

```

<?php

/**
 * @file
 *   An example module demonstrating the CTools multistep form wizard
 *
 * @see
 *   Full module available at:
 *   https://github.com/jgottwig/ctools-wizard-example
 */

define('OBJECT_ID', 1);           // Make it easier to identify object by defining
ID here

/**
 * Implements hook_menu()
 */
function multiform_menu() {
  $items['multiform'] = array(
    'title' => t('My Example Multipage Form'),
    'page callback' => 'multiform_form_builder',
    'access arguments' => array('view content'),
  );
  return $items;
}

/**
 * Our multipage form setup function
 */
function multiform_form_builder($step = NULL) {

  ctools_include('wizard');           // Used for multi-page forms
  ctools_include('object-cache');     // Used to handle data storage for multi-page
forms

  /**
   * The $form_info provides configuration options
   * for the ctools wizard
   */
  $form_info = array(
    'id' => 'multiform_form',
    'path' => 'multiform/%step',
    'show trail' => TRUE,
    'show back' => TRUE,
    'show cancel' => TRUE,
    'show return' => FALSE,
    'cancel callback' => '_multiform_builder_cancel',
    'finish callback' => '_multiform_builder_finish',
    'next callback' => '_multiform_builder_next',
    'next text' => t('Next'),
    'back text' => t('Previous'),
    'finish text' => t('Submit'),
    'order' => array(
      'step-1' => t('Step One'),
      'step-2' => t('Step Two'),
      'step-3' => t('Step Three'),
    ),
    'forms' => array(
      'step-1' => array(
        'form id' => 'step_1_form',

```

```

        'include' => drupal_get_path('module', 'multiform') .
'/multiform.step_1.inc',
    ),
    'step-2' => array(
        'form id' => 'step_2_form',
        'include' => drupal_get_path('module', 'multiform') .
'/multiform.step_2.inc',
    ),
    'step-3' => array(
        'form id' => 'step_3_form',
        'include' => drupal_get_path('module', 'multiform') .
'/multiform.step_3.inc',
    ),
    ),
);

if (empty($step)) {
    $step = 'step-1';
}

$object = _multiform_get_obj_cache(OBJECT_ID);

$form_state = array(
    'object_id' => OBJECT_ID,
    'object' => &$object,
);

return ctools_wizard_multistep_form($form_info, $step, $form_state);
}

/**
 * Retrieve object-cache or create
 */
function _multiform_get_obj_cache($id) {
    ctools_include('object-cache');
    if (!$cache = ctools_object_cache_get('multiform', $id)) {
        $cache = new stdClass();
        $cache->locked = ctools_object_cache_test('multiform', $id);
    }
    return $cache;
}

/**
 * Update the object
 */
function _multiform_update_obj_cache($id, $content) {
    ctools_include('object-cache');
    $cache = ctools_object_cache_set('multiform', $id, $content);
}

/**
 * Next button callback
 */
function _multiform_builder_next(&$form_state) {
    _multiform_update_obj_cache(OBJECT_ID, $form_state['object']);
}

/**
 * Finished button callback
 */
function _multiform_builder_finish() {

```

```
    _multiform_clear_obj_cache(OBJECT_ID);
}

/**
 * Cancel button callback
 */
function _multiform_builder_cancel(&$form_state) {
    // Note that we could do something with our $form_state if we wanted
    _multiform_clear_obj_cache(OBJECT_ID);
}

/**
 * Clear our object cache
 */
function _multiform_clear_obj_cache($id) {
    ctools_include('object-cache');
    ctools_object_cache_clear('multiform', $id);
    // Anything else you want to do here?
}
```