**LAPORAN UJIAN AKHIR SEMESTER**

****

Mata Kuliah: Pemrograman Berorientasi Objek

Topik: Math Quiz Game

Kelas: MI 2022C

Nama: Ahmad Balya Diwanullatif

NIM: 22091397095

**Link Sumber Kode Program:**

<https://github.com/22091397095-AhmadBalyaDiwanullatifMIC22/UAS-PBO.git>

**PROGRAM STUDI D4 MANAJEMEN INFORMATIKA**

**FAKULTAS VOKASI**

**UNIVERSITAS NEGERI SURABAYA**

**2023**

**Penjelasan Aplikasi:**

Aplikasi yang Anda buat adalah sebuah permainan matematika sederhana berbasis teks menggunakan bahasa pemrograman PHP. Aplikasi ini terdiri dari tiga file kode utama: index.php, class\_ui.php, dan class\_game.php. Berikut adalah penjelasan mengenai fungsionalitas dan struktur aplikasi:

1. index.php

* Inisialisasi Kelas: Membuat instance dari dua kelas utama, yaitu game dan ui.
* Loop Utama: Menampilkan menu utama dan memberikan pilihan kepada pengguna untuk memulai permainan atau keluar dari aplikasi.
* Switch Case: Menangani pilihan pengguna dengan menggunakan struktur switch case, di mana pilihan 1 memulai permainan, dan pilihan 2 keluar dari aplikasi.
* Perulangan Do-While: Melakukan loop selama pengguna tidak memilih untuk keluar (pilihan 2).

2. class\_ui.php

* Kelas ui: Berisi fungsi-fungsi yang berhubungan dengan antarmuka pengguna (UI).
* showMenu: Menampilkan menu utama permainan dengan opsi untuk memulai permainan atau keluar.
* chooseLevel: Menampilkan opsi level dan meminta pengguna untuk memilih level permainan.

3. class\_game.php

* Kelas game: Berisi logika permainan matematika.
* Properti Privat: Menyimpan skor pengguna dan jumlah pertanyaan total.
* setLevel: Mengatur level permainan berdasarkan pilihan pengguna.
* startGame: Memulai permainan dengan menampilkan selamat datang dan menjalankan loop untuk menghasilkan pertanyaan matematika kepada pengguna.
* generateQuestion: Menghasilkan pertanyaan matematika secara acak.
* getUserAnswer: Meminta jawaban dari pengguna untuk pertanyaan yang diberikan.
* checkAnswer: Memeriksa jawaban pengguna dan memberikan skor berdasarkan jawaban yang benar atau salah.

Cara Kerja Aplikasi:

* Pengguna menjalankan index.php.
* Tampilan menu utama ditampilkan oleh kelas ui.
* Pengguna memilih untuk memulai permainan atau keluar.
* Jika memilih memulai permainan, kelas game diinisialisasi, dan pengguna diminta untuk memilih level.
* Permainan dimulai dengan menampilkan pertanyaan matematika kepada pengguna sesuai dengan level yang dipilih.
* Setelah pengguna menjawab semua pertanyaan, skor akhir ditampilkan.

**Fitur Pada Aplikasi:**

1. Menu Utama:

* Tampilan menu utama dengan informasi versi aplikasi.
* Pilihan untuk memulai permainan (Level 1-5) atau keluar dari aplikasi.

2. Pilihan Level:

* Pengguna dapat memilih level permainan dengan jumlah pertanyaan yang berbeda.
* Level 1: 5 pertanyaan
* Level 2: 10 pertanyaan
* Level 3: 15 pertanyaan
* Level 4: 20 pertanyaan
* Level 5: 25 pertanyaan

3. Permainan Matematika:

* Pengguna diberikan pertanyaan matematika secara acak.
* Pertanyaan terdiri dari dua bilangan dan operator matematika (+, -, \*, /).
* Pengguna diminta untuk memberikan jawaban.

4. Penilaian dan Skor:

* Setelah menjawab pertanyaan, pengguna diberikan umpan balik apakah jawaban benar atau salah.
* Skor pengguna diperbarui setiap kali jawaban benar.
* Skor akhir dan jumlah pertanyaan ditampilkan setelah selesai bermain.

5. Logika Permainan:

* Pemilihan level mengatur jumlah pertanyaan yang akan muncul selama permainan.
* Pengguna mendapatkan skor berdasarkan jumlah jawaban benar.

6. Kontrol Permainan:

* Struktur loop (do-while) memungkinkan pengguna untuk terus memainkan game atau keluar dari aplikasi.
* Pilihan tidak valid memberikan pesan kesalahan dan otomatis memilih Level 1.

7. Antarmuka Pengguna (UI):

* Tampilan menu dan pilihan level dirancang secara sederhana menggunakan teks.

8. Random Question Generation:

* Pertanyaan matematika dihasilkan secara acak dengan bilangan dan operator matematika yang berbeda.

9. Informasi Konfigurasi:

* Level dan jumlah pertanyaan setiap level diatur di kelas game.

10. Pesan Informasi:

* Pesan informasi, seperti pesan selamat datang dan pesan keluar, ditampilkan selama bermain.

**Kode Program:**

Index.php:

<?php

include 'class\_game.php';

include 'class\_ui.php';

$game = new game();

$ui = new ui();

do {

    $ui->showMenu();

    $choice = trim(fgets(STDIN));

    switch ($choice) {

        case '1':

            $level = $ui->chooseLevel();

            $game->setLevel($level);

            $game->startGame();

            break;

        case '2':

            echo "Mengeluarkan Game. Dadahh!\n";

            break;

        default:

            echo "Pilihan Salah. Tolong coba lagi.\n";

    }

} while ($choice !== '2');

?>

**Penjelasan Kode**:  
**1. Include Statements**:

* Kode dimulai dengan dua pernyataan include untuk mengimpor file kelas class\_game.php dan class\_ui.php. Ini memungkinkan penggunaan kelas-kelas tersebut dalam file utama.

**2. Inisialisasi Objek**:

* Objek dari dua kelas utama, yaitu game dan ui, dibuat menggunakan sintaks new.

**3. Loop Utama (do-while)**:

* Sebuah loop do-while digunakan untuk menjalankan permainan. Loop ini terus berjalan selama pengguna tidak memilih opsi untuk keluar (memilih '2').

**4. Menampilkan Menu**:

* Method showMenu dari objek ui dipanggil untuk menampilkan menu utama permainan.

**5. Meminta Input Pengguna**:

* Menggunakan fgets(STDIN), kode meminta input dari pengguna untuk memilih opsi dari menu.

**6. Switch Case**:

* Struktur switch case digunakan untuk menangani pilihan pengguna.
* Jika pengguna memilih '1', maka permainan dimulai dengan meminta pengguna untuk memilih level dan menjalankan permainan.
* Jika pengguna memilih '2', maka pesan keluar ditampilkan, dan permainan berakhir.
* Untuk pilihan lainnya, pesan kesalahan ditampilkan, dan pengguna diminta untuk mencoba lagi.

**7. Loop Berdasarkan Pilihan Pengguna**:

* Loop do-while akan terus berjalan sampai pengguna memilih untuk keluar (memilih '2').

Class\_game.php:

<?php

class game {

    private $score = 0;

    private $totalQuestions;

    public function setLevel($level) {

        switch ($level) {

            case '1':

                $this->totalQuestions = 5;

                break;

            case '2':

                $this->totalQuestions = 10;

                break;

            case '3':

                $this->totalQuestions = 15;

                break;

            case '4':

                $this->totalQuestions = 20;

                break;

            case '5':

                $this->totalQuestions = 25;

                break;

            default:

                echo "Level tidak valid. Mengatur ke Level 1.\n";

                $this->totalQuestions = 5;

        }

    }

    public function startGame() {

        echo "Selamat Datang di game berbasis teks PHP!\n";

        echo "Jawab pertanyaan dibawah yaa!:\n";

        for ($i = 1; $i <= $this->totalQuestions; $i++) {

            $question = $this->generateQuestion();

            $userAnswer = $this->getUserAnswer($question);

            if ($this->checkAnswer($question, $userAnswer)) {

                echo "Benar!\n";

                $this->score++;

            } else {

                echo "Salah. Jawaban yang benar adalah: " . $question['answer'] . "\n";

            }

        }

        echo "Game Selesai! Skor Final Kamu Adalah: " . $this->score . " Dari " . $this->totalQuestions . "\n";

    }

    private function generateQuestion() {

        $num1 = rand(1, 10);

        $num2 = rand(1, 10);

        $operator = ['+', '-', '\*', '/'] [rand(0, 3)];

        $answer = 0;

        switch($operator){

            case '+':

                $answer = $num1 + $num2;

                break;

            case '-':

                $answer = $num1 - $num2;

                break;

            case '\*':

                $answer = $num1 \* $num2;

                break;

            case '/':

                $answer = $num1 / $num2;

                break;

        }

        return [

            'question' => "$num1 $operator $num2",

            'answer' => $answer

        ];

    }

    private function getUserAnswer($question) {

        echo "Pertanyaan: " . $question['question'] . " = ";

        return trim(fgets(STDIN));

    }

    private function checkAnswer($question, $userAnswer) {

        return (int)$userAnswer === $question['answer'];

    }

}

?>

**Penjelasan Kode:**

**1. Properti Privat**:

* Kelas memiliki dua properti privat, yaitu $score untuk menyimpan skor pengguna dan $totalQuestions untuk menyimpan jumlah pertanyaan yang akan ditampilkan.

**2. Metode setLevel**:

* Metode ini menerima parameter $level dan mengatur jumlah total pertanyaan berdasarkan level yang dipilih.
* Menggunakan struktur switch case untuk menentukan jumlah pertanyaan berdasarkan level yang dipilih.
* Jika level tidak valid, pesan kesalahan ditampilkan dan level diatur ke level 1 dengan 5 pertanyaan.

**3. Metode startGame**:

* Metode ini memulai permainan matematika.
* Menampilkan pesan selamat datang dan instruksi permainan.
* Menggunakan loop for untuk menghasilkan pertanyaan sejumlah total pertanyaan yang sudah diatur.
* Untuk setiap pertanyaan, pengguna diminta jawaban, dan jawaban tersebut diperiksa.
* Skor pengguna diperbarui berdasarkan jawaban yang benar atau salah.
* Setelah selesai, ditampilkan skor final pengguna.

**4. Metode generateQuestion**:

* Metode ini menghasilkan pertanyaan matematika secara acak.
* Memilih dua bilangan acak ($num1 dan $num2) dan operator matematika acak ($operator).
* Menghitung jawaban berdasarkan operasi matematika yang dipilih.

**5. Metode getUserAnswer**:

* Metode ini meminta jawaban dari pengguna untuk pertanyaan yang diberikan.
* Menampilkan pertanyaan dan mengembalikan jawaban setelah di-trim.

**6. Metode checkAnswer**:

* Metode ini memeriksa apakah jawaban pengguna benar atau salah.
* Membandingkan jawaban pengguna dengan jawaban yang benar dan mengembalikan hasilnya.

Class\_ui.php:

<?php

class ui {

    public function showMenu() {

        echo "=====================\n";

        echo "|                   |\n";

        echo "|   Math Quiz Game  |\n";

        echo "|    Version 2.0    |\n";

        echo "|                   |\n";

        echo "=====================\n";

        echo "===== Main Menu =====\n";

        echo "|| [1] Mulai Game  ||\n";

        echo "|| [2] Keluar      ||\n";

        echo "=====================\n";

        echo "Masukkan Angka: ";

    }

    public function chooseLevel() {

        echo "Pilih Level:\n";

        echo "[1] Level 1 (5 pertanyaan)\n";

        echo "[2] Level 2 (10 pertanyaan)\n";

        echo "[3] Level 3 (15 pertanyaan)\n";

        echo "[4] Level 4 (20 pertanyaan)\n";

        echo "[5] Level 5 (25 pertanyaan)\n";

        echo "Masukkan Angka: ";

        return trim(fgets(STDIN));

    }

}

?>

**Penjelasan Kode:**

**1. Metode showMenu**:

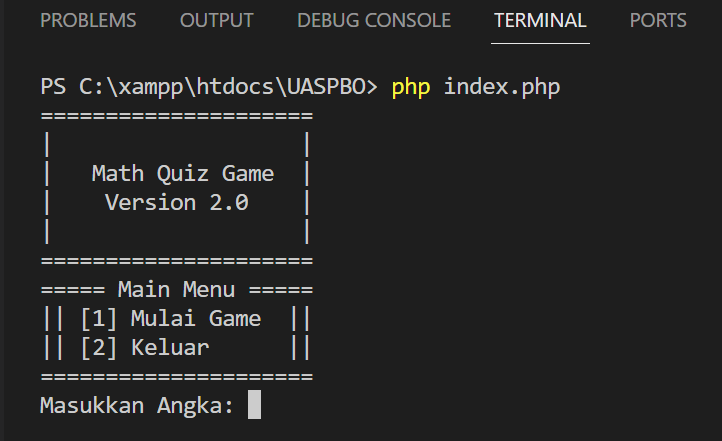
* Metode ini menampilkan menu utama permainan.
* Menampilkan judul permainan, versi, dan tata letak menu dengan opsi untuk memulai permainan atau keluar.
* Menampilkan instruksi untuk memasukkan angka.

**2. Metode chooseLevel**:

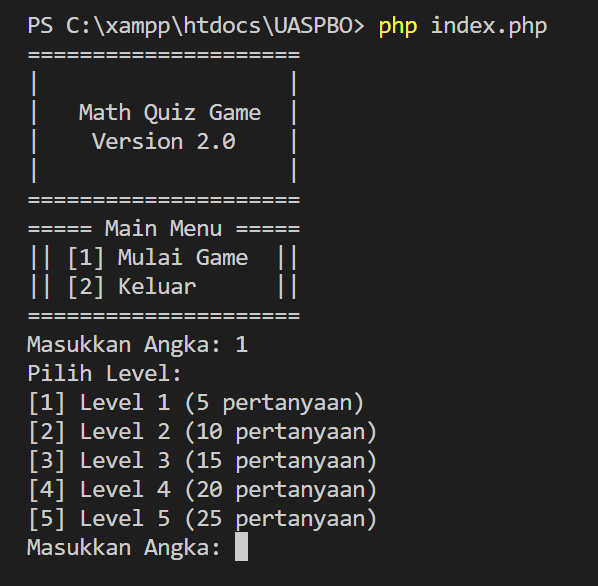
* Metode ini menampilkan opsi level untuk dipilih oleh pengguna.
* Menampilkan level 1 hingga 5 dengan jumlah pertanyaan yang sesuai.
* Meminta pengguna untuk memasukkan angka untuk memilih level.
* Mengembalikan input pengguna setelah di-trim.

**Tampilan UI Aplikasi basis Teks:**

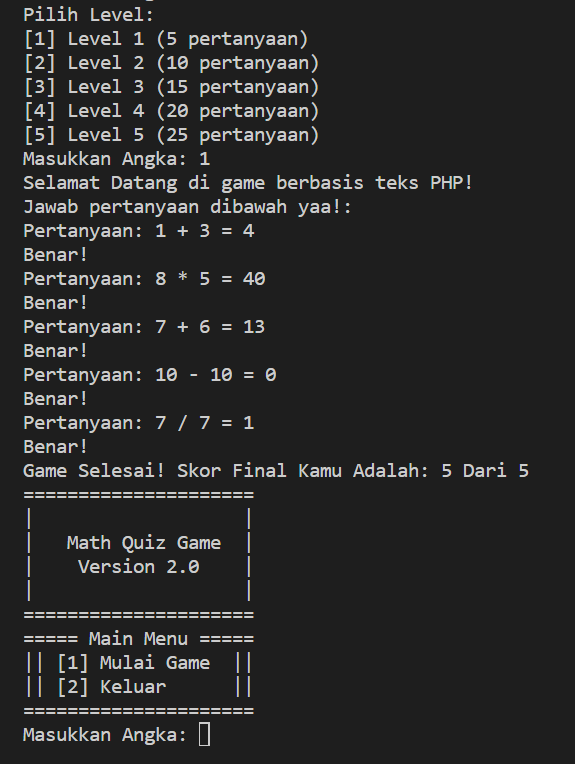
Main Menu:



Mulai Game:



Masuk Game setelah memilih level:



Keluar dari game:

