# Inheritance

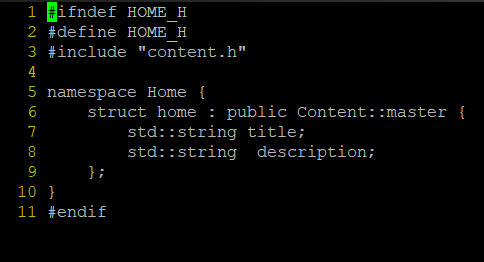
## Create Header File

Create header file for template which will inherit from master.h

For this example, create home:

cd data  
vi home.h

### home.h File

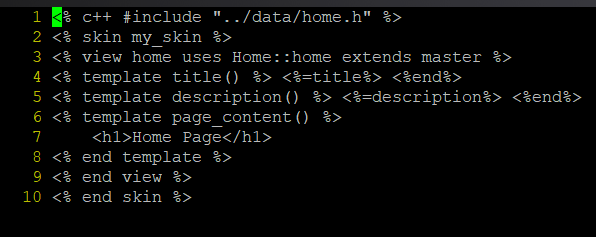


## Create Template File

For this example, create home:

cd templates  
vi home.tmpl

### home.tmpl File



## Building Templates

**IMPORTANT NOTE:** When building the templates, REMEMBER to include the parent template FIRST

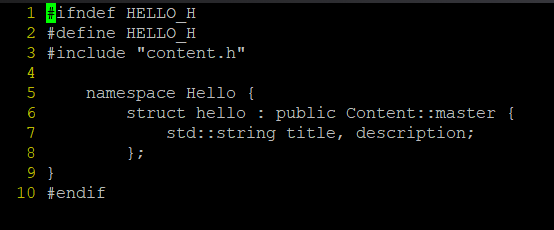
cd templates  
cppcms\_tmpl\_cc master.tmpl home.tmpl -o my\_skin.cpp

# Routing

To better visualize routing, lets create another template hello.h & hello.tmpl which will inherit from master.

## Hello

### hello.h File



### hello.tmpl File



### Build Templates

cd templates  
cppcms\_tmpl\_cc master.tmpl home.tmpl hello.tmpl -o my\_skin.cpp

## Configuring Routes

Inside main.cpp, include:

#include <cppcms/url\_dispatcher.h>  
#include <cppcms/url\_mapper.h>  
#include "data/home.h"  
#include "data/hello.h"

Apply following changes to main.cpp:

class Main : public cppcms::application {  
 public:  
 Main(cppcms::service &srv) : cppcms::application(srv) {  
 dispatcher().assign("(/?)",&Main::home,this);  
 mapper().assign("home", "/");  
  
 dispatcher().assign("/hello",&Main::hello,this);  
 mapper().assign("hello","/hello");  
  
 }  
 virtual void main(std::string url) {  
 cppcms::application::main(url);  
 }  
   
 void home() {  
 Home::home h;  
 h.title = "Home";  
 h.description = "Home Page";  
 render("home",h);  
 }  
  
 void hello () {  
 Hello::hello h;  
 h.title="Hello";  
 h.description = "Hello World Page";  
 render("hello", h);  
 }   
};

## Compile

export LD\_LIBRARY\_PATH=/usr/local/lib/  
g++ main.cpp templates/my\_skin.cpp -o main -lcppcms -lbooster

## Run

./main -c config.js