In previous examples we have taken 1 variable for 1 value.

Like if we have to add 2 values we took 2 variables like x,y.

So, if we have to add 10 numbers and print total of it

Should we take 10 variables?

We should be able to manage a large collection using a single identity.

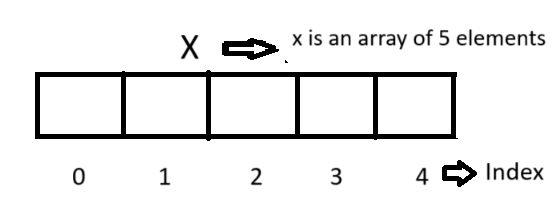
Then make use of Array.

**Array**

**Int x[5]** => x is an array of 5 elements with indexed from 0 to 4.

How much memory will allocate>

According to 5 int.



Base index is 0

X[0] = 57 -> Assign 57 to x of 0

X[1] = 39 -> Assign 30 to x of 1

At the time of declaration when we write x[5] it mens we create an array of length 5 this is not x of 5.

When we assign value and write x[0] that means x of 0.

#include<stdio.h>

int main(){

int x[10], y, t;

y = 0;

while(y<=9){

printf("Enter a number: ");

scanf("%d", &x[y]);

y++;

}

y = 0;

t = 0;

while(y<=9){

t = t+x[y];

y++;

}

printf("Result: %d\n", t);

return 0;

}

Find second Latest value

#include<stdio.h>

int main()

{

int x[10], largest, smallest, secondLastest, y;

y = 0;

while(y<=9){

printf("Enter a number: ");

scanf("%d", &x[y]);

y++;

}

y = 0;

largest = x[0];

smallest = x[0];

while(y<=9){

if(x[y] > largest)

{

largest = x[y];

}

if(x[y] < smallest)

{

smallest = x[y];

}

y++;

}

if(largest == smallest)

{

printf("All values are same, hence we can not determine second Largest value\n");

}

else

{

secondLastest = smallest;

y = 0;

while(y<=9)

{

if(x[y]>secondLastest && x[y]!= largest)

{

secondLastest = x[y];

}

y++;

}

printf("Second Lastest is %d\n", secondLastest);

}

printf("Largest is %d\n", largest);

printf("Smallest is %d\n", smallest);

return 0;

}