ASSIGNMENT - 1

1 . Area of a rectangle program in java :

Program :

```
import java.util.*;

class rectangle{

public static void main(String [] x)

{

Scanner sc = new Scanner (System.in);

System.out.println("Enter length : ");

int l=sc.nextInt();

System.out.println("Enter breadth: ");

int b=sc.nextInt();

System.out.println("Area of the rectangle is "+l*b);

}

}
```

Output :

```
Enter length :

12

Enter breadth:

34

Area of the rectangle is 408
```

2. Program to find a given number is Armstrong or not :

Program :

```
import  java.util.*;

class  amstrong{

public static void main(String [] x)

{

Scanner sc = new Scanner (System.in);

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

int n=sc.nextInt();

int k=n;

int r;

double  s=0;

while  (n>0){

        r=n%10;

        s=s+Math.pow(r,3);

        n=n/10;

}

if(s==k)

System.out.println("Give number is an armstrong number");

else

System.out.println("Give number is not an armstrong number");

}

}
```

Output :

Enter the number :

153

Give number is an armstrong number

Enter the number :

44

Give number is not an armstrong number

3 . Program to find whether a number is palindrome or not :

Program :

```
import java.util.*;
class palindrome{
public static void main(String [] x)
{
Scanner sc = new Scanner(System.in);
System.out.println("Enter the number : ");
int n=sc.nextInt();
int k=n,r,s=0;
while(n>0){
        r=n%10;
        s=(s*10)+r;
        n=n/10;}
if(s==k)
System.out.println("It is a palindrome number");
```

```
else
System.out.println("It is not a palindrome number");
}}
```

Output :

```
Enter the number :
121
It is a palindrome number
Enter the number :
2322
It is not a palindrome number
```

4 . Program to generate first N numbers :

Program :

```
import java.util.*;
class primeseries{
public static boolean prime(int n){
int i,s=0;
for(i=2;i<n/2;i++)
{
if(n%i==0)
s+=1;
}
if (s==0)
return true;
```

```
else

return false;

}

public static void main(String [] x)

{

Scanner sc = new Scanner(System.in);

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

int n=sc.nextInt(),i=2,s=0;

while(s<n){

if(prime(i)){

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

s+=1;

}

i++;

}

}

}
```

Output :

```
Enter the number :

100

2    3    4    5    7    11    13    17    19    23    29    31    37    41    43    47    53    59    61
67    71    73    79    83    89    97    101    103    107    109    113    127    131    137    139    149
151    157    163    167    173    179    181    191    193    197    199    211    223    227    229    233
239    241    251    257    263    269    271    277    281    283    293    307    311    313    317    331
337    347    349    353    359    367    373    379    383    389    397    401    409    419    421    431
433    439    443    449    457    461    463    467    479    487    491    499    503    509    521    523
```

5 . Program to find even numbers between to numbers :

Program :

```
import java.util.*;

class evennumbers{

public static void main(String [] x)

{

Scanner sc = new Scanner(System.in);

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

int n1=sc.nextInt();

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

int n2=sc.nextInt();

System.out.println("The even numbers between "+n1+" and "+n2+"are :");

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

if(i%2==0)

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

}

}
```

Output :

```
Enter the number one :

100

Enter the number two :

500

The even numbers between 100 and 500are :

102  104  106  108  110  112  114  116  118  120  122  124  126  128  130  132  134  136  138
140  142  144  146  148  150  152  154  156  158  160  162  164  166  168  170  172  174  176
178  180  182  184  186  188  190  192  194  196  198  200  202  204  206  208  210  212  214
```

216 218 220 222 224 226 228 230 232 234 236 238 240 242 244 246 248 250 252
254 256 258 260 262 264 266 268 270 272 274 276 278 280 282 284 286 288 290
292 294 296 298 300 302 304 306 308 310 312 314 316 318 320 322 324 326 328
330 332 334 336 338 340 342 344 346 348 350 352 354 356 358 360 362 364 366
368 370 372 374 376 378 380 382 384 386 388 390 392 394 396 398 400 402 404
406 408 410 412 414 416 418 420 422 424 426 428 430 432 434 436 438 440 442
444 446 448 450 452 454 456 458 460 462 464 466 468 470 472 474 476 478 480
482 484 486 488 490 492 494 496 498

Theoretical Questions :

1.Abstraction :

Abstraction is a process of hiding the implementation details and showing only functionality to the user. It displays just the essential things to the user and hides the internal information .

2.Encpsulation :

It is the technique of making the fields in a class private and providing access to the fields via public methods. If a field is declared private, it cannot be accessed by anyone outside the class, thereby hiding the fields within the class. Therefore encapsulation is also referred to as data hiding .

3 . Jdk :

JDK is an acronym for Java Development Kit. It is a software development environment which is used to develop Java applications and applets. It physically exists. It contains JRE + development tools.

4 . Jvm :

JVM is an acronym for Java Virtual Machine; it is an abstract machine which provides the runtime environment in which Java bytecode can be executed. JVM is platform dependent .

5 . Inheritance:

Inheritance is a mechanism by which one object acquires all the properties and behavior of another object of another class. It is used for Code Reusability and Method Overriding.

6 . How java achieved platform Independence :

Java is platform independent because the Java compiler converts the source code to bytecode, which is Intermediate Language. Bytecode can be executed on any platform (OS) using JVM(Java Virtual Machine).

7.Syntax of main function :

public void main(String [] args)

8. Conditional Operator :

The conditional operator is also known as the ternary operator. This operator consists of three operands and is used to evaluate Boolean expressions. The goal of the operator is to decide; which value should be assigned to the variable.

The operator is written as:

Variable x= (expression) ? value if true : value if false .

9 . Data Types in Java :

Data Types are dived into two :

1. Primitive
2. Non Primitive

There are 8 types of primitive data types:

- boolean data type
- byte data type
- char data type
- short data type
- int data type
- long data type
- float data type
- double data type

Non Primitive Data Types : Classes , Interfaces , Arrays etc..

10. Constant :

Constant is used to declare a variable to make its values does not change and it is declared by using final and static keywords before the variable.

Static final a =100.