***A gyakorlati vizsgafeladat jellemzői***

A 12012-16 azonosító számú Webfejlesztés I. megnevezésű szakmai követelménymodulhoz tartozó tantárgyak és témakörök oktatása során fejlesztendő kompetenciák:

Feladatok:

* Kiválasztja a tervezett fejlesztéshez szükséges internet/intranet technológiát az elterjedt hálózati szabványok, protokollok ismeretében
* Részt vesz komplex webportálok fejlesztésében
* Interaktív web oldalakat készít (programoz)
* MVC elvű web alkalmazásokat készít
* JavaScript keretrendszerek segítségével interaktivitás ad a weboldalakhoz
* Egyszerű szerver oldali programozási feladatot lát el
* Web alkalmazásokat tesztel
* Angol nyelvű szakmai szöveget értelmez
* Betartja a munka-, baleset-, tűz- és környezetvédelmi előírásokat

Szakmai ismeretek és szakmai készségek:

* HTML5, CSS3, JSON, XML ismeretek
* AJAX webfejlesztési technika alapjai
* JavaScript vagy TypeScript nyelv alapismerete
* JavaScript könyvtárak alapszintű használata (jQuery)
* Szerveroldali programozási nyelv alapismerete (PHP)
* MVC (Model-View-Controller) architektúra

Szakmai készségek:

* Kódolás leíró nyelv segítségével
* Szövegesen megfogalmazott feladatok, specifikációk vizualizálása
* Forráskód konstrukciók megértése az ismert programozási nyelveken
* Integrált fejlesztői keretrendszer használata

Módszertani kompetenciák

* Logikus gondolkodás
* Hibakeresés (diagnosztizálás)
* Problémamegoldás, hibaelhárítás

Személyes kompetenciák

* Precizitás
* Megbízhatóság
* Döntésképesség
* Önállóság

***Útmutató a feladat megoldásához***

A gyakorlati vizsgán egy összetett web programozási feladatot kell megoldania számítógépen,  
az Ön általa megismert szoftverfejlesztő eszközök felhasználásával.

Gyakorlati tanácsok:

Törekedjen azoknak a feladatoknak a megoldására, amelyeket a feladatban talál!

Ha a feladat nem ír elő konkrét megoldási módszert, akkor olyan megoldási módszert válasszon, amely a leggyorsabb feladatmegoldást eredményezi!

A kód tesztelése elsősorban a megadott vagy a fejlesztés során az adatbázisban felvitt adatokra történjen meg!

Az itt megadott feladatok megoldása során használja az Ön által korábban megismert fejlesztői eszközök egyikét!

A feladat tesztelésére a helyi számítógépre telepített webszerver és MySQL adatbázis lesz segítségére. A webszerver dokumentroot-ja a **C:\XAMPP\htdocs** mappa. A php kiterjesztésű állományokat a webszerver automatikusan feldolgozza, csak az eredmény kerül megjelenítésre a böngészőben.

*A programkód elkészítése során törekedjen tagolt és jól áttekinthető programkód készítésére!*

Ahol a feladat megoldása során szükséges ‑ leginkább az adatbázissal kapcsolatos műveletek során ‑ ne feledkezzen meg a hibák kezeléséről sem! A felhasználó és egyéb látogatók számára az alkalmazás az adatbázis szerkezetéről információt **NE** szolgáltasson!

A weblapon a diákok számára cégek által meghirdetett egynapos munkalehetőségek adatai szerepelnek. A forrás kiegészítésére, értelmezésére van szükség ahhoz, hogy az adatbázisban tárolt adatokat megjelenítsék vagy bővítsék.

Az alkalmazás oldalának vázát a következő kódrészlet szemlélteti:

<body>

<div id="tartalom">

<header>

</header>

<div id="adatok">

</div>

</div>

<footer>

</footer>

</body>

A feladat megkezdése előtt importálja a diakmunka.sql fájlból az adatbázist a helyi MySQL adatbázis-kezelő rendszerbe. Az SQL állomány tartalmazza az adatbázis létre hozásához szükséges parancsot is. Az importálást követően az alkalmazásba bejelentkezni az adatbázis **felhasznalo** táblájában tárolt **nev** és **jelszo** mezők segítségével lehet (pl admin, admin).

A munka tábla tartalmazza a diákok részére elérhető feladatokat, **munkaado** tábla viszont a munkaadók nevét és székhelyük települését tartalmazza.

**Kapcsolat az adatbázis-szerverrel:**

**1. feladat**

Az előbbiekben importált adatbázishoz **diakmunka** felhasználói névvel és jelszóval jogosultságot állítson be, mely segítségével adatokat kérdezhet le vagy adatokat módosíthat. Amennyiben nem tudja a jogosultságot beállítani, akkor az adatbázis-kapcsolat konfigurációja során használja a **root** felhasználót, mely fiók nincs jelszóval védve.

**2. feladat**

A **config** mappába **connect.php** néven hozza létre az adatbázis csatlakozásához szükséges konfigurációs fájlt! Az adatbázis kapcsolatának leírására használt változó neve: $**link**. A kapcsolat beállítása során a szükséges beállításokat és ellenőrzéseket hajtsa végre!

Az adatbázisban tárolt adatok UTF-8 kódolásúak.

A megoldás során a hibák kezelését is valósítsa meg!

**Adatok rendezett megjelenítése:**

**3. feladat**

A webes alkalmazásban bármilyen jogosultsággal hitelesített felhasználó a „Munkaadók” menüelemre kattintva táblázatban jelenítse meg a munkaadók listáját!

A projekt **php** mappájába készítse el a **munkaadok.php** fájlt, mely megvalósítja az adatbázisban tárolt munkaadók listájának lekérdezését és megjeleníthető formába rendezését.

A feladat megoldása során lehetőleg alkalmazza a **jQuery** és **AJAX** technológiákat.

A projekt **JS** mappájában megtalálható a **jQuery** **API**.

A lekérdezett adatokat egy kétoszlopos szegélyezett táblázatban jelenítse meg!

Amennyiben senki nem hitelesítette magát a webes alkalmazásban, akkor irányítsa át a **jog.php** oldalra!

**Űrlap feldolgozása:**

**4. feladat**

Admin felhasználóként bejelentkezve az „Új munkaadó” menüelemre kattintva két bevitelei mezőből álló űrlap jelenjen meg. Az űrlaphoz kapcsolódó gomb felirata a „Munkaadó rögzítése” legyen!

**HTML5** technológia segítségével írja elő a mezők kitöltését!

Az űrlapot feldolgozásra elküldeni csak abban az esetben lehetséges, ha a kliensoldali ellenőrzése is megtörténik, tehát mindkét mező kitöltésre került. Üres vagy nem megfelelő kitöltés esetén értesítse a látogatót a helytelen kitöltésről! Ellenőrizni kell, hogy mindkét mezőt kitöltötték-e, valamit **nem** csak nyomtatásban nem látható karaktereket tartalmaz.

A feladat megoldása során lehetőleg alkalmazza a **jQuery** és **AJAX** technológiákat.

A projekt **JS** mappájában megtalálható a **jQuery** **API**.

Amennyiben senki nem hitelesítette magát a webes alkalmazásban, akkor irányítsa át a **jog.php** oldalra!

**HTML5, CSS:**

**5. feladat**

Az alkalmazáshoz csatolt stíluslapot módosítsa, úgy hogy a **not-logged** osztályú elem a következő tulajdonságokkal rendelkezzen:

* sötétkék folytonos kerettel látja el az üzenetet
* a keret sarkait lekerekíti
* szélességet 500 képpontra állítja be
* alsó- és felső margót alkalmaz, középre igazítja
* belső margót az aktuális elemre érvényes betűméret felére állítja
* feltűnő betűszín
* kétszeres betűméret
* eltérő betűtípust alkalmaz
* megjelenő szöveg félkövér