

if (n1 % i == 0 && n2 % i == 0) {

System.out.print(i + " ");

}

}

}

}
```

20] WAP to check number is Armstrong number or not

Example : -

A] find total digit in number

b] Extract digit and find pow of each digit

c] add the each digit power

num = 145

5 ===➔ 5 * 5 * 5 == 125

4 * 4 * 4 == 64 total is :=====190

1 * 1 * 1== 1

190 == 145 =======➔ no Armstrong number

```
package preprogramingM1;

import java.util.Scanner;

public class ArmstorngNumber {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the number");

int num = scanner.nextInt();

int temp = num;

int count = 0;

int sum = 0;

for (int i = num; i > 0; i = i / 10) {

count++;

}

System.out.println(count);

while (num > 0) {

int rem = num % 10;

int pow = 1;

for (int i = 1; i <= count; i++) {

pow = pow * rem;

}

sum = sum + pow;

num = num / 10;

}

if (sum == temp) {

System.out.println("Number is Amrostorng number");

} else {

System.out.println("Number is not armostorng number");

}

}

}
```

21] WAP to find the square root of given number :

```
package preprogramingM1;
```

```
import java.util.Scanner;

public class SquareRoot {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the numbdr ");

int num = scanner.nextInt();

for (int i = 1; i < num / 2; i++) {

if ((i * i) == num) {

System.out.println("Square root is : " + i);

break;

} else if (i * i > num) {

System.out.println("square is in decimal format");

break;

}

}

}

}
```

Pattern Program

1] WAP to print following pattern

```
*

* *

* * *

* * * *

* * * * *
```

```
public class PatternType2 {

public static void main(String[] args) {

int num = 5;

System.out.println(

"====================================== Left Upper Trangle ========================================");

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

for (int j = 1; j <= num; j++) {

if (j <= i)

System.out.print(" * ");

else

System.out.print(" ");

}

System.out.println();

}

}
```

```
}
```

2] WAJP to print following Pattern

```
*

* *

* * *

* * * *

* * * * *

System.out.println(

"==================================== Right Upper Trangle ========================================");

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

for (int j = 1; j <= num; j++) {

if (j + i >= num + 1)

System.out.print(" * ");

else

System.out.print(" ");

}

System.out.println();

}
```

3] WAJP to print following pattern

```
* * * * *

* * * *

* * *

* *

*

System.out.println(

"==================================== Left Down Trangle ========================================");

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

for (int j = 1; j <= num; j++) {

if (i + j <= num + 1)

System.out.print(" * ");

else

System.out.print(" ");

}

System.out.println();

}
```

4] WAJP to print following pattern

```
* * * * *

* * * *

* * *
```

```
* *

*

System.out.println(

"=================================== Right Down Trangle ========================================");

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

for (int j = 1; j <= num; j++) {

if (j >= i)

System.out.print(" * ");

else

System.out.print(" ");

}

System.out.println();

}
```

5] WAJP to print following pattern

```
* * * * * * * * *

* * * * * * * *

* * * * * *

* * * *

* *

System.out.println(

"=================================== (Left Down Trangle) + (Right Down Trangle)
========================================");

for (int i = 1; i <= num * 2 - 1; i++) {

for (int j = 1; j <= num * 2 - 1; j++) {

if (i + j <= num + 1 || j - i >= num - 1)

System.out.print(" * ");

else

System.out.print(" ");

}

System.out.println();

}
```

6] WAJP to print following pattern

```
* *

* * * *

* * * * * *

* * * * * * * *

* * * * * * * * *

System.out.println(

"=================================== (Right Upper Trangle) + (Left Upper Trangle)
========================================");
```

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

for (int j = 1; j <= num * 2 - 1; j++) {

if (i + j >= num * 2 || i >= j)

System.out.print(" * ");

else

System.out.print(" ");

}

System.out.println();

}
```

7] WAJP to print following pattern

```
* * * * *

* * * *

* * *

* *

*
```

```
*

* *

* * *

* * * *

System.out.println(

"======================================= (Left Down Trangle) + (Left Upper Trangle)
========================================");

for (int i = 1; i <= num * 2 - 1; i++) {

for (int j = 1; j <= num; j++) {

if (i + j <= num + 1 || i - j >= num)

System.out.print(" * ");

else

System.out.print(" ");

}

System.out.println();

}
```

8] WAJP to print Following pattern

```
* * * * *

* * * *

* * *

* *

*
```

```
* *

* * *
```

```
* * * *

* * * * *

System.out.println(

"====================================== (Rigth Down Trangle) + (Rigth Upper Trangle)
========================================");

for (int i = 1; i <= num * 2 - 1; i++) {

for (int j = 1; j <= num; j++) {

if (j >= i || i + j >= num * 2)

System.out.print(" * ");

else

System.out.print(" ");

}

System.out.println();

}
```

9] WAJP to print following pattern

```
*

* * *

* * * * *

* * * * * * *

System.out.println("================================Upper prymid=====================");

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

for (int j = 1; j <= num * 2 - 1; j++) {

if (i + j >= num + 1 && j - i <= num - 1) {

System.out.print(" * ");

} else {

System.out.print(" ");

}

}

System.out.println();

}
```

Q10] WAJP to print following pattern

```
* * * * * * *

* * * * *

* * *

*

System.out.println("================================ Down prymid=======================================");

System.out.println();

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

for (int j = 1; j <= num * 2 - 1; j++) {
```

```
if (i <= j && i + j <= num * 2)
System.out.print(" * ");
else
System.out.print(" ");
}
System.out.println();
}
```

Q11] WAJP to print following pattern

```
*
* *
* * *
* * * *
* * *
* *
*
System.out.println("================================ Left prymid=========================================");
System.out.println();
for (int i = 1; i <= num * 2 - 1; i++) {
for (int j = 1; j <= num; j++) {
if (i >= j && i + j <= num * 2)
System.out.print(" * ");
else
System.out.print(" ");
}
System.out.println();
}
```

Q12] WAJP to print following pattern

```
*
* *
* * *
* * * *
* * *
* *
*
System.out.println("================================ Right prymid=========================================");
System.out.println();
for (int i = 1; i <= num * 2 - 1; i++) {
for (int j = 1; j <= num; j++) {
if (i + j >= num + 1 && i - j <= num - 1)
```

```
System.out.print(" * ");
else
System.out.print(" ");
}
System.out.println();
}
```

14] WAJP to print following pattern

```
* * * *
* * * *
* * * *
* * * *
System.out.println("================================= Square ========================================");
System.out.println();
for (int i = 1; i <= num; i++) {
for (int j = 1; j <= num * 2 - 1; j++) {
if (i + j <= num * 2 && i + j >= num + 1)
System.out.print(" * ");
else
System.out.print(" ");
}
System.out.println();
}
```

15] WAJP to print following pattern

```
*
* * *
* * * * *
* * * * * * *
* * * * *
* * *
*
System.out.println();
System.out.println("================================= Diamond ========================================");
System.out.println();
for (int i = 1; i <= num * 2 - 1; i++) {
for (int j = 1; j <= num * 2 - 1; j++) {
if (i + j >= num + 1 && j - i <= num - 1 && i - j <= num - 1 && i + j <= num * 3 - 1)
System.out.print(" * ");
else
System.out.print(" ");
```

```
}
System.out.println();
}
```

16] WAJP to print following pattern

```
* *
* * * *
* * * * * *
* * * * * * *
* * * * * *
* * * *
* *
```

```
System.out.println("================================ Butterfly ========================================");
System.out.println();
for (int i = 1; i <= num * 2 - 1; i++) {
for (int j = 1; j <= num * 2 - 1; j++) {
if ((i >= j && i + j <= num * 2) || (i + j >= num * 2 && j >= i))
System.out.print(" * ");
else
System.out.print(" ");
}
System.out.println();
}
```

17] WAJP to print following pattern

```
* * * * * * *
* * * * *
* * *
*
* * *
* * * * *
* * * * * * *
```

```
System.out.println();
System.out.println("================================ Butterfly ========================================");
System.out.println();
for (int i = 1; i <= num * 2 - 1; i++) {
for (int j = 1; j <= num * 2 - 1; j++) {
if ((j >= i && i + j <= num * 2) || (j <= i && i + j >= num * 2))
System.out.print(" * ");
else
System.out.print(" ");
```

```
}

System.out.println();

}
```

18] WAJP to print following pattern

```
* * * * * *

* *

* *

* * * * * * * * *

* *

* *

* * * * * *
```

```
System.out.println("================================ sastik ========================================");

System.out.println();

for (int i = 1; i <= num * 2 - 1; i++) {

for (int j = 1; j <= num * 2 - 1; j++) {

if (i == num || j == num || (i == 1 && j >= num) || (i == num * 2 - 1 && j <= num)

|| (j == 1 && i <= num) || (j == num * 2 - 1 && i >= num))

System.out.print(" * ");

else

System.out.print(" ");

}

System.out.println();

}
```