

Variables

- Data Types
- Variables
- Constants
- Type
- Input
- Doubts

Data Type: Property
of the variable

Tells the compiler :

"what type of data it can store"
& "how much size is needed"

Primitive data types :

int
4 Bytes

numbers without
decimals

547

-321

Large nos. :

e.g. $20 * 19 * 18 * 17 * 16 * 15 * 14 * \dots$

long
8 Bytes

Integer Values

Numbers with decimal :

float , double
↓ ↓
4 Bytes 8 Bytes

452.71

bool : true / false

char : one single character

a - z

A - Z

0 - 9

special character

a b A B 1 2 @ \$

Size of any data type :

```
cout << sizeof (int);
```

```
sizeof (char);
```

```
sizeof (bool);
```

```
#include <limits.h>
```

INT_MIN

INT_MAX

INT_MIN

INT_MAX

CHAR_MIN

CHAR_MAX

LONG_MIN

LONG_MAX

Derived data type:

string → collection of characters

abc

a1bc

Variables: m/m location which has
a name and it stores
data

Declaration:

datatype name;

int a;

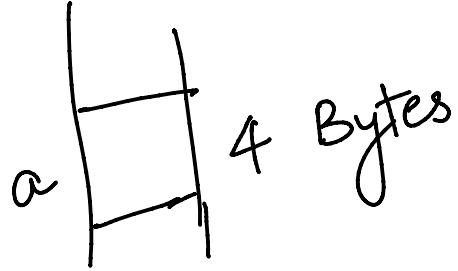
float c;

long b;

double d;

string f;

copy b;
char e;
string f;



int a;
int b; \Rightarrow int a, b;

int a;
double a; \times Same name X

Initialisation: Assign value
for the first
time.

int a;
a = 215; \rightarrow int a = 215;

int a = 72, b = 35;

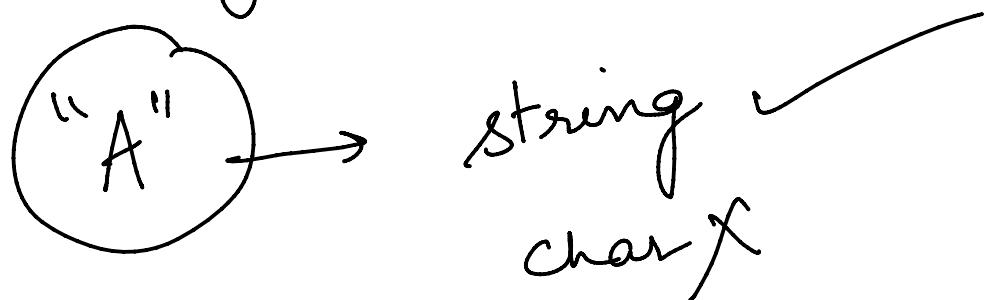
double c = 52.73;

char d = 'A';

string s = "Hello"

char : → single quotes

string : → double quotes



int a = 25;

cout << "a"; → o/p: a

cout << a; → o/p: 25

int a = 72;

cout << "a"; → a

--
cout << "a"; → a

cout << a; → 72

double b = 5.7;

cout << "b";

b

cout << b;

5.7

char ch = '@';

cout << "ch";

ch

cout << ch;

@

string s = "Hello";

cout << "s";

s

cout << s;

Hello

If uninitialized,

we get garbage value

int a = 7
Value of a is 7

String Var
 value

cout << "Value of a is" << a;

Naming Variables :

- | | |
|--------------------------|-------|
| 1.) Lower case alphabets | a - z |
| Upper case alphabets | A - Z |
| Digits | 0 - 9 |
| Underscore | - |

a - b ✓

alzy ✓

A - 3 a ✓

A-3a ✓

a*b ✗

2) Name can't start with digit

2a ✗

- ab ✓

3) Keywords int float if
 else for main
 return

Name should be meaningful .

string s;

string name;

Ab] different
ab]

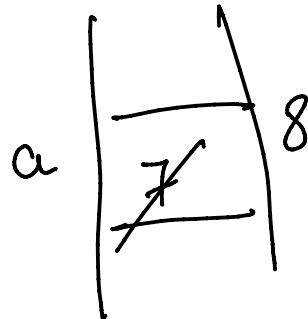
abs J ~ 11

int a = 7;

;

a = 8;

cout << a; → 8



"vary"
↓
change

Constants : Fixed Value
that cannot be
modified in the
Program

const int a = 5;

Or

int const a = 5;

;

```
;  
a = 7;    // error
```

Input from User :

```
int a = 56;
```

```
int b;           double c;  
cin >> b;       cin >> c
```

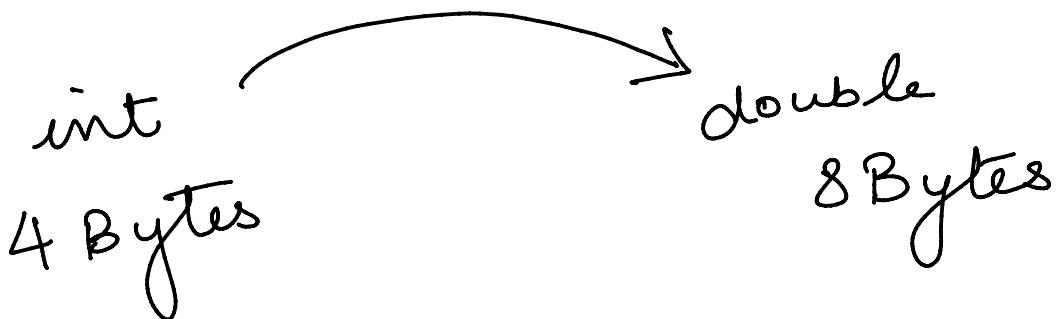
```
int e;           cin >> e >> f;  
double f;
```

```
int a;  
cin >> a;      // input  
cout << a;      // output
```

Type Casting :

Type casting =

changing the datatype



Widening type conversion

int a = 5;

double b = a;

char ch = 'A';

int i = ch; // ascii value

Narrowing type casting :



Data loss

double a = 5.8913;

int b = a;

// $b = 5$

Conversion to string :

to_string()

int a = 72;

string s = to_string(a); "72"

double b = 23.42;

string s1 = to_string(b); "23.42"

String to Integer stoi

String to Integer Show

"123" → 123

"1a2" → 1

"Hello" → Error

string to double stod

string to float stof

int

```
stringToInt (string str){  
    int a = stoi (str);  
    return a;  
}
```