# Contents

- Input strings & getline() function
- Writing C++ code without standart *namespace* library
- Determine to max/min value
- <cmath>

# Ex-1: input strings

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string fullName;
    cout << "Type your full name: ";
    cin >> fullName;
    cout << "Your name is: " << fullName;
    return 0;
}
```

```
Type your full name: …
Your name is:….
```

# Ex-2: getline() function

```
#include <iostream>
#include <string>
using namespace std;
int main(){
string fullName;
cout << "Type your full name: ";
getline (cin, fullName);
cout << "Your name is: " << fullName;
    return 0;
}
```

Type your full name: …
Your name is:….

# Ex-3,4: Adding numbers & string

```
#include <iostream>
using namespace std;
int main(){
    int x = 10;
    int y = 20;
    int z = x+y;
    cout << z ;
    return 0;
}
```

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string x = "10" ;
    string y = "20" ;
    string z = x + y;
    cout << z;
    return 0;
}
```

# Ex-3,4: Adding numbers & string

```
#include <iostream>
using namespace std;
int main(){
    int x = 10;
    int y = 20;
    int z = x+y;
    cout << z ;
    return 0;
}
```

```
30
```

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string x = "10" ;
    string y = "20" ;
    string z = x + y;
    cout << z;
    return 0;
}
```

```
1020
```

# Ex-5: without *namespace*

```
#include <iostream>
#include <string>
int main(){
      std :: string greeting = “hello”;
      std :: cout  << greeting;
      return 0;
}
```

**std** using

**::** using

```
hello
```

# Ex-6: Determining max/min values

```
#include <iostream>
#include <string>
using namespace std;
int main(){
        cout <<  max(10, 100)  << "\n";
        cout <<  min(10, 100) ;
         return 0;
}
```

```
100
10
```

# <cmath>

# Ex-7: <cmath>header

```
#include <iostream>
#include <cmath>
using namespace std;

int main() {
  cout << sqrt(100) << "\n";
  cout << round(7.2) << "\n";
  cout << log(2) << "\n";
  return 0;
}
```

```
10
7
0.693147
```

# Math functions

| | | |
|---|---|---|
| `fabs( x )` | absolute value of x | `fabs( 5.1 )` is `5.1`<br>`fabs( 0.0 )` is `0.0`<br>`fabs( -8.76 )` is `8.76` |
| `floor( x )` | rounds x to the largest integer not greater than x | `floor( 9.2 )` is `9.0`<br>`floor( -9.8 )` is `-10.0` |
| `fmod( x, y )` | remainder of x/y as a floating-point number | `fmod( 2.6, 1.2 )` is `0.2` |
| `log( x )` | natural logarithm of x (base e) | `log( 2.718282 )` is `1.0`<br>`log( 7.389056 )` is `2.0` |
| `log10( x )` | logarithm of x (base 10) | `log10( 10.0 )` is `1.0`<br>`log10( 100.0 )` is `2.0` |
| `pow( x, y )` | x raised to power y ( $x^y$ ) | `pow( 2, 7 )` is `128`<br>`pow( 9, .5 )` is `3` |
| `sin( x )` | trigonometric sine of x (x in radians) | `sin( 0.0 )` is `0` |
| `sqrt( x )` | square root of x (where x is a nonnegative value) | `sqrt( 9.0 )` is `3.0` |
| `tan( x )` | trigonometric tangent of x (x in radians) | `tan( 0.0 )` is `0` |