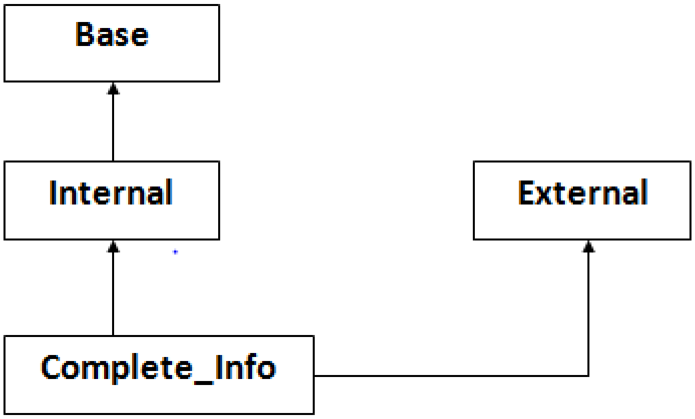
PRACTICE TEST FOR ST-2

35m : 53s
to test end

1
2
3
4
5

INHERITANCE

Consider a class named Base having its data members as name and its roll number.A another class named Internal inherits this class in public mode and having its data members as 4 subject marks as internal.Their is an independent class named External having external marks of 4 subjects.The last class named Complete_Info being derived from External and Internal, calculates the total of internal marks obtained by student as well as sum of external marks and grand total.



Generate a code to display the name ,roll_no,sum of internal marks,sum of external marks, grand total and Grade of student. Grade can be calculated using following criteria.

Total Marks	Grade
less than 40	R
41 to 60	C
61 -80	B
81-100	A

For Example
Input:
Raj
1
2
1

1
2
3
4
5

5

5

4

Output:

Raj 1 5 19 24 R

Explanation:

Input Contains

NAME OF THE STUDENT

Raj

ROLL_NO

1

INTERNAL MARKS IN FOUR SUBJECTS

2

1

1

1

EXTERNAL MARKS IN FOUR SUBJECTS

5

5

5

4

Output contains

Name Roll_no Total _internal_ marks Total _external_ marks Total_marks Grade

Note: All elements in output are separated by space

YOUR ANSWER

We recommend you take a quick tour of our editor before you proceed. The timer will pause up to 90 seconds for the tour. Start tour ✕

Draft saved 10:45 am Original code C++

```
#include ↔

using namespace std;

class Base
```

PRACTICE TEST FOR ST-2

35m : 53s
to test end

1
2
3
4
5

```
      int roll;
      void get();
};

class Internal: public Base
{
    public:
    int M1, M2,M3,M4;
    void getIMarks();
     };
class External
{
    public:
    int E1, E2,E3,E4;
    void getEMarks();
};



void Base::get()
            {
    scanf("%s",&name);
            cin>>roll;
        }
 void Internal::getIMarks()
            {
            cin>>M1>>M2>>M3>>M4;
        }
void External::getEMarks()
            {
            cin>>E1>>E2>>E3>>E4;
        }
class Complete_Info:public Internal,public External
    {
    public:
    void get()
        {
        Base::get();
    }
    void getIMarks()
        {
       Internal::getIMarks();
    }
    void getEMarks()
        {
```

PRACTICE TEST FOR ST-2

35m : 53s
to test end

1

2

3

4

5

```
    void Display()
        {
        //  cout<<"hello";
        int sum,sum1;
        sum=M1+M2+M3+M4;
        sum1=E1+E2+E3+E4;
      int total;
        total=sum+sum1;
        char  gr;
        if(total<=100 && total>=81)
            {
            gr='A';
        }
        if(total<=60 && total>=41)
            {
            gr='C';
        }
        if(total<=80 && total>=61)
            {
            gr='B';
        }
        if(total<40)
            {
            gr='R';
        }
      // printf("%s",name);
        cout<<" "<<roll<<" "<<sum<<" "<<sum1<<" "<<total<<" "<<gr;

    }
};
```

```
int main(void)
{
  Complete_Info c;
  c.get();
  c.getIMarks();
  c.getEMarks();
  c.Display();
   return 0;
}

```

Line: 82 Col: 9

□ Test against custom input

Run Code

Submit code & Continue

PRACTICE TEST FOR ST-2

35m : 53s
to test end

to edit them on windows.

1
2
3
4
5

Status: No test cases passed.

Tip: Debug your code against custom input

Testcase 1: *Wrong Answer*

Input [Download]

```
Raj
1
2
1
1
1
5
5
5
34
```

Your Output

```
 1 5 49 54 C
```

Expected Output [Download]

```
Raj 1 5 49 54 C
```

Testcase 2: *Wrong Answer*

Input [Download]

```
Rohit
1
10
10
10
10
10
20
10
20
```

Your Output

```
 1 40 60 100 A
```

PRACTICE TEST FOR ST-2

35m : 53s
to test end

1

2

3

4

5

About Privacy Policy Terms of Service