Cheatography

C++ Basics Cheat Sheet

by Dockson via cheatography.com/24842/cs/6284/

Iterator for loop

```
for(std::vector<int>::iterator it = v.begin(); it
!= v.end(); ++it) {
        it- >do Som eth ing();
 }
```

Lambda

```
[captures] (parameters) mutable -> return-type
{body}
[] = argument att fånga (behandlas som globala
variab ler).
() = normala funkti ons arg ument.
-> return -type = datatyp att returnera.
mutable = lägg till om du vill ändra på något.
Non-const function.
{} = function goes here.
Lambda går att skriva helt utan (), -> return -type
och mutable.
```

Felhantering

```
try {
        foo() //foo kastar ett error med
throw( "Fe ltext") någonstans i koden
}
catch( exc ept ion & e)
{
        std ::cout << e.what() << std::endl; //
blir "Feltext"
}
```

for_each loop med lambda

```
vector<int> v;
int sum{0};
for_ea ch( v.b egin(), v.end(),sum -> void {
       sum += x;
});
```

Läsa från fil med ifstream

```
ifstream file("/some/path")
string line;
vector <st rin g> file_c ontent;
while (getli ne( file, line)) {
        fil e_c ont ent.pu sh_ bac k(l ine);
}
```

Range-based for loop

```
vector<int> v;
for(int i : v) { // i = förväntad datatyp
    //d oSo mething
}
```

Hitta saker med egen jämförelse

```
find_if (InputIt first, InputIt last,
UnaryPredicate boolFunction) //returns iterator to
element if found, otherwise returns last
find_i f_not() finns också.
```

Kommandoradsargument med validering

```
int main(int argc, char* argv[ ]) { // plocka ut
argument.
        vec tor <st rin g> arg(argv, argv + argc);
       int num_of _it era tions;
        ist rin gstream arg(ar g.a t(1));
        if( !(arg >> num_of _it era tions)) //om
arg inte kan göras till int så kastas logic_ error
{
                throw logic_ err or( " Error
meddel and e");
       }
}
```

Huvudklass till subklass (Employee ← Boss)

```
Boss b = dynamic_cast<Boss>(employee);
if (b != nullptr) {
        std ::cout << b -> get_bonus
}
```

Konstruktorer/tilldelning

```
Ptr(Ptr const& rhs); //kopieringskostruktor
Ptr& operator = (Ptr const& rhs); //kopi eri ngs -
til lde lning
Ptr(Ptr && this) //flyt tko nst ruktor
Ptr& operator = (Ptr && rhs(?)) //flytt tillde -
lning
```


Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
https://apollopad.com

Slumpa tal

```
#include <random>
std::r and om_ device rnd;
std::u nif orm _in t_d ist rib uti on< int>
num(1, 100); // tal mellan 1-100
std::cout << num(rnd) << std::endl;
```

By **Dockson**
cheatography.com/dockson/

Published 6th December, 2015.
Last updated 12th May, 2016.

Sponsored by **ApolloPad.com**
Everyone has a novel in them. Finish Yours!
https://apollopad.com