4.9 — Boolean values

ALEX APRIL 7, 2021

[illegible]

[illegible]

[illegible] Boolean [illegible]

Boolean variables

[illegible]

[illegible] bool [illegible]

```
bool b;
```

[illegible] false [illegible]

```
bool b1 { true };
bool b2 { false };
b1 = false;
bool b3 {}; // default initialize to false
```

[illegible]

```
bool b1 { !true }; // b1 will be initialized with the value false
bool b2 { !false }; // b2 will be initialized with the value true
```

[illegible]

ŋrps cŇmpnd_jqcŇĦSf cwqt_js_rcqmqf ceg rcecpqqẽĊd_jqcgmpẽĊrps cĦAca_sqcAmmjc_l qe ars_jjwqrmpceg rcecpqĤf cwe pcqml qgbcpcbe l g rcep_jqwncĦ

Printing Boolean variables

V f cl eu cenpg reAmmjc_l e_js cqeu grf eqrbIJJams rĤqrbIJJams renpg rqẽdmpe*/ jqc*Ĥe l bẽdmpe*ps c*IJ

```
#include <iostream>

int main()
{
    std::cout << true << '\n'; // true evaluates to 1
    std::cout << !true << '\n'; // !true evaluates to 0

    bool b{false};
    std::cout << b << '\n'; // b is false, which evaluates to 0
    std::cout << !b << '\n'; // !b is true, which evaluates to 1
    return 0;
}
```

Ns rns rqIJ

```
1
0
0
1
```

HdeIJwns eu_l reqrbIJJams remenpg renrps cŇmpnd_jqcŇeg qrc_benďĩ empẽĤwns e_l eş qqce*rbIJJ mmj_jnf_*ĦGcpcNe l ecv_k njcIJ

```
#include <iostream>

int main()
{
    std::cout << true << '\n';
    std::cout << false << '\n';

    std::cout << std::boolalpha; // print bools as true or false

    std::cout << true << '\n';
    std::cout << false << '\n';
    return 0;
}
```

Sf genpg rqIJ

```
1
0
true
false
```

Xns e_l eşqce*rbIJJ m`mmj_jnf_* emrs pl egre`_ai emdĦ

Integer to Boolean conversion

Xns e_l Ne g g jgce eAmmjc_l eu grf e l eg rcecps qg eeş l gdmpk eg g jgx_rgml IJ

```
#include <iostream>

int main()
{
 bool b{ 4 }; // error: narrowing conversions disallowed
 std::cout << b;

 return 0;
}
```

[illegible]

[illegible]

```
#include <iostream>

int main()
{
 std::cout << std::boolalpha; // print bools as true or false

 bool b1 = 4 ; // copy initialization allows implicit conversion from int to bool
 std::cout << b1 << '\n';

 bool b2 = 0 ; // copy initialization allows implicit conversion from int to bool
 std::cout << b2 << '\n';


 return 0;
}
```

[illegible]

```
true
false
```

Inputting Boolean values

[illegible]

[illegible]

```
#include <iostream>

int main()
{
 bool b{}; // default initialize to false
 std::cout << "Enter a boolean value: ";
 std::cin >> b;
 std::cout << "You entered: " << b << '\n';

 return 0;
}
```

```
Enter a Boolean value: true
You entered: 0
```

[illegible]

[illegible]

[illegible]

```cpp
#include <iostream>

int main()
{
 bool b{};
 std::cout << "Enter a boolean value: ";
 // Treat "false" and "true" as booleans
 std::cin >> std::boolalpha;
 std::cin >> b;
 std::cout << "You entered: " << b << '\n';

 return 0;
}
```

[illegible]

Boolean return values

[illegible]

[illegible]

```cpp
#include <iostream>

// returns true if x and y are equal, false otherwise
bool isEqual(int x, int y)
{
    return (x == y); // operator== returns true if x equals y, and false
otherwise
}

int main()
{
    std::cout << "Enter an integer: ";
    int x{};
    std::cin >> x;

    std::cout << "Enter another integer: ";
    int y{};
    std::cin >> y;

    std::cout << std::boolalpha; // print bools as true or false

    std::cout << x << " and " << y << " are equal? ";
    std::cout << isEqual(x, y); // will return true or false

    return 0;
}
```

[illegible]

```
Enter an integer: 5
Enter another integer: 5
5 and 5 are equal? true
```

```
Enter an integer: 6
Enter another integer: 4
6 and 4 are equal? false
```

[illegible]

[illegible]

[illegible]

Mcvrėjc qqml

Ĭfiĭ H rpmbs argml ęrnegdęr_rck cl rq

A_ai ęrnęr_ `jcęndęml rcl rq

Opct gms qęjc qqml

Ĭfi Ejm_rd eęnmd rę sk `cpq

Leave a comment... Put C++ code between triple-backticks (markdown style):```Your C++ cod

Name*

Email*

@t_r_pqęlmk ę frrnqĪĦep_t_r_pĦmk ħę pceml l carcbęrnęwns pęnmtgbcbęck _gę bbpcqĦ

Mmrgdwęk cę `ms rępcnjgcqĪęę ę POST COMMENT

589 COMMENTS

Newest