

DESKRIPSI DAN SPESIFIKASI CIPTAAN



Jenis Ciptaan : Program Komputer

Judul Ciptaan: E-Learning UNIDA Gontor

Pencipta

Rila Setyaningsih

Abdullah

Edy Prihantono

Hustinawaty

Pemegang Hak Cipta

Universitas Darussalam Gontor

Tanggal Diumumkan

8 Desember 2019

Diumumkan pertama kali di : Ponorogo

Deskripsi Dan Spesifikasi Ciptaan

E-learning merupakan pembelajaran jarak jauh (*distance learning*) yang memanfaatkan teknologi komputer, jaringan komputer dan/atau internet. *E-Learning* memungkinkan pembelajar untuk belajar melalui komputer di tempat mereka masing-masing tanpa harus secara fisik pergi mengikuti pelajaran/perkuliahan di kelas. *E-Learning* sering pula dipahami sebagai suatu bentuk pembelajaran berbasis web yang bisa diakses dari intranet di jaringan lokal atau internet. Untuk memaksimalkan proses pembelajaran, e-learning yang dikembangkan di UNIDA Gontor dilengkapi juga dengan fitur-fitur topik perkuliahan, forum diskusi, tugas dan juga kuis. Prosedur penggunaan e-learning ini adalah: (1) pengguna log-in ke halaman e-learning.unida.gontor.ac.id; (2) pengguna memilih kursus/mata kuliah yang diikuti; (3) pengguna (dosen) dapat mengunggah materi dengan memilih topik yang diinginkan, sedangkan untuk pengguna mahasiswa dapat melakukan unduh materi dengan klik topik yang diinginkan; (4) untuk bergabung dalam form diskusi pengguna dapat memilih halaman diskusi dengan menuliskan pertanyaan ataupun pokok bahasan yang ingin didiskusikan; (5) tugas dan kuis dapat diakses juga dengan klik item yang tersedia; (6) untuk mengakhiri aktivitas, pengguna dapat memilih menu *log-out*. Untuk dapat tergabung sebagai pengguna e-learning UNIDA Gontor, terlebih dahulu pengguna harus di *enroll* oleh admin. Melalui website e-learning tersebut, pengguna (dosen dan mahasiswa) dapat melakukan pembelajaran secara daring tanpa harus bertatap muka.

USER MANUAL

PENGGUNAAN PROGRAM

E-LEARNING UNIDA GONTOR



جامعة دار السلام كونتور
UNIVERSITY OF DARUSSALAM GONTOR

Home Link Kontak

You are not logged in

E-Learning UNIDA Gontor

Selamat Datang di Virtual Class
Universitas Darussalam Gontor

Masuk Kehalaman Login

Untuk dapat menggunakan E-Learning, anda harus login terlebih dahulu dengan akun yang sudah disediakan.

Login

Forgotten your username or password?



nsferring data from integra.unida.gontor.ac.id...



Ekonomi Politik Islam

Access

• Teacher: Rudi Candra



Pengantar Ilmu Hubungan Internasional

Access

• Teacher: Sofi Mubarak



Komunikasi Kontekstual

Mata Kuliah ini mengajarkan tentang komunikasi yang berkaitan dengan isu-isu kontekstual

Access

• Teacher: mrfai mrfai

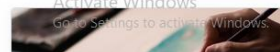


Teori Komunikasi

Mata Kuliah ini mengajarkan tentang teori-teori komunikasi

Access

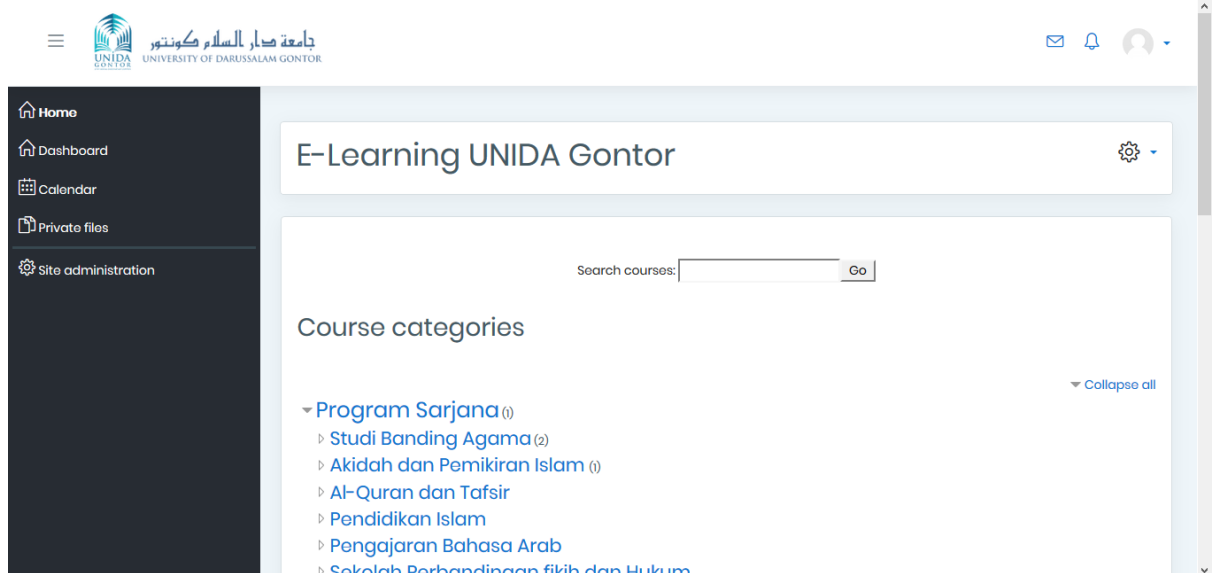
• Teacher: abdullah abdullah



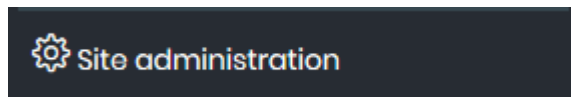
Universitas Darussalam Gontor
Dengan
Universitas Gunadarma

Cara Membuat Kategori, Sub Kategori & Course

1. Kita Login dulu, lalu sesudah login akan ada tampilan seperti ini.



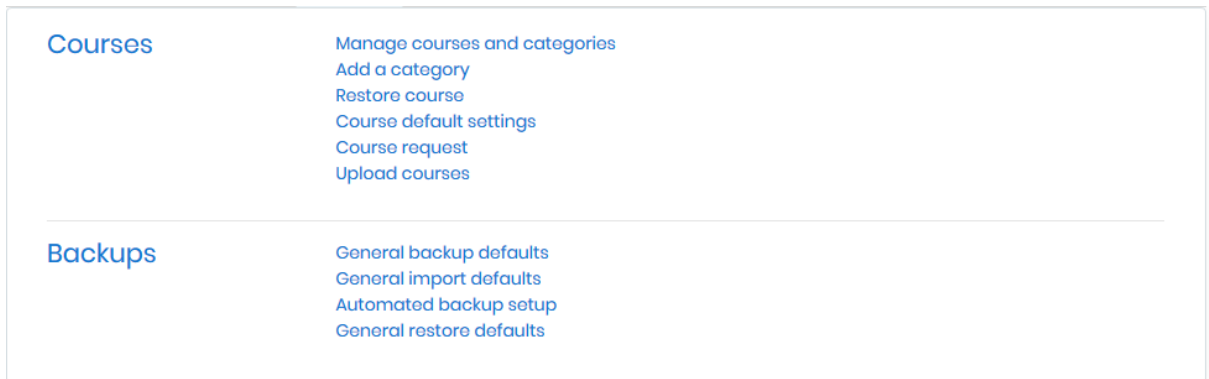
2. Lalu disebelah samping kiri kita pilih site administration.



3. Ini adalah tampilan menu Site Administration



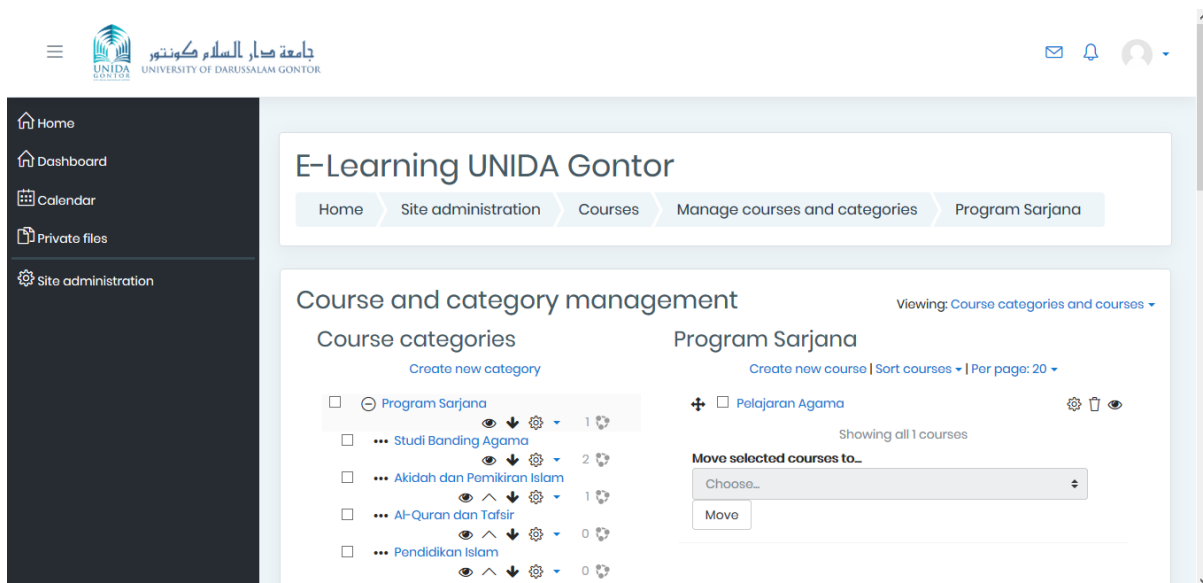
- Lalu kita pilih
- Ini adalah menu dalam Course



- Lalu pilih Manage Course and Categories

Manage courses and categories

- Ini adalah Tampilan Manage and Categories



- Selanjutnya untuk Membuat Kategori Klik Create New Category

Create new category

9. Ini adalah tampilan halaman create a new category.

E-Learning UNIDA Gontor

[Home](#) > [Category: Program Sarjana](#) > [Add a subcategory](#)

Add new category

Parent category

Category name ⓘ

Category ID number ?

Description

↓ **i** **B** **I** **≡** **≡** **🔗** **🔄** **🖼️** **📺** **📄**

10. Ubah Parent Category Menjadi Top Dan masukkan nama Kategori(Saya menamakan dengan Contoh Kategori.). Lalu Klik Create Category

Parent category: Top

Category name: Contoh Kategori

Category ID number:

Description:

Create category Cancel

11. Kategori “Contoh Kategori” Sudah dibuat.

E-Learning UNIDA Gontor

Home > Site administration > Courses > Manage courses and categories > Contoh Kategori

Course and category management

Viewing: Course categories and courses

Course categories

- Program Sarjana
- Program Magister
- Contoh Kategori

Contoh Kategori

No courses in this category

Move selected courses to...

Choose...

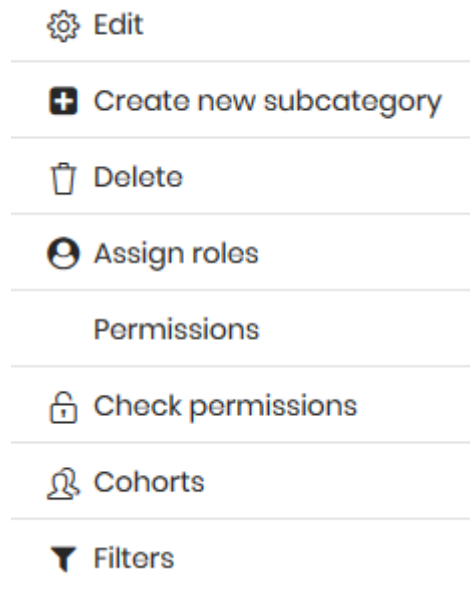
Move

12. Untuk Membuat Sub Kategori, Kita lihat bagian “Contoh Kategori”

13. Lalu, Pilih logo Setting



14. Lalu Akan Muncul Tampilan Sperti Gambar Dibawah Ini.



15. Lalu Pilih “Create New Subcategory”

+ Create new subcategory

16. Pilih Parent Category “Contoh Kategori” (Karena kita ingin membuat sub kategori dari kategori contoh kategori. Lalu kita namakan category nama menjadi “Contoh Subcategory”).

A form for creating a new subcategory with the following fields:

- Parent category: Dropdown menu with "Contoh Kategori" selected.
- Category name: Text input field containing "Contoh SubKategori".
- Category ID number: Text input field.
- Description: Rich text editor with a toolbar containing icons for undo, redo, italic, bold, list, link, unlink, image, video, and link.

At the bottom of the form are two buttons: "Create category" (highlighted in blue) and "Cancel".

17. Jadilah Subkategori yang bernama “Contoh Kategori”

Course and category management Viewing: [Course categories and courses](#) ▾

Course categories

[Create new category](#)

- ⊕ Program Sarjana 👁️ ⬇️ ⚙️ ▾ 1 🔄
- ⊕ Program Magister 1 👁️ ⬆️ ⚙️ ▾ 0 🔄
- ⊖ Contoh Kategori 👁️ ⬆️ ⚙️ ▾ 0 🔄
- ⋮ Contoh SubKategori 👁️ ⬆️ ⚙️ ▾ 0 🔄

Sorting

Selected categories ▾

Sort by Category name ascending ▾

Sort by Course full name ascending ▾

Contoh SubKategori

[Create new course](#) | [Sort courses](#) ▾ | [Per page: 20](#) ▾

No courses in this category

Move selected courses to...

Choose... ▾

Move

18. Untuk membuat Course, kita klik “Contoh Subkategori”, Lalu kita klik “Create A New Course”

Contoh SubKategori

[Create new course](#) | [Sort courses](#) ▾ | [Per page: 20](#) ▾

No courses in this category

Move selected courses to...

Choose... ▾

Move

19. Ini adalah tampilan dari Add a new Course

E-Learning UNIDA Gontor

Home > Site administration > Courses > Manage courses and categories > Add a new course

Add a new course

[Expand all](#)

▼ **General**

Course full name ⓘ ⓘ

Course short name ⓘ ⓘ

Course category ⓘ

Course visibility ⓘ

Course start date ⓘ

20. Lalu isikan Course Full Name dengan "Contoh Pembuatan Course" dan Course Short Name dengan "CPC" (nama bisa disesuaikan)

▼ **General**

Course full name ⓘ ⓘ

Course short name ⓘ ⓘ

Course category ⓘ

Course visibility ⓘ

Course start date ⓘ

Course end date ⓘ Enable

Course ID number ⓘ

21. Lalu Klik Save And Display

22. Jika sudah, tampilannya akan seperti ini.

Contoh Pembuatan Course

Home > Courses > Contoh Kategori > Contoh SubKategori > CPC > Participants

Participants

No filters applied

Search keyword or select filter

Enrol users

First name **All** A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Surname **All** A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Nothing to display

Select all Deselect all ? With selected users... Choose...

Proceed to course content Enrol users


23. Lalu pada Navigation Menu, Pilih CPC untuk masuk Course CPC / Contoh Pembuatan Course.

- CPC
- Participants
- Badges
- Competencies
- Grades

- Home
- Dashboard
- Calendar
- Private files

- Site administration

24. Ini adalah Tampilan dari Course "CPC"



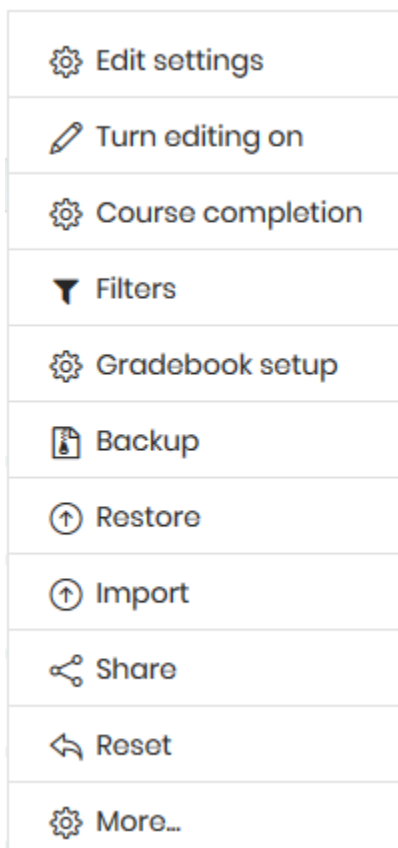
The screenshot shows a course page with the following elements:

- Title:** Contoh Pembuatan Course
- Breadcrumb Trail:** Home > Courses > Contoh Kategori > Contoh SubKategori > CPC
- Announcements:** A section with a megaphone icon and the text "Announcements".
- Topics:** Three horizontal bars representing topics, labeled "TOPIC 1", "TOPIC 2", and "TOPIC 3".
- Settings:** A gear icon in the top right corner.

25. Klik Logo Setting yang berada di pojok kanan atas.



26. Lalu kita Klik Turn Editing On



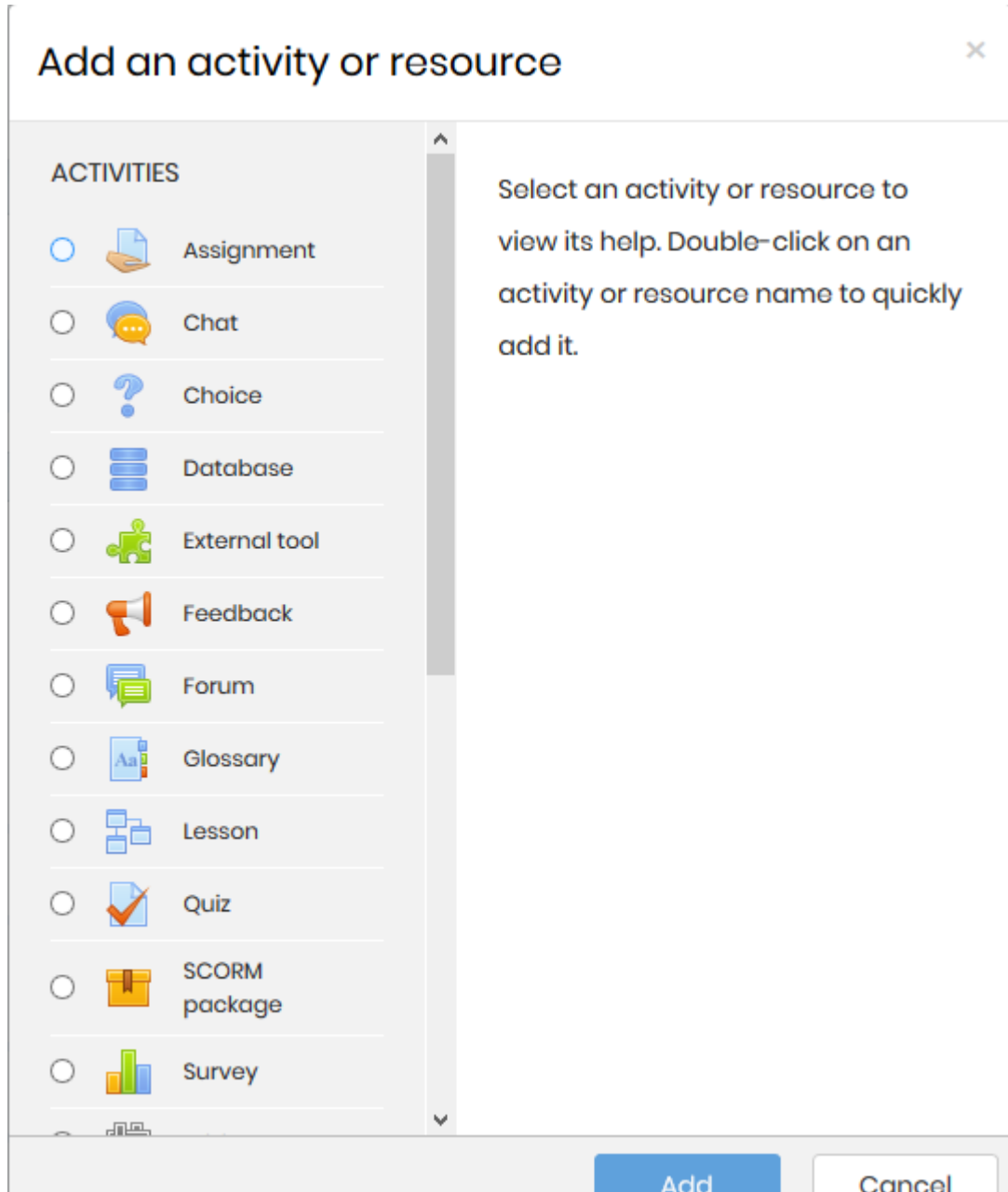
The dropdown menu contains the following options:

- Edit settings
- Turn editing on
- Course completion
- Filters
- Gradebook setup
- Backup
- Restore
- Import
- Share
- Reset
- More...

27. Ini adalah tampilan ketika turn editing on sudah diklik, Lalu kita klik Add an activity or resource untuk menambahkan soal.

The screenshot displays a course editing interface. At the top, the title "Contoh Pembuatan Course" is shown next to a gear icon. Below the title is a breadcrumb trail: Home > Courses > Contoh Kategori > Contoh SubKategori > CPC. The main content area is divided into sections. The first section is "Announcements", with a plus icon on the left and an "Edit" dropdown menu on the right. Below this is a section for "TOPIC 1", also with a plus icon on the left and an "Edit" dropdown menu on the right. Underneath "TOPIC 1" is a section for "TOPIC 2", with a plus icon on the left and an "Edit" dropdown menu on the right. At the bottom of the interface, there is a button labeled "Add an activity or resource".

28. Ini adalah tampilan pilihan jenis aktifitas yang akan kita buat.




29. Karena kita ingin membuat soal, kita pilih Quiz

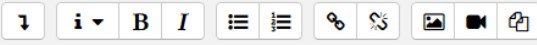



30. Kita masukan Namanya "Soal A" (Kata kata bisa disesuaikan)

▼ **General**

Name  Soal A

Description



Display description on course page 


31. Lalu Klik Save and display.

Save and display


32. Ini adalah tampilan soal yang sudah dibuat, untuk menambahkan pertanyaan, klik edit quiz.

Contoh Pembuatan Course

Home > Courses > Contoh Kategori > Contoh SubKategori > CPC > General > Soal A

Soal A 

Grading method: Highest grade

No questions have been added yet 

[Edit quiz](#)

[Back to the course](#)

33. Ini adalah tampilan Edit Quiz.

Contoh Pembuatan Course


Home > Courses > Contoh Kategori > Contoh SubKategori > CPC > General > Soal A > Edit quiz

Editing quiz: Soal A ?

Questions: 0 | This quiz is open

Maximum grade

Repaginate Total of marks: 0.00

 Shuffle ? [Add](#) ▼

34. Untuk menambahkan pertanyaan, klik Add lalu pilih a new question.

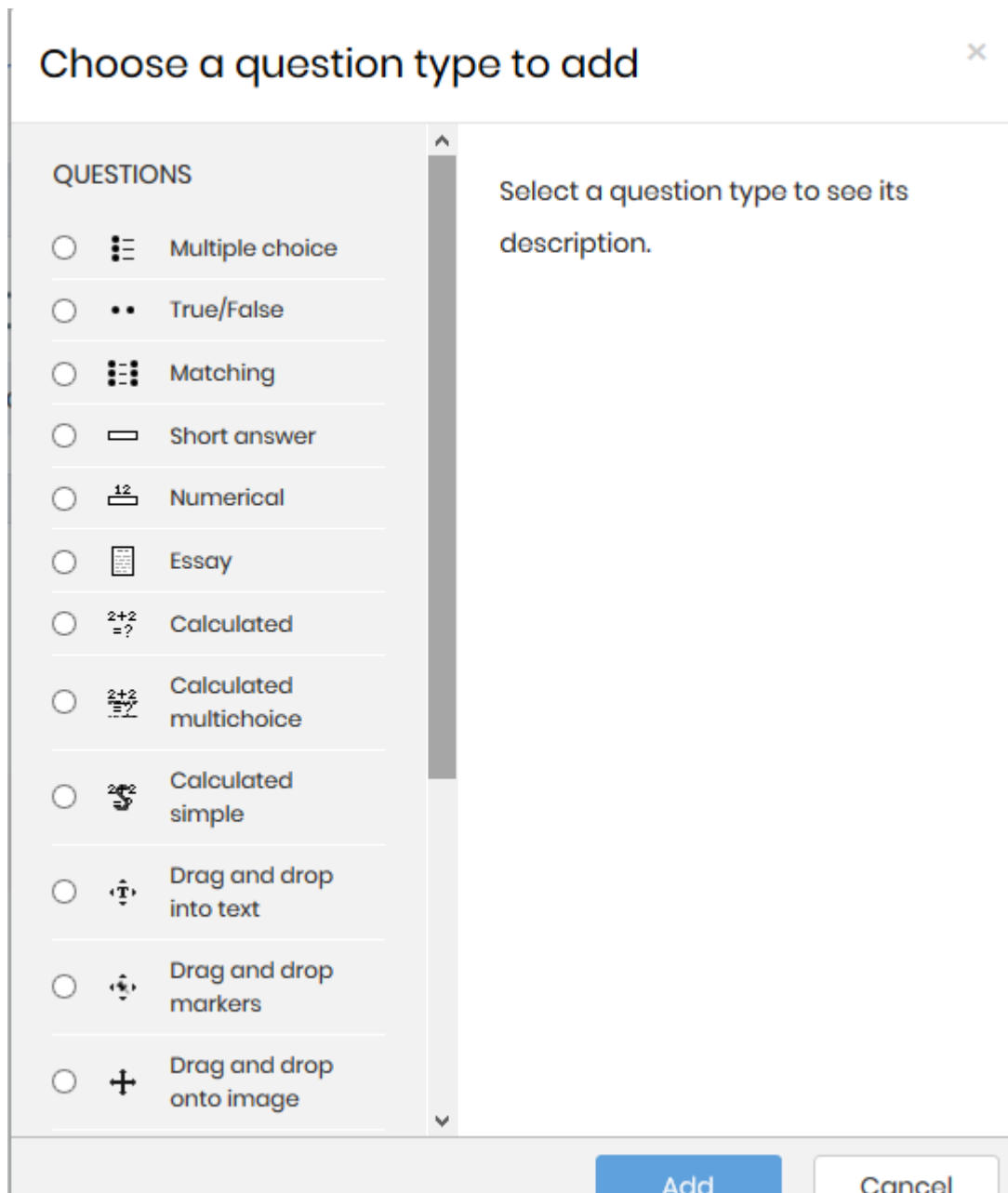
▼

a new question

from question bank

a random question

35. Ini adalah tampilan jenis pertanyaan yang akan kita pilih. Karena kita akan membuat soal pilihan ganda, kita klik Multiple Choice.



36. Ini adalah tampilan pembuatan pertanyaan multiple choice.

Contoh Pembuatan Course

Home Courses Contoh Kategori Contoh SubKategori CPC General Soal A

Question bank Questions Editing a Multiple choice question

Adding a Multiple choice question ?

[Expand all](#)

▼ **General**

Category

Question name ⓘ

Question text ⓘ

¶ **i** **B** **I** **☰** **☷** **🔗** **🔄** **🖼️** **📺** **📄**

37. Masukkan question name dengan nama "CPC" (nama bisa disesuaikan) dan Question text (isi dari pertanyaan tersebut)

Question name ⓘ

Question text ⓘ

¶ **i** **B** **I** **☰** **☷** **🔗** **🔄** **🖼️** **📺** **📄**

Alat untuk menentukan waktu adalah|

Default mark ⓘ

General feedback ?

¶ **i** **B** **I** **☰** **☷** **🔗** **🔄** **🖼️** **📺** **📄**

38. Lalu untuk pilihannya disesuaikan, dan pastikan disalah satu pilihan Gradenya bernilai 100%.

Choice 1

Jam

Grade: 100%

Feedback

Choice 2

Timbangan

Grade: None

Feedback

Choice 3

Penggaris

39. Lalu kita klik Save Change.

Save changes

40. Jadilah 1 buah pertanyaan yang berjudul CPC. Lalu kita klik save untuk diuji coba.

Editing quiz: Soal A

Questions: 1 | This quiz is open

Maximum grade: 10.00

Save

Repaginate

Select multiple items

Total of marks: 100



Page 1

Shuffle

Add

1



CPC Alat untuk menentukan waktu adalah



1.00

Add

41. Jika sudah disave kita klik Soal A

Soal A

42. Ini adalah tampilan jika kita akan masuk ke soal A .

Soal A ⚙️

Grading method: Highest grade

Summary of your previous attempts

Attempt	State	Marks / 100	Grade / 10.00	Review
Preview	Finished Submitted Monday, 5 March 2018, 5:37 AM	1.00	10.00	Review

Highest grade: 10.00 / 10.00.

[Re-attempt quiz](#)

43. Klik Attempt Quiz.

Re-attempt quiz

44. Ini adalah tampilan soal yang sudah kita buat, mari kita jawab, lalu klik finish attempt.

Contoh Pembuatan Course

Home > Courses > Contoh Kategori > Contoh SubKategori > CPC > General > Soal A > Preview

Question 1
Not yet answered
Marked out of 1.00
Flag question
Edit question

Alat untuk menentukan waktu adalah

Select one:

a. Penggaris

b. Timbangan

c. Jam

QUIZ NAVIGATION

1

[Finish attempt ...](#)

[Start a new preview](#)

[Finish attempt ...](#)

45. Jika sudah, klik Submit All and finish.

Soal A

Summary of attempt

Question	Status
1	Answer saved

[Return to attempt](#)

[Submit all and finish](#)

QUIZ NAVIGATION

1

[Finish attempt ...](#)

[Start a new preview](#)

46. Ini adalah tampilan ketika jawabannya benar.

The screenshot displays a quiz result for a question about time measurement. The question is "Alat untuk menentukan waktu adalah" (Instrument for measuring time is). The correct answer is "c. Jam" (Clock). The user's score is 1.00 out of 1.00, and the grade is 10.00 out of 10.00 (100%). The quiz navigation shows 1 question completed.

Started on Monday, 5 March 2018, 5:41 AM
State Finished
Completed on Monday, 5 March 2018, 5:42 AM
Time taken 1 min 54 secs
Marks 1.00/1.00
Grade 10.00 out of 10.00 (100%)

Question 1
Correct
Mark 1.00 out of 1.00
Flag question
Edit question

Alat untuk menentukan waktu adalah

Select one:

- a. Penggaris
- b. Timbangan
- c. Jam ✓

Your answer is correct.
The correct answer is: Jam

QUIZ NAVIGATION
1
Finish review
Start a new preview

47. Ini adalah tampilan ketika jawabannya salah.

The screenshot displays a quiz result for a question about time measurement. The user selected "b. Penggaris" (Ruler) as the answer, which is incorrect. The user's score is 0.00 out of 1.00, and the grade is 0.00 out of 10.00 (0%). The quiz navigation shows 1 question completed.

Started on Monday, 5 March 2018, 5:43 AM
State Finished
Completed on Monday, 5 March 2018, 5:43 AM
Time taken 9 secs
Marks 0.00/1.00
Grade 0.00 out of 10.00 (0%)

Question 1
Incorrect
Mark 0.00 out of 1.00
Flag question
Edit question

Alat untuk menentukan waktu adalah

Select one:

- a. Timbangan
- b. Penggaris ✗
- c. Jam

Your answer is incorrect.

QUIZ NAVIGATION
1
Finish review
Start a new preview

48. Ini adalah halaman laporannya, untuk menambah pertanyaan, klik logo setting yang terletak diujung kanan atas.

The screenshot shows a quiz report for "Soal A". The grading method is "Highest grade". The report includes a table of previous attempts and a "Re-attempt quiz" button.

Soal A ⚙️

Grading method: Highest grade

Summary of your previous attempts

Attempt	State	Marks / 100	Grade / 10.00	Review
Preview	Finished Submitted Monday, 5 March 2018, 5:43 AM	0.00	0.00	Review

Highest grade: 0.00 / 10.00.

[Re-attempt quiz](#)

49. Lalu klik Edit Quiz.

Edit quiz

50. Kita akan kembali kehalaman edit Quiz, Untuk menambah pertanyaan lagi, silahkan ikuti langkah 32 sampai 39.

Contoh Pembuatan Course

Home Courses Contoh Kategori Contoh SubKategori CPC General Soal A Edit quiz

Editing quiz: Soal A ?

Questions: 1 | This quiz is open

Maximum grade

Save

Repaginate

Select multiple items

Total of marks: 1.00



Shuffle ?

Page 1

Add ▼

1



CPC Alat untuk menentukan waktu adalah



1.00



Add ▼

SOURCE CODE E-LEARNING UNIDA GONTOR

Admin

059a1b7a2fdd2931c352331361b0189355a80680

```
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * This hack is intended for clustered sites that do not want
 * to use shared cacher for component cache.
 *
 * This file needs to be called after any change in PHP files in dataroot,
 * that is before upgrade and install.
 *
 * @package core
 * @copyright 2013 Petr Skoda (skodak) {@link http://skodak.org}
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', true);
define('ABORT_AFTER_CONFIG', true); // We need just the values from config.php.
define('CACHE_DISABLE_ALL', true); // This prevents reading of existing caches.
define('IGNORE_COMPONENT_CACHE', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php');

// Now get cli options.
list($options, $unrecognized) = cli_get_params(
    array(
        'file' => false,
        'rebuild' => false,
        'print' => false,
        'help' => false
    ),
    array(
```

```

        'h' => 'help'
    )
);

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized), 2);
}

if (!$options['rebuild'] and !$options['file'] and !$options['print']) {
    $help =
    "Create alternative component cache file

Options:
-h, --help      Print out this help
--rebuild      Rebuild \$CFG->alternative_component_cache file
--file=filepath Save component cache to file
--print        Print component cache file content

Example:
\$ php admin/cli/rebuild_alternative_component_cache.php --rebuild
";

    echo $help;
    exit(0);
}

error_reporting(E_ALL | E_STRICT);
ini_set('display_errors', 1);

$content = core_component::get_cache_content();

if ($options['print']) {
    echo $content;
    exit(0);
}

if ($options['rebuild']) {
    if (empty($CFG->alternative_component_cache)) {
        fwrite(STDERR, 'config.php does not contain $CFG->alternative_component_cache setting');
        fwrite(STDERR, "\n");
        exit(2);
    }
    $target = $CFG->alternative_component_cache;
} else {
    $target = $options['file'];
}

if (!$target) {
    fwrite(STDERR, "Invalid target file $target");
    fwrite(STDERR, "\n");
}

```



```

    exit(1);
}

$bytes = file_put_contents($target, $content);

if (!$bytes) {
    fwrite(STDERR, "Error writing to $target");
    fwrite(STDERR, "\n");
    exit(1);
}

// Success.
echo "File $target was updated\n";
exit(0);
<?php

// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Automated backups CLI cron
 *
 * This script executes
 *
 * @package core
 * @subpackage cli
 * @copyright 2010 Sam Hemelryk
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php'); // cli only functions
require_once($CFG->libdir.'/cronlib.php');

// now get cli options
list($options, $unrecognized) = cli_get_params(array('help'=>false),
        array('h'=>'help'));

```

```
if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknooption', 'admin', $unrecognized));
}
```

```
if ($options['help']) {
    $help =
    "Execute automated backups.
```

This script executes automated backups completely and is designed to be called via cron.

Options:

```
-h, --help      Print out this help
```

Example:

```
\$sudo -u www-data /usr/bin/php admin/cli/automated_backups.php
";
```

```
    echo $help;
    die;
}
if (CLI_MAINTENANCE) {
    echo "CLI maintenance mode active, backup execution suspended.\n";
    exit(1);
}
```

```
if (moodle_needs_upgrading()) {
    echo "Moodle upgrade pending, backup execution suspended.\n";
    exit(1);
}
```

```
require_once($CFG->libdir.'/adminlib.php');
require_once($CFG->libdir.'/gradelib.php');
```

```
if (!empty($CFG->showcronsql)) {
    $DB->set_debug(true);
}
if (!empty($CFG->showcrondebugging)) {
    set_debugging(DEBUG_DEVELOPER, true);
}
```

```
$starttime = microtime();
```

```
/// emulate normal session
cron_setup_user();
```

```
/// Start output log
$timenow = time();
```

```

mtrace("Server Time: ".date('r',$timenow)."\n\n");

// Run automated backups if required.
require_once($CFG->dirroot.'/backup/util/includes/backup_includes.php');
require_once($CFG->dirroot.'/backup/util/helper/backup_cron_helper.class.php');
backup_cron_automated_helper::run_automated_backup(backup_cron_automated_helper::RUN_I
MMEDIATELY);

mtrace("Automated cron backups completed correctly");

$difftime = microtime_diff($starttime, microtime());
mtrace("Execution took ".$difftime." seconds");
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * This script allows to do backup.
 *
 * @package core
 * @subpackage cli
 * @copyright 2013 Lancaster University
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', 1);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php');
require_once($CFG->dirroot.'/backup/util/includes/backup_includes.php');

// Now get cli options.
list($options, $unrecognized) = cli_get_params(array(
    'courseid' => false,
    'courseshortname' => "",
    'destination' => "",
    'help' => false,
), array('h' => 'help'));

```

```

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

if ($options['help'] || !($options['courseid'] || $options['courseshortname'])) {
    $help = <<<EOL
Perform backup of the given course.

Options:
--courseid=INTEGER      Course ID for backup.
--courseshortname=STRING Course shortname for backup.
--destination=STRING    Path where to store backup file. If not set the backup
                        will be stored within the course backup file area.
-h, --help              Print out this help.

Example:
\sudo -u www-data /usr/bin/php admin/cli/backup.php --courseid=2 --
destination=/moodle/backup/\n
EOL;

    echo $help;
    die;
}

$admin = get_admin();
if (!$admin) {
    mtrace("Error: No admin account was found");
    die;
}

// Do we need to store backup somewhere else?
$dir = rtrim($options['destination'], '/');
if (!empty($dir)) {
    if (!file_exists($dir) || !is_dir($dir) || !is_writable($dir)) {
        mtrace("Destination directory does not exists or not writable.");
        die;
    }
}

// Check that the course exists.
if ($options['courseid']) {
    $course = $DB->get_record('course', array('id' => $options['courseid']), '*', MUST_EXIST);
} else if ($options['courseshortname']) {
    $course = $DB->get_record('course', array('shortname' => $options['courseshortname']), '*',
MUST_EXIST);
}

cli_heading('Performing backup...');
$bc = new backup_controller(backup::TYPE_1COURSE, $course->id, backup::FORMAT_MOODLE,
backup::INTERACTIVE_YES, backup::MODE_GENERAL, $admin->id);

```

```

// Set the default filename.
$format = $bc->get_format();
$type = $bc->get_type();
$id = $bc->get_id();
$users = $bc->get_plan()->get_setting('users')->get_value();
$anonymised = $bc->get_plan()->get_setting('anonymize')->get_value();
$filename = backup_plan_dbops::get_default_backup_filename($format, $type, $id, $users,
$anonymised);
$bc->get_plan()->get_setting('filename')->set_value($filename);

// Execution.
$bc->finish_ui();
$bc->execute_plan();
$results = $bc->get_results();
$file = $results['backup_destination']; // May be empty if file already moved to target location.

// Do we need to store backup somewhere else?
if (!empty($dir)) {
    if ($file) {
        mtrace("Writing " . $dir.'/'.$filename);
        if ($file->copy_content_to($dir.'/'.$filename)) {
            $file->delete();
            mtrace("Backup completed.");
        } else {
            mtrace("Destination directory does not exist or is not writable. Leaving the backup in the
course backup file area.");
        }
    }
} else {
    mtrace("Backup completed, the new file is listed in the backup area of the given course");
}
$bc->destroy();
exit(0);
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Build and store theme CSS.

```

```

*
* @package core
* @subpackage cli
* @copyright 2017 Ryan Wyllie <ryan@moodle.com>
* @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
*/

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once("$CFG->libdir/clilib.php");
require_once("$CFG->libdir/csslib.php");
require_once("$CFG->libdir/outputlib.php");

$longparams = [
    'themes' => null,
    'direction' => null,
    'help' => false,
    'verbose' => false
];

$shortmappings = [
    't' => 'themes',
    'd' => 'direction',
    'h' => 'help',
    'v' => 'verbose'
];

// Get CLI params.
list($options, $unrecognized) = cli_get_params($longparams, $shortmappings);

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

if ($options['help']) {
    echo
    "Compile the CSS for one or more installed themes.
    Existing CSS caches will be replaced.
    By default all themes will be recompiled unless otherwise specified.

    Options:
    -t, --themes A comma separated list of themes to be compiled
    -d, --direction Only compile a single direction (either ltr or rtl)
    -v, --verbose Print info comments to stdout
    -h, --help Print out this help

    Example:
    \$ sudo -u www-data /usr/bin/php admin/cli/build_theme_css.php --themes=boost --direction=ltr
";
}

```

```

    die;
}

if (empty($options['verbose'])) {
    $trace = new null_progress_trace();
} else {
    $trace = new text_progress_trace();
}

cli_heading('Build theme css');

// Determine which themes we need to build.
$themenames = [];
if (is_null($options['themes'])) {
    $trace->output('No themes specified. Finding all installed themes.');
```

```

    $themenames = array_keys(core_component::get_plugin_list('theme'));
} else {
    if (is_string($options['themes'])) {
        $themenames = explode(',', $options['themes']);
    } else {
        cli_error('--themes must be a comma separated list of theme names');
```

```

    }
}

$trace->output('Checking that each theme is correctly installed...');
$themeconfigs = [];
foreach ($themenames as $themename) {
    if (is_null(theme_get_config_file_path($themename))) {
        cli_error("Unable to find theme config for {$themename}");
    }

    // Load the config for the theme.
    $themeconfigs[] = theme_config::load($themename);
}

$directions = ['ltr', 'rtl'];

if (!is_null($options['direction'])) {
    if (!in_array($options['direction'], $directions)) {
        cli_error("--direction must be either ltr or rtl");
    }

    $directions = [$options['direction']];
}

$trace->output('Building CSS for themes: ' . implode(' ', $themenames));
theme_build_css_for_themes($themeconfigs, $directions);

exit(0);
<?php
```

```
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * CLI script allowing to get and set config values.
 *
 * This is technically just a thin wrapper for {@link get_config()} and
 * {@link set_config()} functions.
 *
 * @package core
 * @subpackage cli
 * @copyright 2017 David Mudrak <david@moodle.com>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */
```

```
define('CLI_SCRIPT', true);
```

```
require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php');
```

\$usage = "Displays the current value of the given site setting. Allows to set it to the given value, too.

Usage:

```
# php cfg.php [--component=<componentname>] [--json] [--shell-arg]
# php cfg.php --name=<configname> [--component=<componentname>] [--shell-arg] [--no-eol]
# php cfg.php --name=<configname> [--component=<componentname>] --set=<value>
# php cfg.php --name=<configname> [--component=<componentname>] --unset
# php cfg.php [--help | -h]
```

Options:

```
-h --help          Print this help.
--component=<frankenstyle> Name of the component the variable is part of. Defaults to core.
--name=<configname>   Name of the configuration variable to get/set. If missing, print all
                    configuration variables of the given component.
--set=<value>        Set the given variable to this value.
--unset             Unset the given variable.
--shell-arg         Escape output values so that they can be directly used as shell script
arguments.
--json              Encode output list of values using JSON notation.
```


`--no-eol` Do not include the trailing new line character when printing the value.

The list of all variables of the given component can be printed as tab-separated list (default) or JSON object (`--json`). Particular values are printed as raw text values, optionally escaped so that they can be directly used as shell script arguments (`--shell-arg`). Single values are displayed with trailing new line by default, unless explicitly disabled (`--no-eol`).

In the read mode, the script exits with success status 0 if the requested value is found. If the requested variable is not set, the script exits with status 3. When listing all variables of the component, the exit status is always 0 even if no variables for the given component are found. When setting/unsetting a value, the exit status is 0. When attempting to set/unset a value that has already been hard-set in `config.php`, the script exits with error status 4. In case of unexpected error, the script exits with error status 1.

Examples:

```
# php cfg.php
Prints tab-separated list of all core configuration variables and their values.

# php cfg.php --json
Prints list of all core configuration variables and their values as a JSON object.

# php cfg.php --name=release
Prints the given configuration variable - e.g. \${CFG->release in this case.

# php cfg.php --component=tool_recyclebin
# Prints tab-separated list of the plugin's configuration variables.

# export DATAROOT=\$(php cfg.php --name=dataroot --shell-arg --no-eol)
Stores the given configuration variable in the shell variable, escaped
so that it can be safely used as a shell argument.

# php cfg.php --name=theme --set=clean
Sets the given configuration variable to the given value.

# php cfg.php --name=noemailver --unset
Unsets the previously configured variable.

";
```

```
list($options, $unrecognised) = cli_get_params([
    'help' => false,
    'component' => null,
    'name' => null,
    'set' => null,
    'unset' => false,
    'shell-arg' => false,
    'json' => false,
    'no-eol' => false,
], [
```

```

    'h' => 'help'
]);

if ($unrecognised) {
    $unrecognised = implode(PHP_EOL.' ', $unrecognised);
    cli_error(get_string('cliunknowoption', 'core_admin', $unrecognised));
}

if ($options['help']) {
    cli_writeln($usage);
    exit(2);
}

if ($options['unset'] || $options['set'] !== null) {
    // Unset the variable or set it to the given value.
    if (empty($options['name'])) {
        cli_error('Missing configuration variable name', 2);
    }

    // Check that the variable is not hard-set in the main config.php already.
    if (array_key_exists($options['name'], $CFG->config_php_settings)) {
        cli_error('The configuration variable is hard-set in the config.php, unable to change.', 4);
    }

    set_config($options['name'], $options['set'], $options['component']);
    exit(0);
}

if ($options['name'] === null) {
    // List all variables provided by the component (defaults to core).
    $got = get_config($options['component']);

    if ($options['json']) {
        cli_writeln(json_encode($got));
    } else {
        foreach ($got as $name => $value) {
            if ($options['shell-arg']) {
                $value = escapeshellarg($value);
            }
            cli_writeln($name."\\t".$value);
        }
    }

    exit(0);
} else {
    // Display the value of a single variable.

    $got = get_config($options['component'], $options['name']);

```

```

if ($got === false) {
    cli_error('No such configuration variable found.', 3);
}

if ($options['shell-arg']) {
    $got = escapeshellarg($got);
}

if ($options['json']) {
    $got = json_encode($got);
}

if ($options['no-eol']) {
    cli_write($got);
} else {
    cli_writeln($got);
}

exit(0);
}
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Validate that the current db structure matches the install.xml files.
 *
 * @package core
 * @copyright 2014 Totara Learning Solutions Ltd {@link http://www.totara.com/}
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 * @author Petr Skoda <petr.skoda@totara.com>
 */

define('CLI_SCRIPT', true);

require(__DIR__ . '/.././config.php');
require_once($CFG->libdir.'/clilib.php');

$help = "Validate database structure

```

Options:

-h, --help Print out this help.

Example:

```
\$ sudo -u www-data /usr/bin/php admin/cli/check_database_schema.php  
";
```

```
list($options, $unrecognized) = cli_get_params(  
    array(  
        'help' => false,  
    ),  
    array(  
        'h' => 'help',  
    )  
);  
  
if ($options['help']) {  
    echo $help;  
    exit(0);  
}  
  
if (empty($CFG->version)) {  
    echo "Database is not yet installed.\n";  
    exit(2);  
}  
  
$dbmanager = $DB->get_manager();  
$schema = $dbmanager->get_install_xml_schema();  
  
if (!$errors = $dbmanager->check_database_schema($schema)) {  
    echo "Database structure is ok.\n";  
    exit(0);  
}  
  
foreach ($errors as $table => $items) {  
    cli_separator();  
    echo "$table\n";  
    foreach ($items as $item) {  
        echo " * $item\n";  
    }  
}  
cli_separator();  
  
exit(1);  
<?php
```

```
// This file is part of Moodle - http://moodle.org/  
//  
// Moodle is free software: you can redistribute it and/or modify  
// it under the terms of the GNU General Public License as published by
```

```

// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * CLI cron
 *
 * This script looks through all the module directories for cron.php files
 * and runs them. These files can contain cleanup functions, email functions
 * or anything that needs to be run on a regular basis.
 *
 * @package core
 * @subpackage cli
 * @copyright 2009 Petr Skoda (http://skodak.org)
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php'); // cli only functions
require_once($CFG->libdir.'/cronlib.php');

// now get cli options
list($options, $unrecognized) = cli_get_params(array('help'=>false),
        array('h'=>'help'));

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

if ($options['help']) {
    $help =
"Execute periodic cron actions.

Options:
-h, --help      Print out this help

Example:
\$sudo -u www-data /usr/bin/php admin/cli/cron.php
";

    echo $help;

```

```

    die;
}

cron_run();
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * This script fixed incorrectly deleted users.
 *
 * @package   core
 * @subpackage cli
 * @copyright 2013 Marina Glancy
 * @license   http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php');

// Get cli options.
list($options, $unrecognized) = cli_get_params(
    array(
        'courses' => false,
        'fix'      => false,
        'help'     => false
    ),
    array(
        'h' => 'help',
        'c' => 'courses',
        'f' => 'fix'
    )
);

if ($options['help'] || empty($options['courses'])) {
    $help =
    "Checks and fixes that course modules and sections reference each other correctly.

```

Compares DB fields `course_sections.sequence` and `course_modules.section` checking that:

- `course_sections.sequence` contains each module id not more than once in the course
- for each moduleid from `course_sections.sequence` the field `course_modules.section` refers to the same section id (this means `course_sections.sequence` is more important if they are different)
- each module in the course is present in one of `course_sections.sequence`
- section sequences do not contain non-existing course modules

If there are any mismatches, the message is displayed. If `--fix` is specified, the records in DB are corrected.

This script may run for a long time on big systems if called for all courses.

Avoid executing the script when another user may simultaneously edit any of the courses being checked (recommended to run in mainenance mode).

Options:

- c, --courses List courses that need to be checked (comma-separated values or * for all). Required
- f, --fix Fix the mismatches in DB. If not specified check only and report problems to STDERR
- h, --help Print out this help

Example:

```
\$sudo -u www-data /usr/bin/php admin/cli/fix_course_sequence.php --courses=*
\$sudo -u www-data /usr/bin/php admin/cli/fix_course_sequence.php --courses=2,3,4 --fix
";
```

```
    echo $help;
    die;
}
```

```
$courseslist = preg_split('/\s*,\s*/', $options['courses'], -1, PREG_SPLIT_NO_EMPTY);
if (in_array('*', $courseslist)) {
    $where = "";
    $params = array();
} else {
    list($sql, $params) = $DB->get_in_or_equal($courseslist, SQL_PARAMS_NAMED, 'id');
    $where = 'WHERE id '. $sql;
}
$coursescount = $DB->get_field_sql('SELECT count(id) FROM {course} '. $where, $params);

if (!$coursescount) {
    cli_error('No courses found');
}
echo "Checking $coursescount courses...\n\n";

require_once($CFG->dirroot. '/course/lib.php');
```

```

$problems = array();
$courses = $DB->get_fieldset_sql('SELECT id FROM {course}'. $where, $params);
foreach ($courses as $courseid) {
    $errors = course_integrity_check($courseid, null, null, true, empty($options['fix']));
    if ($errors) {
        if (!empty($options['fix'])) {
            // Reset the course cache to make sure cache is recalculated next time the course is viewed.
            rebuild_course_cache($courseid, true);
        }
        foreach ($errors as $error) {
            cli_problem($error);
        }
        $problems[] = $courseid;
    } else {
        echo "Course [$courseid] is OK\n";
    }
}
if (!count($problems)) {
    echo "\n...All courses are OK\n";
} else {
    if (!empty($options['fix'])) {
        echo "\n...Found and fixed ".count($problems)." courses with problems. "\n";
    } else {
        echo "\n...Found ".count($problems)." courses with problems. To fix run:\n";
        echo "\$sudo -u www-data /usr/bin/php admin/cli/fix_course_sequence.php --courses=".join(', ',
$problems)." --fix". "\n";
    }
}
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * This script fixed incorrectly deleted users.
 *
 * @package core
 * @subpackage cli
 * @copyright 2013 Petr Skoda (http://skodak.org)
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later

```



```

*/

define('CLI_SCRIPT', true);

require(__DIR__.'/../config.php');
require_once($CFG->libdir.'/clilib.php');

// Now get cli options.
list($options, $unrecognized) = cli_get_params(array('help'=>false),
    array('h'=>'help'));

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

if ($options['help']) {
    $help =
        "Fix incorrectly deleted users.

        This script detects users that are marked as deleted instead
        of calling delete_user().

        Deleted users do not have original username, idnumber or email,
        we must also delete all roles, enrolments, group memberships, etc.

        Please note this script does not delete any public information
        such as forum posts.

        Options:
        -h, --help      Print out this help

        Example:
        \$sudo -u www-data /usr/bin/php admin/cli/fix_deleted_users.php
        ";

    echo $help;
    die;
}

cli_heading('Looking for sloppy user deletes');

// Look for sloppy deleted users where somebody only flipped the deleted flag.
$sql = "SELECT *
        FROM {user}
        WHERE deleted = 1 AND email LIKE '%@%' AND username NOT LIKE '%@%'";
$rs = $DB->get_recordset_sql($sql);
foreach ($rs as $user) {
    echo "Redeleting user $user->id: $user->username ($user->email)\n";
    delete_user($user);
}

```

```
}
```

```
cli_heading('Deleting all leftovers');
```

```
$DB->set_field('user', 'idnumber', '', array('deleted'=>1));
```

```
$DB->delete_records_select('role_assignments', "userid IN (SELECT id FROM {user} WHERE deleted = 1)");
```

```
$DB->delete_records_select('cohort_members', "userid IN (SELECT id FROM {user} WHERE deleted = 1)");
```

```
$DB->delete_records_select('groups_members', "userid IN (SELECT id FROM {user} WHERE deleted = 1)");
```

```
$DB->delete_records_select('user_enrolments', "userid IN (SELECT id FROM {user} WHERE deleted = 1)");
```

```
$DB->delete_records_select('user_preferences', "userid IN (SELECT id FROM {user} WHERE deleted = 1)");
```

```
$DB->delete_records_select('user_info_data', "userid IN (SELECT id FROM {user} WHERE deleted = 1)");
```

```
$DB->delete_records_select('user_lastaccess', "userid IN (SELECT id FROM {user} WHERE deleted = 1)");
```

```
$DB->delete_records_select('external_tokens', "userid IN (SELECT id FROM {user} WHERE deleted = 1)");
```

```
$DB->delete_records_select('external_services_users', "userid IN (SELECT id FROM {user} WHERE deleted = 1)");
```

```
exit(0);
```

```
<?php
```

```
// This file is part of Moodle - http://moodle.org/
```

```
//
```

```
// Moodle is free software: you can redistribute it and/or modify  
// it under the terms of the GNU General Public License as published by  
// the Free Software Foundation, either version 3 of the License, or  
// (at your option) any later version.
```

```
//
```

```
// Moodle is distributed in the hope that it will be useful,  
// but WITHOUT ANY WARRANTY; without even the implied warranty of  
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
// GNU General Public License for more details.
```

```
//
```

```
// You should have received a copy of the GNU General Public License  
// along with Moodle. If not, see http://www.gnu.org/licenses/.
```

```
/**
```

```
* This script fixes orphaned question categories.
```

```
*
```

```
* Orphaned question categories have had their associated context deleted  
* but the category itself remains in the database with an invalid context.
```

```
*
```

```
* @package core
```

```
* @subpackage cli
```

```
* @copyright 2013 Tyler Bannister (tyler.bannister@remote-learner.net)
```

```
* @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
*/
```

```
define('CLI_SCRIPT', true);
```

```
require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php');
require_once($CFG->libdir.'/questionlib.php');
```

```
$long = array('fix' => false, 'help' => false);
$short = array('f' => 'fix', 'h' => 'help');
```

```
// Now get cli options.
```

```
list($options, $unrecognized) = cli_get_params($long, $short);
```

```
if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}
```

```
if ($options['help']) {
```

```
    $help =
        "Fix orphaned question categories.
```

```
    This script detects question categories that have had their
    context deleted, thus severing them from their original purpose.
```

```
    This script will find the orphaned categories and delete the unused
    questions in each category found. Used questions will not be
    deleted, instead they will be moved to a rescue question category.
```

```
Options:
```

```
-h, --help      Print out this help
-f, --fix       Fix the orphaned question categories in the DB.
                If not specified only check and report problems to STDERR.
```

```
Example:
```

```
\$sudo -u www-data /usr/bin/php admin/cli/fix_orphaned_question_categories.php
\$sudo -u www-data /usr/bin/php admin/cli/fix_orphaned_question_categories.php -f
";
```

```
echo $help;
die;
}
```

```
cli_heading('Checking for orphaned categories');
```

```
$sql = 'SELECT qc.id, qc.contextid, qc.name
        FROM {question_categories} qc
        LEFT JOIN {context} c ON qc.contextid = c.id
        WHERE c.id IS NULL';
```

```

$categories = $DB->get_recordset_sql($sql);

$i = 0;
foreach ($categories as $category) {
    $i += 1;
    echo "Found orphaned category: {$category->name}\n";
    if (!empty($options['fix'])) {
        echo "Cleaning...";
        // One transaction per category.
        $transaction = $DB->start_delegated_transaction();
        question_category_delete_safe($category);
        $transaction->allow_commit();
        echo " Done!\n";
    }
}

if (($i > 0) && !empty($options['fix'])) {
    echo "Found and removed {$i} orphaned question categories\n";
} else if ($i > 0) {
    echo "Found {$i} orphaned question categories. To fix, run:\n";
    echo "\$sudo -u www-data /usr/bin/php admin/cli/fix_orphaned_question_categories.php --
fix\n";
} else {
    echo "No orphaned question categories found.\n";
}

$categories->close();
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * Enable or disable maintenance mode.
 *
 * @package core
 * @subpackage cli
 * @copyright 2009 Petr Skoda (http://skodak.org)
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later

```

```

*/

define('CLI_SCRIPT', true);

require(__DIR__.'/../config.php');
require_once("$CFG->libdir/clilib.php");
require_once("$CFG->libdir/adminlib.php");

// Now get cli options.
list($options, $unrecognized) = cli_get_params(array('enable'=>false, 'enablelater'=>0,
'enableold'=>false, 'disable'=>false, 'help'=>false),
        array('h'=>'help'));

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

if ($options['help']) {
    $help =
    "Maintenance mode settings.
    Current status displayed if not option specified.

    Options:
    --enable          Enable CLI maintenance mode
    --enablelater=MINUTES Number of minutes before entering CLI maintenance mode
    --enableold       Enable legacy half-maintenance mode
    --disable         Disable maintenance mode
    -h, --help       Print out this help

    Example:
    \$ sudo -u www-data /usr/bin/php admin/cli/maintenance.php
    "; //TODO: localize - to be translated later when everything is finished

    echo $help;
    die;
}

cli_heading(get_string('sitemaintenancemode', 'admin')." ($CFG->wwwroot)");

if ($options['enablelater']) {
    if (file_exists("$CFG->dataroot/climaintenance.html")) {
        // Already enabled, sorry.
        echo get_string('clistatusenabled', 'admin')." \n";
        return 1;
    }

    $time = time() + ($options['enablelater']*60);
    set_config('maintenance_later', $time);
}

```

```

echo get_string('clistatusenabledlater', 'admin', userdate($time))."\n";
return 0;

} else if ($options['enable']) {
    if (file_exists("$CFG->dataroot/climaintenance.html")) {
        // The maintenance is already enabled, nothing to do.
    } else {
        enable_cli_maintenance_mode();
    }
    set_config('maintenance_enabled', 0);
    unset_config('maintenance_later');
    echo get_string('sitemaintenanceoncli', 'admin')."\n";
    exit(0);

} else if ($options['enableold']) {
    set_config('maintenance_enabled', 1);
    unset_config('maintenance_later');
    echo get_string('sitemaintenanceon', 'admin')."\n";
    exit(0);

} else if ($options['disable']) {
    set_config('maintenance_enabled', 0);
    unset_config('maintenance_later');
    if (file_exists("$CFG->dataroot/climaintenance.html")) {
        unlink("$CFG->dataroot/climaintenance.html");
    }
    echo get_string('sitemaintenanceoff', 'admin')."\n";
    exit(0);
}

if (!empty($CFG->maintenance_enabled) or file_exists("$CFG->dataroot/climaintenance.html")) {
    echo get_string('clistatusenabled', 'admin')."\n";

} else if (isset($CFG->maintenance_later)) {
    echo get_string('clistatusenabledlater', 'admin', userdate($CFG->maintenance_later))."\n";

} else {
    echo get_string('clistatusdisabled', 'admin')."\n";
}
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.

```

```

//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * MySQL collation conversion tool.
 *
 * @package   core
 * @copyright 2012 Petr Skoda (http://skodak.org)
 * @license   http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php'); // cli only functions

if ($DB->get_dbfamily() !== 'mysql') {
    cli_error('This function is designed for MySQL databases only!');
}

// now get cli options
list($options, $unrecognized) = cli_get_params(array('help'=>false, 'list'=>false, 'collation'=>false,
'available'=>false),
    array('h'=>'help', 'l'=>'list', 'a'=>'available'));

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknooption', 'admin', $unrecognized));
}

$help =
    "MySQL collation conversions script.

```

It is strongly recommended to stop the web server before the conversion.
This script may be executed before the main upgrade - 1.9.x data for example.

Options:

```

--collation=COLLATION Convert MySQL tables to different collation
-l, --list          Show table and column information
-a, --available    Show list of available collations
-h, --help         Print out this help

```

Example:

```

\ $ sudo -u www-data /usr/bin/php admin/cli/mysql_collation.php --collation=utf8mb4_unicode_ci
";

```

```

if (!empty($options['collation'])) {
    $collations = mysql_get_collations();
    $collation = clean_param($options['collation'], PARAM_ALPHANUMEXT);
    $collation = strtolower($collation);
}

```

```

if (!isset($collations[$collation])) {
    cli_error("Error: collation '$collation' is not available on this server!");
}

$collationinfo = explode('_', $collation);
$charset = reset($collationinfo);

$engine = strtolower($DB->get_dbengine());

// Do checks for utf8mb4.
if (strpos($collation, 'utf8mb4') === 0) {
    // Do we have the right engine?
    if ($engine !== 'innodb' && $engine !== 'xtradb') {
        cli_error("Error: '$collation' requires InnoDB or XtraDB set as the engine.");
    }
    // Are we using Barracuda?
    if ($DB->get_row_format() != 'Barracuda') {
        // Try setting it here.
        try {
            $DB->execute("SET GLOBAL innodb_file_format=Barracuda");
        } catch (dml_exception $e) {
            cli_error("Error: '$collation' requires the file format to be set to Barracuda.
                An attempt was made to change the format, but it failed. Please try doing this
manually.");
        }
        echo "GLOBAL SETTING: innodb_file_format changed to Barracuda\n";
    }
    // Is one file per table being used?
    if (!$DB->is_file_per_table_enabled()) {
        try {
            $DB->execute("SET GLOBAL innodb_file_per_table=1");
        } catch (dml_exception $e) {
            cli_error("Error: '$collation' requires the setting 'innodb_file_per_table' be set to 'ON'.
                An attempt was made to change the format, but it failed. Please try doing this
manually.");
        }
        echo "GLOBAL SETTING: innodb_file_per_table changed to 1\n";
    }
    // Is large prefix set?
    if (!$DB->is_large_prefix_enabled()) {
        try {
            $DB->execute("SET GLOBAL innodb_large_prefix=1");
        } catch (dml_exception $e) {
            cli_error("Error: '$collation' requires the setting 'innodb_large_prefix' be set to 'ON'.
                An attempt was made to change the format, but it failed. Please try doing this
manually.");
        }
        echo "GLOBAL SETTING: innodb_large_prefix changed to 1\n";
    }
}
}

```



```

$sql = "SHOW VARIABLES LIKE 'collation_database'";
if (!$dbcollation = $DB->get_record_sql($sql)) {
    cli_error("Error: Could not access collation information on the database.");
}
$sql = "SHOW VARIABLES LIKE 'character_set_database'";
if (!$dbcharset = $DB->get_record_sql($sql)) {
    cli_error("Error: Could not access character set information on the database.");
}
if ($dbcollation->value !== $collation || $dbcharset->value !== $charset) {
    // Try to convert the DB.
    echo "Converting database to '$collation' for $CFG->wwwroot:\n";
    $sql = "ALTER DATABASE $CFG->dbname DEFAULT CHARACTER SET $charset DEFAULT COLLATE
= $collation";
    try {
        $DB->change_database_structure($sql);
    } catch (exception $e) {
        cli_error("Error: Tried to alter the database with no success. Please try manually changing the
database
        to the new collation and character set and then run this script again.");
    }
    echo "DATABASE CONVERTED\n";
}

echo "Converting tables and columns to '$collation' for $CFG->wwwroot:\n";
$prefix = $DB->get_prefix();
$prefix = str_replace('_', '\\_', $prefix);
$sql = "SHOW TABLE STATUS WHERE Name LIKE BINARY '$prefix%'";
$rs = $DB->get_recordset_sql($sql);
$converted = 0;
$skipped = 0;
$errors = 0;
foreach ($rs as $table) {
    echo str_pad($table->name, 40). " - ";

    if ($table->collation === $collation) {
        echo "NO CHANGE\n";
        $skipped++;
    } else {
        try {
            $DB->change_database_structure("ALTER TABLE $table->name CONVERT TO CHARACTER
SET $charset COLLATE $collation");
            echo "CONVERTED\n";
            $converted++;
        } catch (ddl_exception $e) {
            $result = mysql_set_row_format($table->name, $charset, $collation, $engine);
            if ($result) {
                echo "CONVERTED\n";
                $converted++;
            } else {
                // We don't know what the problem is. Stop the conversion.
            }
        }
    }
}

```

```
cli_error("Error: Tried to convert $table->name, but there was a problem. Please check
the details of this
```

```
    table and try again.");
    die();
}
}
}
```

```
$sql = "SHOW FULL COLUMNS FROM $table->name WHERE collation IS NOT NULL";
$rs2 = $DB->get_recordset_sql($sql);
foreach ($rs2 as $column) {
    $column = (object)array_change_key_case((array)$column, CASE_LOWER);
    echo ' '.str_pad($column->field, 36). " - ";
    if ($column->collation === $collation) {
        echo "NO CHANGE\n";
        $skipped++;
        continue;
    }
}
```

```
// Check for utf8mb4 collation.
$rowformat = $DB->get_row_format_sql($engine, $collation);
```

```
if ($column->type === 'tinytext' or $column->type === 'mediumtext' or $column->type ===
'text' or $column->type === 'longtext') {
    $notnull = ($column->>null === 'NO') ? 'NOT NULL' : 'NULL';
    $default = (!is_null($column->default) and $column->default !== "") ? "DEFAULT '$column-
>default'" : "";
```

```
// primary, unique and inc are not supported for texts
$sql = "ALTER TABLE $table->name
    MODIFY COLUMN $column->field $column->type
    CHARACTER SET $charset
    COLLATE $collation $notnull $default";
$DB->change_database_structure($sql);
```

```
} else if (strpos($column->type, 'varchar') === 0) {
    $notnull = ($column->>null === 'NO') ? 'NOT NULL' : 'NULL';
    $default = !is_null($column->default) ? "DEFAULT '$column->default'" : "";
```

```
if ($rowformat !== "") {
    $sql = "ALTER TABLE $table->name $rowformat";
    $DB->change_database_structure($sql);
}
```

```
$sql = "ALTER TABLE $table->name
    MODIFY COLUMN $column->field $column->type
    CHARACTER SET $charset
    COLLATE $collation $notnull $default";
$DB->change_database_structure($sql);
} else {
    echo "ERROR (unknown column type: $column->type)\n";
    $errors++;
}
```

```

        continue;
    }
    echo "CONVERTED\n";
    $converted++;
}
$rs2->close();
}
$rs->close();
echo "Converted: $converted, skipped: $skipped, errors: $errors\n";
exit(0); // success

} else if (!empty($options['list'])) {
    echo "List of tables for $CFG->wwwroot:\n";
    $prefix = $DB->get_prefix();
    $prefix = str_replace('_', '\\_', $prefix);
    $sql = "SHOW TABLE STATUS WHERE Name LIKE BINARY '$prefix%'";
    $rs = $DB->get_recordset_sql($sql);
    $counts = array();
    foreach ($rs as $table) {
        if (isset($counts[$table->collation])) {
            $counts[$table->collation]++;
        } else {
            $counts[$table->collation] = 1;
        }
        echo str_pad($table->name, 40);
        echo $table->collation. "\n";
        $collations = mysql_get_column_collations($table->name);
        foreach ($collations as $columnname=>$collation) {
            if (isset($counts[$collation])) {
                $counts[$collation]++;
            } else {
                $counts[$collation] = 1;
            }
            echo '  ';
            echo str_pad($columnname, 36);
            echo $collation. "\n";
        }
    }
    $rs->close();

    echo "\n";
    echo "Table collations summary for $CFG->wwwroot:\n";
    foreach ($counts as $collation => $count) {
        echo "$collation: $count\n";
    }
    exit(0); // success

} else if (!empty($options['available'])) {
    echo "List of available MySQL collations for $CFG->wwwroot:\n";
    $collations = mysql_get_collations();
    foreach ($collations as $collation) {

```

```

        echo " $collation\n";
    }
    die;

} else {
    echo $help;
    die;
}

// ===== Some functions =====

function mysql_get_collations() {
    global $DB;

    $collations = array();
    $sql = "SHOW COLLATION
        WHERE Collation LIKE 'utf8\_%' AND Charset = 'utf8'
        OR Collation LIKE 'utf8mb4\_%' AND Charset = 'utf8mb4'";
    $rs = $DB->get_recordset_sql($sql);
    foreach ($rs as $collation) {
        $collations[$collation->collation] = $collation->collation;
    }
    $rs->close();

    $collation = $DB->get_dbcollation();
    if (isset($collations[$collation])) {
        $collations[$collation] .= ' (default)';
    }

    return $collations;
}

function mysql_get_column_collations($tablename) {
    global $DB;

    $collations = array();
    $sql = "SELECT column_name, collation_name
        FROM INFORMATION_SCHEMA.COLUMNS
        WHERE table_schema = DATABASE() AND table_name = ? AND collation_name IS NOT NULL";
    $rs = $DB->get_recordset_sql($sql, array($tablename));
    foreach($rs as $record) {
        $collations[$record->column_name] = $record->collation_name;
    }
    $rs->close();
    return $collations;
}

function mysql_set_row_format($tablename, $charset, $collation, $engine) {
    global $DB;

```

```

$sql = "SELECT row_format
FROM INFORMATION_SCHEMA.TABLES
WHERE table_schema = DATABASE() AND table_name = ?";
$rs = $DB->get_record_sql($sql, array($tablename));
if ($rs) {
    if ($rs->row_format == 'Compact' || $rs->row_format == 'Redundant') {
        $rowformat = $DB->get_row_format_sql($engine, $collation);
        // Try to convert to compressed format and then try updating the collation again.
        $DB->change_database_structure("ALTER TABLE $tablename $rowformat");
        $DB->change_database_structure("ALTER TABLE $tablename CONVERT TO CHARACTER SET
$charset COLLATE $collation");
    } else {
        // Row format may not be the problem. Can not diagnose problem. Send fail reply.
        return false;
    }
} else {
    return false;
}
return true;
}
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * MySQL table row compression tool tool.
 *
 * @package core
 * @copyright 2014 Totara Learning Solutions Ltd {@link http://www.totarams.com/}
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 * @author Petr Skoda <petr.skoda@totarams.com>
 */

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir . '/clilib.php');

```

```

if ($DB->get_dbfamily() !== 'mysql') {
    cli_error('This script is used for MySQL databases only.');
```

```

}

$engine = strtolower($DB->get_dbengine());
if ($engine !== 'innodb' and $engine !== 'xtradb') {
    cli_error('This script is for MySQL servers using InnoDB or XtraDB engines only.');
```

```

}

list($options, $unrecognized) = cli_get_params(
    array('help' => false, 'info' => false, 'list' => false, 'fix' => false, 'showsql' => false),
    array('h' => 'help', 'i' => 'info', 'l' => 'list', 'f' => 'fix', 's' => 'showsql')
);

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

```

```

$help =
    "Script for detection of row size problems in MySQL InnoDB tables.

```

By default InnoDB storage table is using legacy Antelope file format which has major restriction on database row size.

Use this script to detect and fix database tables with potential data overflow problems.

Options:

```

-i, --info      Show database information
-l, --list      List problematic tables
-f, --fix       Attempt to fix all tables (requires SUPER privilege)
-s, --showsql   Print SQL statements for fixing of tables
-h, --help     Print out this help

```

Example:

```

\ $ sudo -u www-data /usr/bin/php admin/cli/mysql_compressed_rows.php -l
";

```

```

/** @var mysql_sql_generator $generator */
$generator = $DB->get_manager()->generator;

```

```

$info = $DB->get_server_info();
$filepertable = $DB->get_record_sql("SHOW VARIABLES LIKE 'innodb_file_per_table'");
$filepertable = $filepertable ? $filepertable->value : "";
$fileformat = $DB->get_record_sql("SHOW VARIABLES LIKE 'innodb_file_format'");
$fileformat = $fileformat ? $fileformat->value : "";
$prefix = $DB->get_prefix();
$database = $CFG->dbname;

```

```

if (!empty($options['info'])) {
    echo "Database version:    ". $info['description'] . "\n";
}

```

```

echo "Database name:    $database\n";
echo "Database engine:  " . $DB->get_dbengine() . "\n";
echo "innodb_file_per_table: $filepertable\n";
echo "innodb_file_format: $fileformat\n";

exit(0);

} else if (!empty($options['list'])) {
    $problem = false;
    foreach ($DB->get_tables(false) as $table) {
        $columns = $DB->get_columns($table, false);
        $size = $generator->guess_antelope_row_size($columns);
        $format = $DB->get_row_format($table);
        if ($size <= $generator::ANTELOPE_MAX_ROW_SIZE) {
            continue;
        }

        echo str_pad($prefix . $table, 32, ' ', STR_PAD_RIGHT);
        echo str_pad($format, 11, ' ', STR_PAD_RIGHT);

        if ($format === 'Compact' or $format === 'Redundant') {
            $problem = true;
            echo " (needs fixing)\n";
        } else if ($format !== 'Compressed' and $format !== 'Dynamic') {
            echo " (unknown)\n";
        } else {
            echo "\n";
        }
    }

    if ($problem) {
        exit(1);
    }
    exit(0);

} else if (!empty($options['fix'])) {
    $fixtables = array();
    foreach ($DB->get_tables(false) as $table) {
        $columns = $DB->get_columns($table, false);
        $size = $generator->guess_antelope_row_size($columns);
        $format = $DB->get_row_format($table);
        if ($size <= $generator::ANTELOPE_MAX_ROW_SIZE) {
            continue;
        }
        if ($format === 'Compact' or $format === 'Redundant') {
            $fixtables[$table] = $table;
        }
    }
}

```

```

if (!$fixtables) {
    echo "No changes necessary\n";
    exit(0);
}

if ($filepertable != 'ON') {
    try {
        $DB->execute("SET GLOBAL innodb_file_per_table=1");
    } catch (dml_exception $e) {
        echo "Cannot enable GLOBAL innodb_file_per_table setting, use --showsql option and
execute the statements manually.";
        throw $e;
    }
}
if ($fileformat != 'Barracuda') {
    try {
        $DB->execute("SET GLOBAL innodb_file_format=Barracuda");
    } catch (dml_exception $e) {
        echo "Cannot change GLOBAL innodb_file_format setting, use --showsql option and execute
the statements manually.";
        throw $e;
    }
}

if (!$DB->is_compressed_row_format_supported(false)) {
    echo "MySQL server is not compatible with compressed row format.";
    exit(1);
}

foreach ($fixtables as $table) {
    $DB->change_database_structure("ALTER TABLE {$prefix}$table ROW_FORMAT=Compressed");
    echo str_pad($prefix . $table, 32, ' ', STR_PAD_RIGHT) . " ... Compressed\n";
}

exit(0);

} else if (!empty($options['showsql'])) {
    $fixtables = array();

    foreach ($DB->get_tables(false) as $table) {
        $columns = $DB->get_columns($table, false);
        $size = $generator->guess_antelope_row_size($columns);
        $format = $DB->get_row_format($table);
        if ($size <= $generator::ANTELOPE_MAX_ROW_SIZE) {
            continue;
        }
        if ($format === 'Compact' or $format === 'Redundant') {
            $fixtables[$table] = $table;
        }
    }
}
if (!$fixtables) {

```



```

        echo "No changes necessary\n";
        exit(0);
    }

    echo "Copy the following SQL statements and execute them using account with SUPER
privilege:\n\n";
    echo "USE $database;\n";
    echo "SET SESSION sql_mode=STRICT_ALL_TABLES;\n";
    echo "SET GLOBAL innodb_file_per_table=1;\n";
    echo "SET GLOBAL innodb_file_format=Barracuda;\n";
    foreach ($fixtables as $table) {
        echo "ALTER TABLE {$prefix}$table ROW_FORMAT=Compressed;\n";
    }
    echo "\n";
    exit(0);

} else {
    echo $help;
    die;
}

<?php

// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * MySQL engine conversion tool.
 *
 * @package   core
 * @subpackage cli
 * @copyright 2009 Petr Skoda (http://skodak.org)
 * @license  http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php'); // cli only functions

```

```

if ($DB->get_dbfamily() !== 'mysql') {
    cli_error('This function is designed for MySQL databases only!');
}

// now get cli options
list($options, $unrecognized) = cli_get_params(array('help'=>false, 'list'=>false, 'engine'=>false,
'available'=>false),
        array('h'=>'help', 'l'=>'list', 'a'=>'available'));

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

```

```

$help =
"MySQL engine conversions script.

```

It is recommended to stop the web server before the conversion.
Do not use MyISAM if possible, because it is not ACID compliant and does not support transactions.

Options:

```

--engine=ENGINE    Convert MySQL tables to different engine
-l, --list          Show table information
-a, --available    Show list of available engines
-h, --help         Print out this help

```

Example:

```

\sudo -u www-data /usr/bin/php admin/cli/mysql_engine.php --engine=InnoDB
";

```

```

if (!empty($options['engine'])) {
    $engines = mysql_get_engines();
    $engine = clean_param($options['engine'], PARAM_ALPHA);
    if (!isset($engines[strtoupper($engine)])) {
        cli_error("Error: engine '$engine' is not available on this server!");
    }

    echo "Converting tables to '$engine' for $CFG->wwwroot:\n";
    $prefix = $DB->get_prefix();
    $prefix = str_replace('_', '\\_', $prefix);
    $sql = "SHOW TABLE STATUS WHERE Name LIKE BINARY '$prefix%'";
    $rs = $DB->get_recordset_sql($sql);
    $converted = 0;
    $skipped = 0;
    $errors = 0;
    foreach ($rs as $table) {
        if (strtoupper($table->engine) === strtoupper($engine)) {
            $newengine = mysql_get_table_engine($table->name);
            echo str_pad($table->name, 40). " - NO CONVERSION NEEDED ($newengine)\n";

```

```

        $skipped++;
        continue;
    }
    echo str_pad($table->name, 40). " - ";

    try {
        $DB->change_database_structure("ALTER TABLE {$table->name} ENGINE = $engine");
        $newengine = mysql_get_table_engine($table->name);
        if (strtoupper($newengine) !== strtoupper($engine)) {
            echo "ERROR ($newengine)\n";
            $errors++;
            continue;
        }
        echo "DONE ($newengine)\n";
        $converted++;
    } catch (moodle_exception $e) {
        echo $e->getMessage()."\n";
        $errors++;
        continue;
    }
}
$rs->close();
echo "Converted: $converted, skipped: $skipped, errors: $errors\n";
exit(0); // success

} else if (!empty($options['list'])) {
    echo "List of tables for $CFG->wwwroot:\n";
    $prefix = $DB->get_prefix();
    $prefix = str_replace('_', '\\_', $prefix);
    $sql = "SHOW TABLE STATUS WHERE Name LIKE BINARY '$prefix%'";
    $rs = $DB->get_recordset_sql($sql);
    $counts = array();
    foreach ($rs as $table) {
        if (isset($counts[$table->engine])) {
            $counts[$table->engine]++;
        } else {
            $counts[$table->engine] = 1;
        }
        echo str_pad($table->engine, 10);
        echo $table->name . "\n";
    }
    $rs->close();

    echo "\n";
    echo "Table engines summary for $CFG->wwwroot:\n";
    foreach ($counts as $engine => $count) {
        echo "$engine: $count\n";
    }
    exit(0); // success

} else if (!empty($options['available'])) {

```

```

    echo "List of available MySQL engines for $CFG->wwwroot:\n";
    $engines = mysql_get_engines();
    foreach ($engines as $engine) {
        echo " $engine\n";
    }
    die;

} else {
    echo $help;
    die;
}

// ===== Some functions =====

function mysql_get_engines() {
    global $DB;

    $sql = "SHOW Engines";
    $rs = $DB->get_recordset_sql($sql);
    $engines = array();
    foreach ($rs as $engine) {
        if (strtoupper($engine->support) !== 'YES' and strtoupper($engine->support) !== 'DEFAULT') {
            continue;
        }
        $engines[strtoupper($engine->engine)] = $engine->engine;
        if (strtoupper($engine->support) === 'DEFAULT') {
            $engines[strtoupper($engine->engine)] .= ' (default)';
        }
    }
    $rs->close();

    return $engines;
}

function mysql_get_table_engine($tablename) {
    global $DB;

    $engine = null;
    $sql = "SHOW TABLE STATUS WHERE Name = '$tablename'"; // no special chars expected here
    $rs = $DB->get_recordset_sql($sql);
    if ($rs->valid()) {
        $record = $rs->current();
        $engine = $record->engine;
    }
    $rs->close();
    return $engine;
}
<?php
// This file is part of Moodle - http://moodle.org/

```

```

//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * CLI script to purge caches without asking for confirmation.
 *
 * @package core
 * @subpackage cli
 * @copyright 2011 David Mudrak <david@moodle.com>
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php');

list($options, $unrecognized) = cli_get_params(array('help' => false), array('h' => 'help'));

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknooption', 'admin', $unrecognized), 2);
}

if ($options['help']) {
    $help =
    "Invalidates all Moodle internal caches

Options:
-h, --help      Print out this help

Example:
\$sudo -u www-data /usr/bin/php admin/cli/purge_caches.php
";

    echo $help;
    exit(0);
}

purge_all_caches();

```

```

exit(0);

<?php

// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * This script allows you to reset any local user password.
 *
 * @package core
 * @subpackage cli
 * @copyright 2009 Petr Skoda (http://skodak.org)
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php'); // cli only functions

// Define the input options.
$longparams = array(
    'help' => false,
    'username' => "",
    'password' => "",
    'ignore-password-policy' => false
);

$shortparams = array(
    'h' => 'help',
    'u' => 'username',
    'p' => 'password',
    'i' => 'ignore-password-policy'
);

// now get cli options
list($options, $unrecognized) = cli_get_params($longparams, $shortparams);

```

```

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

```

```

if ($options['help']) {
    $help =
"Reset local user passwords, useful especially for admin accounts.

```

There are no security checks here because anybody who is able to execute this file may execute any PHP too.

Options:

```

-h, --help          Print out this help
-u, --username=username  Specify username to change
-p, --password=newpassword  Specify new password
--ignore-password-policy  Ignore password policy when setting password

```

Example:

```

\sudo -u www-data /usr/bin/php admin/cli/reset_password.php
\sudo -u www-data /usr/bin/php admin/cli/reset_password.php --username=rosaura --
password=jiu3jiu --ignore-password-policy
";

```

```

    echo $help;
    die;
}
if ($options['username'] == "" ) {
    cli_heading('Password reset');
    $prompt = "Enter username (manual authentication only)";
    $username = cli_input($prompt);
} else {
    $username = $options['username'];
}

```

```

if (!$user = $DB->get_record('user', array('auth'=>'manual', 'username'=>$username,
'mnethostid'=>$CFG->mnet_localhost_id))) {
    cli_error("Can not find user '$username'");
}

```

```

if ($options['password'] == "" ) {
    $prompt = "Enter new password";
    $password = cli_input($prompt);
} else {
    $password = $options['password'];
}

```

```

$errmsg = ""; //prevent eclipse warning
if (!$options['ignore-password-policy'] ) {
    if (!check_password_policy($password, $errmsg)) {

```

```

        cli_error(html_to_text($errmsg, 0));
    }
}

$hashedpassword = hash_internal_user_password($password);

$DB->set_field('user', 'password', $hashedpassword, array('id'=>$user->id));

echo "Password changed\n";

exit(0); // 0 means success.
<?php
// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * This script implements some useful svg manipulation tricks.
 *
 * @package core_admin
 * @subpackage cli
 * @copyright 2012 Petr Skoda {@link http://skodak.org}
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

define('CLI_SCRIPT', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/clilib.php');

// Now get cli options.
list($options, $unrecognized) = cli_get_params(array('help'=>false, 'ie9fix'=>false,
'noaspectratio'=>false, 'path'=>$CFG->dirroot),
    array('h'=>'help'));

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

```



```

// If necessary add files that should be ignored - such as in 3rd party plugins.
$blacklist = array();
$path = $options['path'];
if (!file_exists($path)) {
    cli_error("Invalid path $path");
}

if ($options['ie9fix']) {
    core_admin_recurse_svgs($path, "", 'core_admin_svgtool_ie9fix', $blacklist);
} else if ($options['noaspectratio']) {
    core_admin_recurse_svgs($path, "", 'core_admin_svgtool_noaspectratio', $blacklist);
} else {
    $help =
        "Some svg image tweaks for icon designers.

```

Options:

```

-h, --help      Print out this help
--ie9fix        Adds preserveAspectRatio="xMinYMid meet" to every svg image
--noaspectratio Removes preserveAspectRatio from svg files
--path=PATH     Path to directory or file to be converted, by default \${CFG->dirroot}

```

Examples:

```

\${php} svgtool.php --ie9fix
\${php} svgtool.php --ie9fix --path=../../pix
\${php} svgtool.php --noaspectratio
";

```

```

    echo $help;
    die;
}

```

```

exit(0);

```

```

/**

```

```

 * Fixes SVG images for IE9.

```

```

 *

```

```

 * @param string $file

```

```

 */

```

```

function core_admin_svgtool_ie9fix($file) {
    global $CFG;

```

```

    if (strpos($file, $CFG->dirroot.DIRECTORY_SEPARATOR) === 0) {

```

```

        $relfile = substr($file, strlen($CFG->dirroot));

```

```

    } else {

```

```

        $relfile = $file;

```

```

    }

```

```

    $content = file_get_contents($file);

```

```

if (!preg_match('/<svg\s[^\>]*>/', $content, $matches)) {
    echo " skipping $relfile (invalid format)\n";
    return;
}
$svg = $matches[0];
if (strpos($svg, 'preserveAspectRatio') !== false) {
    return;
}

if (!is_writable($file)) {
    echo " skipping $relfile (can not modify file)\n";
    return;
}

$newsvg = rtrim($svg, '>').' preserveAspectRatio="xMinYMid meet">';

$content = str_replace($svg, $newsvg, $content);
echo "converting $relfile\n";
file_put_contents($file, $content);
}

/**
 * Removes preserveAspectRatio attributes from SVG images.
 *
 * @param string $file
 */
function core_admin_svgtool_noaspectratio($file) {
    global $CFG;

    if (strpos($file, $CFG->dirroot.DIRECTORY_SEPARATOR) === 0) {
        $relfile = substr($file, strlen($CFG->dirroot));
    } else {
        $relfile = $file;
    }

    $content = file_get_contents($file);

    if (!preg_match('/<svg\s[^\>]*>/', $content, $matches)) {
        echo " skipping $relfile (invalid format)\n";
        return;
    }
    $svg = $matches[0];
    if (strpos($svg, 'preserveAspectRatio="xMinYMid meet"') === false) {
        return;
    }

    if (!is_writable($file)) {
        echo " skipping $relfile (can not modify file)\n";
        return;
    }
}

```

```

$newsVG = preg_replace('/ ?preserveAspectRatio="xMinYMid meet"/, "", $svg);

$content = str_replace($svg, $newsVG, $content);
echo "resetting $relfile\n";
file_put_contents($file, $content);
}

/**
 * Recursively works through directories of this theme, finding and fixing SVG images.
 *
 * @param string $base
 * @param string $sub
 * @param string $filecallback
 * @param array $blacklist
 */
function core_admin_recurse_svgs($base, $sub, $filecallback, $blacklist) {
    if (is_dir("$base/$sub")) {
        $items = new DirectoryIterator("$base/$sub");
        foreach ($items as $item) {
            if ($item->isDot()) {
                continue;
            }
            $file = $item->getFilename();
            core_admin_recurse_svgs("$base/$sub", $file, $filecallback, $blacklist);
        }
        unset($item);
        unset($items);
        return;
    } else if (is_file("$base/$sub")) {
        if (substr($sub, -4) !== '.svg') {
            return;
        }
        $file = realpath("$base/$sub");
        if (in_array($file, $blacklist)) {
            return;
        }
        $filecallback($file);
    }
}
}
<?php

// This file is part of Moodle - http://moodle.org/
//
// Moodle is free software: you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License, or
// (at your option) any later version.
//
// Moodle is distributed in the hope that it will be useful,

```

```

// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License
// along with Moodle. If not, see <http://www.gnu.org/licenses/>.

/**
 * This script creates config.php file and prepares database.
 *
 * This script is not intended for beginners!
 * Potential problems:
 * - su to apache account or sudo before execution
 * - not compatible with Windows platform
 *
 * @package core
 * @subpackage cli
 * @copyright 2009 Petr Skoda (http://skodak.org)
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or later
 */

// Force OPcache reset if used, we do not want any stale caches
// when detecting if upgrade necessary or when running upgrade.
if (function_exists('opcache_reset') and !isset($_SERVER['REMOTE_ADDR'])) {
    opcache_reset();
}

define('CLI_SCRIPT', true);
define('CACHE_DISABLE_ALL', true);

require(__DIR__.'../../config.php');
require_once($CFG->libdir.'/adminlib.php'); // various admin-only functions
require_once($CFG->libdir.'/upgradelib.php'); // general upgrade/install related functions
require_once($CFG->libdir.'/clilib.php'); // cli only functions
require_once($CFG->libdir.'/environmentlib.php');

// now get cli options
$lang = isset($SESSION->lang) ? $SESSION->lang : $CFG->lang;
list($options, $unrecognized) = cli_get_params(
    array(
        'non-interactive' => false,
        'allow-unstable' => false,
        'help' => false,
        'lang' => $lang
    ),
    array(
        'h' => 'help'
    )
);

if ($options['lang']) {

```

```

    $SESSION->lang = $options['lang'];
}

$interactive = empty($options['non-interactive']);

if ($unrecognized) {
    $unrecognized = implode("\n ", $unrecognized);
    cli_error(get_string('cliunknowoption', 'admin', $unrecognized));
}

if ($options['help']) {
    $help =
"Command line Moodle upgrade.
Please note you must execute this script with the same uid as apache!

```

Site defaults may be changed via local/defaults.php.

Options:

```

--non-interactive    No interactive questions or confirmations
--allow-unstable     Upgrade even if the version is not marked as stable yet,
                    required in non-interactive mode.
--lang=CODE         Set preferred language for CLI output. Defaults to the
                    site language if not set. Defaults to 'en' if the lang
                    parameter is invalid or if the language pack is not
                    installed.
-h, --help         Print out this help

```

Example:

```

\sudo -u www-data /usr/bin/php admin/cli/upgrade.php
"; //TODO: localize - to be translated later when everything is finished

```

```

    echo $help;
    die;
}

```

```

if (empty($CFG->version)) {
    cli_error(get_string('missingconfigversion', 'debug'));
}

```

```

require("$CFG->dirroot/version.php"); // defines $version, $release, $branch and $maturity
$CFG->target_release = $release; // used during installation and upgrades

```

```

if ($version < $CFG->version) {
    cli_error(get_string('downgradedcore', 'error'));
}

```

```

$oldversion = "$CFG->release ($CFG->version)";
$newversion = "$release ($version)";

```

```

if (!moodle_needs_upgrading()) {
    cli_error(get_string('cliupgradenoneed', 'core_admin', $newversion), 0);
}

```

```

}

// Test environment first.
list($envstatus, $environment_results) = check_moodle_environment(normalize_version($release),
ENV_SELECT_RELEASE);
if (!$envstatus) {
    $errors = environment_get_errors($environment_results);
    cli_heading(get_string('environment', 'admin'));
    foreach ($errors as $error) {
        list($info, $report) = $error;
        echo "!! $info !!\n$report\n\n";
    }
    exit(1);
}

// Test plugin dependencies.
$failed = array();
if (!$core_plugin_manager::instance()->all_plugins_ok($version, $failed)) {
    cli_problem(get_string('pluginscheckfailed', 'admin', array('pluginslist' => implode(', ',
array_unique($failed))));
    cli_error(get_string('pluginschecktodo', 'admin'));
}

if ($interactive) {
    $a = new stdClass();
    $a->oldversion = $oldversion;
    $a->newversion = $newversion;
    echo cli_heading(get_string('databasechecking', '', $a)) . PHP_EOL;
}

// make sure we are upgrading to a stable release or display a warning
if (isset($maturity)) {
    if (($maturity < MATURITY_STABLE) and !$options['allow-unstable']) {
        $maturitylevel = get_string('maturity' . $maturity, 'admin');

        if ($interactive) {
            cli_separator();
            cli_heading(get_string('notice'));
            echo get_string('maturitycorewarning', 'admin', $maturitylevel) . PHP_EOL;
            echo get_string('morehelp') . ': ' . get_docs_url('admin/versions') . PHP_EOL;
            cli_separator();
        } else {
            cli_problem(get_string('maturitycorewarning', 'admin', $maturitylevel));
            cli_error(get_string('maturityallowunstable', 'admin'));
        }
    }
}

if ($interactive) {
    echo html_to_text(get_string('upgradesure', 'admin', $newversion))."\n";
    $prompt = get_string('cliyesnoprompt', 'admin');
}

```

```

    $input = cli_input($prompt, "", array(get_string('clianswer', 'admin'), get_string('clianswer',
'admin')));
    if ($input == get_string('clianswer', 'admin')) {
        exit(1);
    }
}

if ($version > $CFG->version) {
    // We purge all of MUC's caches here.
    // Caches are disabled for upgrade by CACHE_DISABLE_ALL so we must set the first arg to true.
    // This ensures a real config object is loaded and the stores will be purged.
    // This is the only way we can purge custom caches such as memcache or APC.
    // Note: all other calls to caches will still use the disabled API.
    cache_helper::purge_all(true);
    upgrade_core($version, true);
}
set_config('release', $release);
set_config('branch', $branch);

// unconditionally upgrade
upgrade_noncore(true);

// log in as admin - we need doanything permission when applying defaults
\core\session\manager::set_user(get_admin());

// apply all default settings, just in case do it twice to fill all defaults
admin_apply_default_settings(NULL, false);
admin_apply_default_settings(NULL, false);

// This needs to happen at the end to ensure it occurs after all caches
// have been purged for the last time.
// This will build a cached version of the current theme for the user
// to immediately start browsing the site.
upgrade_themes();

echo get_string('cliupgradefinished', 'admin')."\\n";
exit(0); // 0 means success

```