Module-1

Assignment

Question-1) Write a programme to generate 9's table.

CODE:

```
package Modele_1;
import java.io.*;
public class multiples_of_9 {
	public static void main(String args[])throws IOException{
		InputStreamReader isr=new InputStreamReader(System.in);
		BufferedReader br=new BufferedReader(isr);
		System.out.println("Enter the number: ");
		int n=Integer.parseInt(br.readLine());
		int i,p=1;
		for(i=1;i<=n;i++) {
			p=9*i;
			System.out.println(9+"x"+i+"="+p);
		}
	}
}
```

OUTPUT:

```
Enter the number:
10
9x1=9
9x2=18
9x3=27
9x4=36
9x5=45
9x6=54
9x7=63
9x8=72
9x9=81
9x10=90
```

Question-2) Create an array of 10 integers and print only the even values.

CODE:

```
package Modele_1;
```

```
import java.io.*;
public class array_of_even_numbers {
	public static void main(String args[])throws IOException{
		InputStreamReader isr=new InputStreamReader(System.in);
		BufferedReader br=new BufferedReader(isr);
		System.out.println("Enter the number of elements in the array:
");
		int n=Integer.parseInt(br.readLine());
		int ar[]=new int[n];
		int i;
		System.out.println("Enter the elements in the array: ");
		for(i=0;i<n;i++) {
			ar[i]=Integer.parseInt(br.readLine());
		}
		System.out.println("The even elements in the array: ");
		for(i=0;i<n;i++) {
			if(ar[i]%2==0) {
				System.out.println(ar[i]);
			}
		}
	}
}
```

OUTPUT:

```
Enter the number of elements in the array:
5
Enter the elements in the array:
10
15
17
16
13
The even elements in the array:
10
16
```

Question-3) Create an integer array of m rows and n columns (where m, n < 10) and print only the odd values.

CODE:

```
package Modele_1;
import java.io.*;
public class odd_number_matrix {
	public static void main(String args[])throws IOException{
		InputStreamReader isr=new InputStreamReader(System.in);
		BufferedReader br=new BufferedReader(isr);
		System.out.println("Enter the number of rows and columns in the
matrix: ");
		int m=Integer.parseInt(br.readLine());
```

```
            int n=Integer.parseInt(br.readLine());
            int ar[][]=new int[m][n];
            int i,j;
            System.out.println("Enter the elements inside the matrix: ");
            for(i=0;i<m;i++) {
                  for(j=0;j<n;j++) {
                        ar[i][j]=Integer.parseInt(br.readLine());
                  }
            }
            System.out.println("The old elements in the matrix: ");
            for(i=0;i<m;i++) {
                  for(j=0;j<n;j++) {
                        if(ar[i][j]%2!=0) {
                              System.out.println(ar[i][j]);
                        }
                  }
            }
      }
}
```

OUTPUT:

```
Enter the number of rows and columns in the matrix:
3
3
Enter the elements inside the matrix:
1
2
3
4
5
6
7
8
9
The old elements in the matrix:
1
3
5
7
9
```

Question-4) You need to print integers till 20, which loop construct is the best for this?

Ans: We can use for loop, while loop as well as do-while loop for this task. But I would prefer to use for loop for this particular task.

CODE:

```java
package Modele_1;
import java.io.*;
public class printing_integers {
	public static void main(String args[])throws IOException{
		InputStreamReader isr=new InputStreamReader(System.in);
		BufferedReader br=new BufferedReader(isr);
		System.out.println("Enter the number: ");
		int n=Integer.parseInt(br.readLine());
		int i;
		System.out.printf("The numbers upto %d: ",n);
		for(i=1;i<=n;i++) {
			System.out.println(i);
		}
	}
}
```

OUTPUT:

```
Enter the number:
20
The numbers upto 20: 1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
```

Question-5 & 6) Create 2 integer matrices of m rows and n column each and add these 2 matrices.

For the above problem add the relevant code to check valid inputs. Hint: To add matrices they must be of equal dimension. [Solution not provided for this one]

CODE:

```
package Modele_1;
import java.io.*;
public class matrix_addition {
	public static void main(String args[])throws IOException{
		InputStreamReader isr=new InputStreamReader(System.in);
		BufferedReader br=new BufferedReader(isr);
		System.out.println("Enter the number of rows and columns of
first matrix: ");
		int m=Integer.parseInt(br.readLine());
		int n=Integer.parseInt(br.readLine());
		System.out.println("Enter the number of rows and columns of
second matrix: ");
		int p=Integer.parseInt(br.readLine());
		int q=Integer.parseInt(br.readLine());
		if(m!=p||n!=q) {
			System.out.println("Addition of matrices not possible");
		}
		else {
			int a[][]=new int[m][n];
			int b[][]=new int[p][q];
			int c[][]=new int[m][n];
			int i=0,j=0;
			System.out.println("Enter the elements of the first
matrix: ");
			for(i=0;i<m;i++) {
				for(j=0;j<n;j++) {
					a[i][j]=Integer.parseInt(br.readLine());
				}
			}
			System.out.println("Enter the elements of the second
matrix: ");
			for(i=0;i<p;i++) {
				for(j=0;j<q;j++) {
					b[i][j]=Integer.parseInt(br.readLine());
				}
			}
			System.out.println("The resultant matrix is: ");
			for(i=0;i<m;i++) {
				for(j=0;j<n;j++) {
					c[i][j]=a[i][j]+b[i][j];
				}
			}
			for(i=0;i<m;i++) {
				for(j=0;j<n;j++) {
					System.out.print(c[i][j]+"\t");
				}
				System.out.println();
			}
		}
	}
}
```

OUTPUT:

```
Enter the number of rows and columns of first matrix:
2
3
Enter the number of rows and columns of second matrix:
2
3
Enter the elements of the first matrix:
1
2
3
4
5
6
Enter the elements of the second matrix:
6
5
4
3
2
1
The resultant matrix is:
7     7     7
7     7     7
```