Ghorxata 1.0

Debugger output

* Ghorxata 1.0 - valid khorxata conversion
* At sumer controls computer user temperature
* Copyright (C) 2007 by Carles Oriol, carles@kumbaworld.com
* Ubuntu CatalanLoco Team
* Translated into english on may 6 2007 by Carles Oriol
#include <math.h>
#include <vertical_mixer.h>
#include <balanca.h>
#include <mesures.h>
#include <strainer_xines.h>
#include <bottle.h>
#include <permanent_coffe_filter.h>
#include <container.h>
#include <ingredients.h>
#include <funnel.h>
#include <chronomether.h>

#define QUANTITY 100
#define QUANTITY_water_mixer_1pass_capacity 300

void main () {
  // ghorxata object creation test;
  // Carles Oriol 21/06/2007
  const int QUANTITY_water = QUANTITY * 6;

  // Build needed objects
  vertical_mixer mixer = new vertical_mixer ( SIZE_mixer );
  strainer_xines strainer = new strainer_xines ();
  container_strainer = new container ( QUANTITY * 8 );

  // Alloc at container enough space to collect maximum ingredients to add
  // and we'll have space left.
  strainer.link( container );

  // Mixel moduel will destroy all the objects inserted in.
  // It's important to put chufas in water for 12h but it's optional.
  mixer << ( new ingredients_chufas ( QUANTITY ) );  // image 1 i 2
  mixer << ( new ingredients_sugar ( QUANTITY ) );  // image 3

  int QUANTITY_done = 0;
  // We'll repeat until we end the watter
  while ( QUANTITY_done < QUANTITY * 6 ) {
    int QUANTITY_to_do_this_pass = min( QUANTITY_water_mixer_1pass_capacity,
                                       (QUANTITY * 6) - QUANTITY_done );

    mixer << ( new ingredients_water ( QUANTITY_to_do_this_pass ) );  // image 4
    mixer.Speed( 3 );  // Speed'range from 1 to 3
    mixer.Run ();  // image 5
    chronomether.sleep ( 60 * 3 );  // in seconds
    mixer.Stop ();
    strainer << ( mixer.Resultat() );
    strainer.use();  // image 6

    // Stainer buffer will remain betwen passes
    // we move stainer left to the mixer and we'll mix with water again
    mixer << ( strainer.left() );
    QUANTITY_done += QUANTITY_to_do_this_pass;
  }

  // Ready objects for the last filter.
  // image 8
  big_funnel funnel = new big_funnel( );
Permanent_coffee_filter filter = new Permanent_coffee_filter();
usedsodabottle bottle = new usedsodabottle();

funnel.link( filter ); // image 9 - Link filter to funnel and the output to the bottle
funnel.link( bottle );

// Warning. In this process we have simplified this part assuming the bottle capacity
// it's enough for the total result.
// Can happen a non recoverable stack overflow if not controlled from the start
// and that would need a sophisticated cleaning kitchen process.
// must be revised before version 1.1

// We add the content of container to the funnel to fine filter and load the object bottle.
funnel << ( container.Resultat() ); // image 10 i image 11

// We have the object ghorxata ready at bottle.Content();
// You just have to fridge it for 24h before use

fridge myfridge = new fridge();
myfridge.add( bottle );

chronomether.sleep ( 60 * 60 * 24 );

myfridge.remove( bottle );

// and ... blaaam!!! We can use ghorxata now!!

// horxata = bottle.Content(); !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
// Object destroyers have a cleaning process

Translation notes: Horxata is a typical soft dring in the catalan countries done with chufa. It's a typical refresh on summer times. This recipe is correct and the results shown at top are real. You just have to read the code to do it.