

```

1 #include <string>
2 #include <iostream>
3
4 class Weather
5 {
6     private:
7         double temperature;
8         double relative_humidity;
9         double dew_point;
10        std::string description;
11
12    public:
13        // Accessors
14        double get_temperature()
15        {
16            return this->temperature;
17        }
18
19        void set_temperature(double temperature)
20        {
21            this->temperature = temperature;
22        }
23 };
24
25 void print_weather(Weather w)
26 {
27     std::cout << "Temperature: " << w.get_temperature() << std::en
dl;
28     //std::cout << "RH: " << w.relative_humidity << std::endl;
29     //std::cout << "DP: " << w.dew_point << std::endl;
30     //std::cout << "Description: " << w.description << std::endl;
31 }
32

```

```

ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
RH: 40
DP: 10
Description: cloudy
ben@computer home $ g++ weather.cpp -o weather
weather.cpp: In function 'void print_weather(Weather)':
weather.cpp:17:39: error: 'double Weather::temperature' is
private within this context
17 |     std::cout << "Temperature: " << w.temperature <<
std::endl;
|
weather.cpp:7:16: note: declared private here
7 |         double temperature;
|
weather.cpp: In function 'int main()':
weather.cpp:26:9: error: 'double Weather::temperature' is
private within this context
26 |     now.temperature = 20.0;
|
weather.cpp:7:16: note: declared private here
7 |         double temperature;
|
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ example.cpp
example.cpp: In constructor 'Weather::Weather(double, dou
ble, std::string)':
example.cpp:15:13: error: 'self' was not declared in this
scope
15 |         self->temperature = temperature;
|
ben@computer home $ g++ example.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ 

```

```

1 #include <string>
2 #include <iostream>
3
4 class Weather
5 {
6     private:
7         double temperature;
8         double relative_humidity;
9         double dew_point;
10        std::string description;
11
12    public:
13        Weather(double temperature, double relative_humidity, std:
14        :)
15        {
16        }
17
18        // Accessors
19        double get_temperature()
20        {
21            return this->temperature;
22        }
23
24        // mutator
25        void set_temperature(double temperature)
26        {
27            this->temperature = temperature;
28        }
29
30    void print_weather(Weather w)
31    {
32        @
-- INSERT --

```

```

ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
RH: 40
DP: 10
Description: cloudy
ben@computer home $ g++ weather.cpp -o weather
weather.cpp: In function 'void print_weather(Weather)':
weather.cpp:17:39: error: 'double Weather::temperature' is
private within this context
17 |     std::cout << "Temperature: " << w.temperature <<
std::endl;
|
weather.cpp:7:16: note: declared private here
7 |         double temperature;
|
weather.cpp: In function 'int main()':
weather.cpp:26:9: error: 'double Weather::temperature' is
private within this context
26 |         now.temperature = 20.0;
|
weather.cpp:7:16: note: declared private here
7 |         double temperature;
|
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ example.cpp
example.cpp: In constructor 'Weather::Weather(double, doubl
le, std::string)':
example.cpp:15:13: error: 'self' was not declared in this
scope
15 |             self->temperature = temperature;
|
ben@computer home $ g++ example.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ 

```

```
1 #include <string>
2 #include <iostream>
3
4 class Weather
5 {
6     private:
7         double temperature;
8         double relative_humidity;
9         double dew_point;
10        std::string description;
11
12    public:
13        Weather(double temperature, double relative_humidity, std::string description)
14        {
15        }
16
17        // Accessors
18        double get_temperature()
19        {
20            return this->temperature;
21        }
22
23        // mutator
24        void set_temperature(double temperature)
25        {
26            this->temperature = temperature;
27        }
28    };
29
30 void print_weather(Weather w)
31 {
32     std::cout << "Temperature: " << w.get_temperature() << std::endl;
33     //std::cout << "RH: " << w.relative_humidity << std::endl;
```

```
7 |     double temperature;
|  
-----  
weather.cpp: In function 'int  
main()':  
weather.cpp:26:9: error: 'double Weather::temperature' is pr  
ivate within this context  
26 |     now.temperature =  
20.0;  
|  
-----  
weather.cpp:7:16: note: declar  
ed private here  
7 |     double temperature;  
|  
-----  
ben@computer home $ g++ weathe  
r.cpp -o weather  
ben@computer home $ ./weather  
Temperature: 4.94066e-324  
ben@computer home $ g++ weathe  
r.cpp -o weather  
ben@computer home $ ./weather  
Temperature: 20  
ben@computer home $ g++ exampl  
e.cpp  
example.cpp: In constructor 'W  
eather::Weather(double, double  
, std::string)':  
example.cpp:15:13: error: 'sel  
f' was not declared in this sc  
ope  
15 |     self->temp  
erature = temperature;  
|  
-----  
ben@computer home $ g++ exampl  
e.cpp  
ben@computer home $ ./a.out  
Temperature: 20  
ben@computer home $
```

```
1 #include <string>
2 #include <iostream>
3
4 class Weather
5 {
6     private:
7         double temperature;
8         double relative_humidity;
9         double dew_point;
10        std::string description;
11
12    public:
13        Weather(double temperature, double relative_humidity, std::string description)
14        {
15            this->temperature = temperature;
16            this->relative_humidity = relative_humidity;
17            this->description = description;
18        }
19
20        // Accessors
21        double get_temperature()
22        {
23            return this->temperature;
24        }
25
26        // mutator
27        void set_temperature(double temperature)
28        {
29            this->temperature = temperature;
30        }
31    };
32
33 void print_weather(Weather w)
```

```
7 |         double tempera
ture;
| |
|-----|
|-----|
weather.cpp: In function 'int
main()':
weather.cpp:26:9: error: 'doub
le Weather::temperature' is pr
ivate within this context
26 |     now.temperature =
20.0;
|           ^
|-----|
weather.cpp:7:16: note: declar
ed private here
7 |         double tempera
ture;
|           ^
|-----|
ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ exampl
e.cpp
example.cpp: In constructor 'W
eather::Weather(double, double
, std::string)':
example.cpp:15:13: error: 'sel
f' was not declared in this sc
ope
15 |             self->temp
erature = temperature;
|           ^
|-----|
ben@computer home $ g++ exampl
e.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $
```

```

9   double dew_point;
10  std::string description;
11
12 public:
13 Weather(double temperature, double relative_humidity, std::string description)
14 {
15     this->temperature = temperature;
16     this->relative_humidity = relative_humidity;
17     this->description = description;
18 }
19
20 // Accessors
21 double get_temperature()
22 {
23     return this->temperature;
24 }
25
26 // mutator
27 void set_temperature(double temperature)
28 {
29     this->temperature = temperature;
30 }
31 };
32
33 void print_weather(Weather w)
34 {
35     std::cout << "Temperature: " << w.get_temperature() << std::endl;
36     //std::cout << "RH: " << w.relative_humidity << std::endl;
37     //std::cout << "DP: " << w.dew_point << std::endl;
38     //std::cout << "Description: " << w.description << std::endl;
39 }
40
41 int main(void)

```

```

7 |         double tempera
ture;
|         ^
|         ~~~~~
weather.cpp: In function 'int
main()':
weather.cpp:26:9: error: 'doub
le Weather::temperature' is pr
ivate within this context
26 |     now.temperature =
20.0;
|     ^
|         ~~~~~
weather.cpp:7:16: note: declar
ed private here
7 |         double tempera
ture;
|         ^
|         ~~~~~
ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ exampl
e.cpp
example.cpp: In constructor 'W
eather::Weather(double, double
, std::string)':
example.cpp:15:13: error: 'sel
f' was not declared in this sc
ope
15 |             self->temper
ature = temperature;
|             ^
|             ~~~
ben@computer home $ g++ exampl
e.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ 

```

```

13 Weather(double temperature, double relative_humidity, std::string description)
14 {
15     this->temperature = temperature;
16     this->relative_humidity = relative_humidity;
17     this->description = description;
18 }
19 // Accessors
20 double get_temperature()
21 {
22     return this->temperature;
23 }
24
25 // mutator
26 void set_temperature(double temperature)
27 {
28     this->temperature = temperature;
29 }
30 }
31 };
32
33 void print_weather(Weather w)
34 {
35     std::cout << "Temperature: " << w.get_temperature() << std::endl;
36     //std::cout << "RH: " << w.relative_humidity << std::endl;
37     //std::cout << "DP: " << w.dew_point << std::endl;
38     //std::cout << "Description: " << w.description << std::endl;
39 }
40
41 int main(void)
42 {
43     Weather now;
44     now.set_temperature(20.0);
45     //now.temperature = 20.0;

```

```

7 |         double tempera
ture;
|         ^
|         ~~~~~
|         |
|         weather.cpp: In function 'int
|         main()':
|         weather.cpp:26:9: error: 'doub
le Weather::temperature' is pr
ivate within this context
26 |     now.temperature =
20.0;
|     ^
|         ~~~~~
|         |
|         weather.cpp:7:16: note: declar
ed private here
7 |             double tempera
ture;
|             ^
|             ~~~~~
|             |
|             ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ exampl
e.cpp
example.cpp: In constructor 'W
eather::Weather(double, double
, std::string)':
example.cpp:15:13: error: 'sel
f' was not declared in this sc
ope
15 |             self->temp
erature = temperature;
|             ^
|             ~~~~~
|             |
ben@computer home $ g++ exampl
e.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ 

```

```

13 Weather(double temperature, double relative_humidity, std::string description)
14 {
15     this->temperature = temperature;
16     this->relative_humidity = relative_humidity;
17     this->description = description;
18 }
19 // Accessors
20 double get_temperature()
21 {
22     return this->temperature;
23 }
24
25 // mutator
26 void set_temperature(double temperature)
27 {
28     this->temperature = temperature;
29 }
30 }
31 };
32
33 void print_weather(Weather w)
34 {
35     std::cout << "Temperature: " << w.get_temperature() << std::endl;
36     //std::cout << "RH: " << w.relative_humidity << std::endl;
37     //std::cout << "DP: " << w.dew_point << std::endl;
38     //std::cout << "Description: " << w.description << std::endl;
39 }
40
41 int main(void)
42 {
43     Weather now(20.0, 50.0, "sunny");
44     now.set_temperature(20.0);
45     //now.temperature = 20.0;

```

```

7 |         double tempera
ture;
|         ^
|_____
|_____
weather.cpp: In function 'int
main()':
weather.cpp:26:9: error: 'doub
le Weather::temperature' is pr
ivate within this context
26 |     now.temperature =
20.0;
|             ^
|_____
weather.cpp:7:16: note: declar
ed private here
7 |         double tempera
ture;
|         ^
|_____
|_____
ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ exampl
e.cpp
example.cpp: In constructor 'W
eather::Weather(double, double
, std::string)':
example.cpp:15:13: error: 'sel
f' was not declared in this sc
ope
15 |         self->temp
erature = temperature;
|             ^
|_____
ben@computer home $ g++ exampl
e.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ 

```

```
13 Weather(double temperature, double relative_humidity, std::string description)
14 {
15     this->temperature = temperature;
16     this->relative_humidity = relative_humidity;
17     this->description = description;
18 }
19
20 // Accessors
21 double get_temperature()
22 {
23     return this->temperature;
24 }
25
26 // mutator
27 void set_temperature(double temperature)
28 {
29     this->temperature = temperature;
30 }
31 };
32
33 void print_weather(Weather w)
34 {
35     std::cout << "Temperature: " << w.get_temperature() << std::endl;
36     //std::cout << "RH: " << w.relative_humidity << std::endl;
37     //std::cout << "DP: " << w.dew_point << std::endl;
38     //std::cout << "Description: " << w.description << std::endl;
39 }
40
41 int main(void)
42 {
43     Weather now(20.0, 50.0, "sunny");
44     print_weather(now);
45 }
```

```
7 |         double tempera
ure;
|           ^
_____
weather.cpp: In function 'int
main()':
weather.cpp:26:9: error: 'doub
le Weather::temperature' is pr
ivate within this context
26 |     now.temperature =
0.0;
|           ^
_____
weather.cpp:7:16: note: declar
ed private here
7 |         double tempera
ure;
|           ^
_____
ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
temperature: 4.94066e-324
ben@computer home $ g++ weathe
r.cpp -o weather
ben@computer home $ ./weather
temperature: 20
ben@computer home $ g++ exampl
e.cpp
example.cpp: In constructor 'W
eather::Weather(double, double
std::string)':
example.cpp:15:13: error: 'sel
f' was not declared in this sc
ope
15 |         self->temp
ature = temperature;
|           ^
_____
ben@computer home $ g++ exampl
e.cpp
ben@computer home $ ./a.out
temperature: 20
ben@computer home $ □
```

```

13     Weather(double temperature, double relative_humidity, std::s
   ring description)
14     {
15         this->temperature = temperature;
16         this->relative_humidity = relative_humidity;
17         this->description = description;
18     }
19
20     // Accessors
21     double get_temperature()
22     {
23         return this->temperature;
24     }
25
26     // mutator
27     void set_temperature(double temperature)
28     {
29         this->temperature = temperature;
30     }
31 }
32
33 void print_weather(Weather w)
34 {
35     std::cout << "Temperature: " << w.get_temperature() << std::endl
36     //std::cout << "RH: " << w.relative_humidity << std::endl;
37     //std::cout << "DP: " << w.dew_point << std::endl;
38     //std::cout << "Description: " << w.description << std::endl;
39 }
40
41 int main(void)
42 {
43     Weather now(20.0, 50.0, "sunny");
44     print_weather(now);

```

```

ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
RH: 40
DP: 10
Description: cloudy
ben@computer home $ g++ weather.cpp -o weather
weather.cpp: In function 'void print_weather(Weather)'
weather.cpp:17:39: error: 'double Weather::temperature'
   is private within this context
  17 |     std::cout << "Temperature: " << w.temperature
    |             ^
    |
weather.cpp:7:16: note: declared private here
  7 |         double temperature;
    |
weather.cpp: In function 'int main()':
weather.cpp:26:9: error: 'double Weather::temperature'
   is private within this context
  26 |     now.temperature = 20.0;
    |             ^
weather.cpp:7:16: note: declared private here
  7 |         double temperature;
    |
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ example.cpp
example.cpp: In constructor 'Weather::Weather(double, d
ouble, std::string)':
example.cpp:15:13: error: 'self' was not declared in th
is scope
  15 |         self->temperature = temperature;
    |             ^
ben@computer home $ g++ example.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ 

```

```

14
15     {
16         this->temperature = temperature;
17         this->relative_humidity = relative_humidity;
18         this->description = description;
19     }
20
21     // Accessors
22     double get_temperature()
23     {
24         return this->temperature;
25     }
26
27     // mutator
28     void set_temperature(double temperature)
29     {
30         this->temperature = temperature;
31     }
32
33
34     int main(void)
35     {
36         Weather now(20.0, 50.0, "sunny");
37         print_weather(now);
38
39         return 0;
40     }

```

7 fewer lines

```

Temperature: 20
RH: 40
DP: 10
Description: cloudy
ben@computer home $ g++ weather.cpp -o weather
weather.cpp: In function 'void print_weather(Weather)':
weather.cpp:17:39: error: 'double Weather::temperature' is
private within this context
    17 |     std::cout << "Temperature: " << w.temperature <
< std::endl;
    |
weather.cpp:7:16: note: declared private here
    7 |         double temperature;
    |
weather.cpp: In function 'int main()':
weather.cpp:26:9: error: 'double Weather::temperature' is
private within this context
    26 |     now.temperature = 20.0;
    |
weather.cpp:7:16: note: declared private here
    7 |         double temperature;
    |
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ example.cpp
example.cpp: In constructor 'Weather::Weather(double, dou
ble, std::string)':
example.cpp:15:13: error: 'self' was not declared in this
scope
    15 |             self->temperature = temperature;
    |
ben@computer home $ g++ example.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ □

```

```

14 {
15     this->temperature = temperature;
16     this->relative_humidity = relative_humidity;
17     this->description = description;
18 }
19
20 // Accessors
21 double get_temperature()
22 {
23     return this->temperature;
24 }
25
26 // mutator
27 void set_temperature(double temperature)
28 {
29     this->temperature = temperature;
30 }
31 void print_weather(Weather w)
32 {
33     std::cout << "Temperature: " << w.get_temperature() << std::endl;
34     //std::cout << "RH: " << w.relative_humidity << std::endl;
35     //std::cout << "DP: " << w.dew_point << std::endl;
36     //std::cout << "Description: " << w.description << std::endl;
37 }
38 };
39
40
41 int main(void)
42 {
43     Weather now(20.0, 50.0, "sunny");
44     print_weather(now);
45
46 more lines

```

```

Temperature: 20
RH: 40
DP: 10
Description: cloudy
ben@computer home $ g++ weather.cpp -o weather
weather.cpp: In function 'void print_weather(Weather)':
weather.cpp:17:39: error: 'double Weather::temperature' is
private within this context
    17 |     std::cout << "Temperature: " << w.temperature <
< std::endl;
    |
ben@computer home $ ./weather
Temperature: 20.0
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ example.cpp
example.cpp: In constructor 'Weather::Weather(double, dou
ble, std::string)':
example.cpp:15:13: error: 'self' was not declared in this
scope
    15 |         self->temperature = temperature;
    |
ben@computer home $ g++ example.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ 

```

```

14
15     {
16         this->temperature = temperature;
17         this->relative_humidity = relative_humidity;
18         this->description = description;
19     }
20
21     // Accessors
22     double get_temperature()
23     {
24         return this->temperature;
25     }
26
27     // mutator
28     void set_temperature(double temperature)
29     {
30         this->temperature = temperature;
31     }
32
33     void print()
34     {
35         std::cout << "Temperature: " << w.get_temperature() << std::endl;
36         //std::cout << "RH: " << w.relative_humidity << std::endl;
37         //std::cout << "DP: " << w.dew_point << std::endl;
38         //std::cout << "Description: " << w.description << std::endl;
39     }
40
41 int main(void)
42 {
43     Weather now(20.0, 50.0, "sunny");
44     print_weather(now);
45 }
```

```

|   ^
|   |
|   |-----^
|   weather.cpp:7:16: note: declared private here
|   7 |       double temperature;
|   |   ^
|   weather.cpp: In function 'int main()'
|   ':
|   weather.cpp:26:9: error: 'double Weather::temperature' is private within
|   this context
|   26 |       now.temperature = 20.0;
|   |   ^
|   weather.cpp:7:16: note: declared private here
|   7 |       double temperature;
|   |   ^
|   ben@computer home $ g++ weather.cpp
|   -o weather
|   ben@computer home $ ./weather
|   Temperature: 4.94066e-324
|   ben@computer home $ g++ weather.cpp
|   -o weather
|   ben@computer home $ ./weather
|   Temperature: 20
|   ben@computer home $ g++ example.cpp
|   example.cpp: In constructor 'Weather
|   ::Weather(double, double, std::string)':
|   example.cpp:15:13: error: 'self' was
|   not declared in this scope
|   15 |           self->temperature;
|   |           ^
|   |-----^
|   ben@computer home $ g++ example.cpp
|   ben@computer home $ ./a.out
|   Temperature: 20
|   ben@computer home $ g++ weather.cpp
|   -o weather
|   ben@computer home $ ./weather
|   Temperature: 20
|   ben@computer home $ 
```

```

14
15     {
16         this->temperature = temperature;
17         this->relative_humidity = relative_humidity;
18         this->description = description;
19     }
20
21     // Accessors
22     double get_temperature()
23     {
24         return this->temperature;
25     }
26
27     // mutator
28     void set_temperature(double temperature)
29     {
30         this->temperature = temperature;
31     }
32
33     void print()
34     {
35         std::cout << "Temperature: " << this->temperature << std::endl;
36         std::cout << "RH: " << w.relative_humidity << std::endl;
37         std::cout << "DP: " << w.dew_point << std::endl;
38         std::cout << "Description: " << w.description << std::endl;
39     }
40
41 int main(void)
42 {
43     Weather now(20.0, 50.0, "sunny");
44     print_weather(now);
45 }
```

```

|   |
|   ^-----|
weather.cpp:7:16: note: declared private here
7 |         double temperature;
|   |
|   weather.cpp: In function 'int main()'
|   :
weather.cpp:26:9: error: 'double Weather::temperature' is private within this context
26 |     now.temperature = 20.0;
|   |
|   weather.cpp:7:16: note: declared private here
7 |         double temperature;
|   |
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ example.cpp
example.cpp: In constructor 'Weather::Weather(double, double, std::string)':
example.cpp:15:13: error: 'self' was not declared in this scope
15 |             self->temperature;
|   |
ben@computer home $ g++ example.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ 
```

```
14
15     {
16         this->temperature = temperature;
17         this->relative_humidity = relative_humidity;
18         this->description = description;
19     }
20
21     // Accessors
22     double get_temperature()
23     {
24         return this->temperature;
25     }
26
27     // mutator
28     void set_temperature(double temperature)
29     {
30         this->temperature = temperature;
31     }
32
33     void print()
34     {
35         std::cout << "Temperature: " << this->temperature << std::endl;
36         std::cout << "RH: " << this->relative_humidity << std::endl;
37         std::cout << "DP: " << this->dew_point << std::endl;
38         std::cout << "Description: " << this->description << std::endl;
39     }
40
41 int main(void)
42 {
43     Weather now(20.0, 50.0, "sunny");
44     now.print();
```

```
|   ^
|   -----
|   weather.cpp:7:16: note: declared private here
|       7 |           double temperature;
|           ^
|   weather.cpp: In function 'int main()':
|   weather.cpp:26:9: error: 'double Weather::temperature' is private within this context
|       26 |           now.temperature = 20.0;
|               ^
|   weather.cpp:7:16: note: declared private here
|       7 |           double temperature;
|           ^
|   ben@computer home $ g++ weather.cpp
|   -o weather
|   ben@computer home $ ./weather
|   Temperature: 4.94066e-324
|   ben@computer home $ g++ weather.cpp
|   -o weather
|   ben@computer home $ ./weather
|   Temperature: 20
|   ben@computer home $ g++ example.cpp
|   example.cpp: In constructor 'Weather::Weather(double, double, std::string)':
|   example.cpp:15:13: error: 'self' was not declared in this scope
|       15 |           self->temperature;
|               |
|               ^
|   ben@computer home $ g++ example.cpp
|   ben@computer home $ ./a.out
|   Temperature: 20
|   ben@computer home $ g++ weather.cpp
|   -o weather
|   ben@computer home $ ./weather
|   Temperature: 20
|   ben@computer home $
```

```

16     this->relative_humidity = relative_humidity;
17     this->description = description;
18 }
19
20 // Accessors
21 double get_temperature()
22 {
23     return this->temperature;
24 }
25
26 // mutator
27 void set_temperature(double temperature)
28 {
29     this->temperature = temperature;
30 }
31
32 void print()
33 {
34     std::cout << "Temperature: " << this->temperature << std
35         ::endl;
36     std::cout << "RH: " << this->relative_humidity << std::e
37         ndl;
38     std::cout << "DP: " << this->dew_point << std::endl;
39     std::cout << "Description: " << this->description << std
40         ::endl;
41 }
42
43 int main(void)
44 {
45     Weather now(20.0, 50.0, "sunny");
46     now.print();

```

```

weather.cpp:17:39: error: 'double Weather::temperature'
is private within this context
17 |     std::cout << "Temperature: " << w.temperature
| << std::endl;
|
weather.cpp:7:16: note: declared private here
7 |         double temperature;
|
weather.cpp: In function 'int main()':
weather.cpp:26:9: error: 'double Weather::temperature'
is private within this context
26 |     now.temperature = 20.0;
|
weather.cpp:7:16: note: declared private here
7 |         double temperature;
|
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 4.94066e-324
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ example.cpp
example.cpp: In constructor 'Weather::Weather(double, d
ouble, std::string)':
example.cpp:15:13: error: 'self' was not declared in th
is scope
15 |             self->temperature = temperature;
| &lt;
ben@computer home $ g++ example.cpp
ben@computer home $ ./a.out
Temperature: 20
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
ben@computer home $ g++ weather.cpp -o weather
ben@computer home $ ./weather
Temperature: 20
RH: 50
DP: 9.88131e-324
Description: sunny
ben@computer home $ █

```

```
1 #include <string>
2 #include <iostream>
3
4 class Weather
5 {
6     private:
7         double temperature;
8         double relative_humidity;
9         double dew_point;
10        std::string description;
11
12    void calculate_dp()
13
14 public:
15    Weather(double temperature, double relative_humidity, std::string
description)
16    {
17        this->temperature = temperature;
18        this->relative_humidity = relative_humidity;
19        this->description = description;
20    }
21
22 // Accessors
23 double get_temperature()
24 {
25     return this->temperature;
26 }
27
28 // mutator
29 void set_temperature(double temperature)
30 {
31     this->temperature = temperature;
32 }
```

```
1 #include <string>
2 #include <iostream>
3
4 class Weather
5 {
6     private:
7         double temperature;
8         double relative_humidity;
9         double dew_point;
10        std::string description;
11
12    void calculate_dp()
13    {
14    }
15
16
17 public:
18    Weather(double temperature, double relative_humidity, std::string
description)
19    {
20        this->temperature = temperature;
21        this->relative_humidity = relative_humidity;
22        this->description = description;
23    }
24
25    // Accessors
26    double get_temperature()
27    {
28        return this->temperature;
29    }
30
31    // mutator
32    void set_temperature(double temperature)
```

ben@computer home \$

```
1 #include <string>
2 #include <iostream>
3
4 class Weather
5 {
6     private:
7         double temperature;
8         double relative_humidity;
9         double dew_point;
10        std::string description;
11
12    void calculate_dp()
13    {
14        dew_point = temperature - relative_humidity;█
15    }
16
17 public:
18    Weather(double temperature, double relative_humidity, std::string
description)
19    {
20        this->temperature = temperature;
21        this->relative_humidity = relative_humidity;
22        this->description = description;
23    }
24
25    // Accessors
26    double get_temperature()
27    {
28        return this->temperature;
29    }
30
31    // mutator
32    void set_temperature(double temperature)
```

ben@computer ~ %

```
1 #include <string>
2 #include <iostream>
3
4 class Weather
5 {
6     private:
7         double temperature;
8         double relative_humidity;
9         double dew_point;
10        std::string description;
11
12    void calculate_dp()
13    {
14        this->dew_point = this->temperature - this->relative_humidity;
15    }
16
17 public:
18    Weather(double temperature, double relative_humidity, std::string
description)
19    {
20        this->temperature = temperature;
21        this->relative_humidity = relative_humidity;
22        this->description = description;
23    }
24
25    // Accessors
26    double get_temperature()
27    {
28        return this->temperature;
29    }
30
31    // mutator
32    void set_temperature(double temperature)
```

ben@computer ~ %

```
1 #include <string>
2 #include <iostream>
3
4 class Weather
5 {
6     private:
7         double temperature;
8         double relative_humidity;
9         double dew_point;
10        std::string description;
11
12    void calculate_dp()
13    {
14        this->dew_point = this->temperature - this->relative_humidity;
15    }
16
17 public:
18    Weather(double temperature, double relative_humidity, std::string
description)
19    {
20        this->temperature = temperature;
21        this->relative_humidity = relative_humidity;
22        this->description = description;
23
24        calculate_dp();  
    }
25
26
27 // Accessors
28 double get_temperature()
29 {
30     return this->temperature;
31 }
32
```

ben@computer ~ \$

```
9  double dew_point;
10 std::string description;
11
12 void calculate_dp()
13 {
14     this->dew_point = this->temperature - this->relative_humidity;
15 }
16
17 public:
18     Weather(double temperature, double relative_humidity, std::string
description)
19     {
20         this->temperature = temperature;
21         this->relative_humidity = relative_humidity;
22         this->description = description;
23
24         calculate_dp();
25     }
26
27 // Accessors
28 double get_temperature()
29 {
30     return this->temperature;
31 }
32
33 // mutator
34 void set_temperature(double temperature)
35 {
36     this->temperature = temperature;
37 }
38
39 void print()
40 {
```

ben@computer ~ %

```
25 }
26
27 // Accessors
28 double get_temperature()
29 {
30     return this->temperature;
31 }
32
33 // mutator
34 void set_temperature(double temperature)
35 {
36     this->temperature = temperature;
37 }
38
39 void print()
40 {
41     std::cout << "Temperature: " << this->temperature << std::endl
42 ;
43     std::cout << "RH: " << this->relative_humidity << std::endl;
44     std::cout << "DP: " << this->dew_point << std::endl;
45     std::cout << "Description: " << this->description << std::endl
46 ;
47 }
48
49 int main(void)
50 {
51     Weather now(20.0, 50.0, "sunny"); I
52     now.print();
53
54     return 0;
55 }
```

ben@computer ~ %

```
25 }
26
27 // Accessors
28 double get_temperature()
29 {
30     return this->temperature;
31 }
32
33 // mutator
34 void set_temperature(double temperature)
35 {
36     this->temperature = temperature;
37 }
38
39 void print()
40 {
41     std::cout << "Temperature: " << this->temperature << std::endl
42 ;
43     std::cout << "RH: " << this->relative_humidity << std::endl;
44     std::cout << "DP: " << this->dew_point << std::endl;
45     std::cout << "Description: " << this->description << std::endl
46 ;
47 }
48
49 int main(void)
50 {
51     Weather now(20.0, 50.0, "sunny");
52     now.print();
53
54     return 0;
55 }
```

```
ben@computer ~ % g++ weather.cpp
ben@computer ~ % g++ weather.cpp -o weather
ben@computer ~ % ./weather
Temperature: 20
RH: 50
DP: -30
Description: sunny
ben@computer ~ %
```