   BANK CUSTOMER

#include<iostream>  
using namespace std;  
class details  
{  
private:  
string name;  
int accno;  
int bal;  
  
public:  
void getdata(void);  
void printdata(void)  
{  
cout<<"Customer name : "<<name<<endl;  
cout<<"Account number : "<<accno<<endl;  
cout<<"Bank balance : "<<bal<<endl;  
}  
};  
int main()  
{  
details s1;  
s1.getdata();  
s1.printdata();  
  
details s2;  
s2.getdata();  
s2.printdata();  
  
details s3;  
s3.getdata();  
s3.printdata();  
  
return 0;  
}  
  
void details::getdata(void)  
{  
cout<<"Enter the Customer Name : "<<endl;  
cin>>name;  
cout<<"Enter the Account number  : "<<endl;  
cin>>accno;  
cout<<"Enter the Bank Balance : "<<endl;  
cin>>bal;  
}

   NATURAL NUMBER

#include<iostream>  
using namespace std;  
int sum(int num)  
{  
int result=0;  
while(num!=0)  
{  
result=result+num;  
num--;  
}  
return result;  
}int main()  
{  
int x,a;  
  
cout<<"Enter the INTEGER number :"<<endl;  
cin>>x;  
  
a=sum(x);  
cout<<"Sum of numbers upto: "<<x<<" is "<<a;  
return 0;  
}

SWAP NUMBERS

#include<iostream>  
using namespace std;  
int swapNums(int &x,int &y);  
int main()  
{  
    int firstNum = 20, secondNum =10;  
    cout<<"Before swaping \n";  
    cout<<"firstNum= "<<firstNum<<endl;  
    cout<<"secondNum= "<<secondNum<<endl;  
  
    // call the function to swap the numbers  
    swapNums(firstNum , secondNum );  
cout<<"After swaping \n";  
cout<<"firstNum= "<<firstNum<<endl;  
    cout<<"secondNum= "<<secondNum<<endl;  
return 0;  
}  
int swapNums(int &x , int &y)  
{  
int z;  
z = x;  
x = y;  
y = z;  
  
return x,y;  
}

  ROBOT  
  
  
#include<iostream>  
using namespace std;  
int main()  
{  
char key;  
cout<<"Enter a Key :";  
cin>>key;  
  
switch(key)  
{  
case 'a' :  
cout<<"MOVE LEFT" ;  
break;  
  
case 'b' :  
cout<<"MOVE RIGHT" ;  
break;  
  
case 'c' :  
cout<<"MOVE FORWARD " ;  
break;  
  
case 'd' :  
cout<<"MOVE BACKWORD" ;  
break;  
  
case 'e' :  
cout<<"JUMP" ;  
break;  
  
case 'f' :  
cout<<"STOP" ;  
break;  
  
default :  
cout<<"Check Your Instruction";  
}  
return 0;  
}

                                  SMALL LARGE NO IN ARRAY

#include<iostream>  
using namespace std;  
int main()  
{  
int array[10] = {2,5,31,54,74,66,99,88,12,50};  
int largest;  
largest = array[0];  
  
for(int i=1 ; i<=9 ; i++)  
{  
if(largest>array[i])  
{  
continue;  
}  
else  
{  
largest=array[i];    
}  
}  
  
cout<<"LARGEST  number is : "<<largest<<endl;  
  
  
int smallest;  
smallest = array[0];  
  
for(int i=1 ; i>=9 ; i++)  
{  
if(smallest<array[i])  
{  
continue;  
}  
else  
{  
smallest=array[i];  
}  
}  
  
cout<<"SMALLEST number is : "<<smallest<<endl;  
  
return 0;

}

OCCURANCE OF NUMBER

#include<iostream>  
using namespace std;  
int main()  
{  
int num,count=0;  
int array[10] = {2,5,6,9,2,5,4,7,2};  
  
cout<<"ENTER THE NUMBER :";  
cin>>num;  
  
for(int i=0 ; i<9 ;i++)  
{  
if(num == array[i])  
{  
count++;  
}  
else  
{  
   continue;  
}  
}  
  
cout<<num<<" is OCCURED "<<count<< "TIMES" ;  
  
return 0;

}

                                                   ASCENDING ORDER  
  
  
public class ascending  
{      
    public static void main(String[] args)  
{          
             
        //Initialize array      
        int [] arr = new int [] {5, 2, 8, 7, 1};      
        int temp = 0;      
             
        //Displaying elements of original array      
        System.out.println("Elements of original array: ");      
        for (int i = 0; i < arr.length; i++)  
{      
            System.out.print(arr[i] + " ");      
        }      
             
        //Sort the array in ascending order      
        for (int i = 0; i < arr.length; i++)  
{      
            for (int j = i+1; j < arr.length; j++)  
{      
               if(arr[i] > arr[j])  
  {      
                   temp = arr[i];      
                   arr[i] = arr[j];      
                   arr[j] = temp;      
               }      
            }      
        }      
           
        System.out.println();      
             
        //Displaying elements of array after sorting      
        System.out.println("Elements of array sorted in ascending order: ");      
        for (int i = 0; i < arr.length; i++)  
{      
            System.out.print(arr[i] + " ");      
        }      
    }      
}

 METHOD OVER LOADING

                        public class method

{  
Static method  
static void mystaticMethod()  
    System.out.println("This is Empty Method");  
  
Public method  
public void myPublicMethod(String sharib)  
    System.out.println("My name is "+ sharib);  
  
myMethod(int n1, int n2)  
  
return n1+n2;  
  
Main Method    
  
 public static void main(String[] args)  
 {  
   mystaticMethod():  
   method myObj = new  method();  
   myObj.myPublicMethod("sharib");  
   
   
   add = myMethod(5,25);  
   
      System.out.println("Addition is " + add);  
  }

}

    BANK  CUSTOMER

#include<iostream>  
using namespace std;  
class details  
{  
private:  
string name;  
int accno;  
int bal;  
  
public:  
void getdata(void);  
void printdata(void)  
{  
cout<<"Customer name : "<<name<<endl;  
cout<<"Account number : "<<accno<<endl;  
cout<<"Bank balance : "<<bal<<endl;  
}  
};  
int main()  
{  
details s1;  
s1.getdata();  
s1.printdata();  
  
details s2;  
s2.getdata();  
s2.printdata();  
  
details s3;  
s3.getdata();  
s3.printdata();  
  
return 0;  
}  
  
void details::getdata(void)  
{  
cout<<"Enter the Customer Name : "<<endl;  
cin>>name;  
cout<<"Enter the Account number  : "<<endl;  
cin>>accno;  
cout<<"Enter the Bank Balance : "<<endl;  
cin>>bal;  
}