

מבדק מסכם - node-javascript

שאלה 1

כאשר בונים אפליקציית Backend עם Express אנחנו מגדירים שרשרת של פונקציות middleware (ראה תמונה)

```
app.get('/hello', function(req, res) {  
  res.send('hello world');  
});
```

עבור איזה סוג של request שנכנס, express יפעיל עבורו את פונקציית ה- middleware .

בחר את התשובה הנכונה:

1. עבור כל request שיש עבורו התאמה ב- method כלומר בדוגמא לעיל עבור request שה- method שלו הוא get.
2. עבור כל request שיש עבורו התאמה ב- path כלומר בדוגמא לעיל עבור request שה- path שלו הוא /hello
3. עבור request שמכיל את ה- headers הבאים:
 - a. headers['path'] = '/hello'
 - b. headers['method'] = 'get'
4. עבור request שה- method שלו הוא מסוג get והנתיב שלו הוא /hello

שאלה 2

מה הפונקציה הבאה מחזירה:

```
async function yoniNumber() {
  ... return 42;
}
```

1. 42
2. Promise
3. '42'
4. null

שאלה 3

אלו מהפעולות הבאות middleware של Express יכול לבצע:

1. להעביר ל middleware הבא על ידי קריאת ל- next
2. להחזיר response
3. להעביר שגיאה על ידי next(new Error)
4. תשובות 1,2,3 נכונות

שאלה 4

איך Session ו- Cookies קשורים אחד לשני?

1. הם לא קשורים
2. כל המידע של ה- session נשמר ב- cookies על גבי הדפדפן
3. ב- cookies נהוג לשמור את ה- session id שבאמצעותו השרת יכול לאתר את המידע ששמור ב- session על המשתמש
4. cookies ו session זה מילים נרדפות

שאלה 5

איך עובד אימות באמצעות JWT ?

1. ה- client מפיק JWT token ושולח אותו ב- Authorization header.

2. ה- client מפיק JWT token מפרטי ההתחברות של המשתמש
3. השרת מפיק JWT token ושומר אותו ב- session
4. השרת מפיק jwt token ושולח ל- client, ה- client צריך לדאוג לשלוח את ה- jwt token דרך ה- Authorization header

שאלה 6

עבור בסיס נתונים רלציוני, כיצד נקשר בין טבלת משתמשים לטבלת כלבים כאשר לכל משתמש יכול להיות מספר רב של כלבים.

1. מדובר על קשר one-2-one ואני צריך לשים עמודת dogId בטבלת המשתמשים
2. מדובר על קשר many-2-many הקשר צריך להיעשות דרך טבלה מקשרת שמכילה עמודת dogId ו- userId
3. מדובר על קשר one-2-many ואני צריך לשים עמודת userId בתוך טבלת הכלבים
4. אי אפשר לייצג קשר כזה בבסיסי נתונים רלציונים וניאלץ להשתמש ב- mongodb

שאלה 7

מהי הדרך הנכונה לבצע שינויים בסכמה של בסיסי נתונים רלציוניים

1. על ידי סקריפט של migration בלבד
2. על ידי שאילתת SQL שמזינים ישירות לבסיס הנתונים
3. לא ניתן לבצע שינויים בסכמה של בסיסי נתונים רלציוניים
4. על ידי תוכנה שמתחברת ישירות לבסיס הנתונים עם gui כמו pgAdmin

שאלה 8

נתונים הפונקציות הבאות:

- readFileSync שקוראת תוכן של קובץ בצורה סינכרונית
- readFile שקוראת קובץ בצורה אסינכרונית

באיזה פונקציה תעדיף להשתמש ב- Javascript ולמה?

1. readFileSync - הרבה יותר קלה לשימוש, סינכרונית, ואני לא צריך לספק callback
2. readFileSync - היא סינכרונית ותחסום משימות אחרות מה שיבטיח שלא יפריעו לקריאת הקובץ
3. readFile - האסינכרונית תאפשר לי לדחות את ביצוע המשימה ולעשות משימות חשובות יותר

4. `readFile` - פונקציות האסינכרוניות משתמשות פעמים רבות ב- `api` יעילים שמוגדרים בחלק של ה- `++c` של `node` ומנצלים את ה `kernel` וריבוי `thread` לביצוע מהיר יותר של המשימה

שאלה 9

`Passport` מאפשר לנו לאמת משתמש. הגדרנו את `Passport` וה- `client` הצליח לבצע `authentication` בהצלחה.
פונקציית `middleware` של `express` קפצה לאחר אימות מוצלח - מה `passport` הולך להוסיף בפונקציה הזאת:

1. `passport` ישלח ארגומנט נוסף לפונקציית ה- `middleware` עם המשתמש שנכנס:
`function(req, res, next, user) { ... }`
2. `passport` ירחיב את ה- `request object` ויוסיף אליו שדה של `user`
`function(req, res, next) { req.user }`
3. ניתן לעשות `require` ל- `passport` ולשלוח את המשתמש שנכנס מבסיס הנתונים
4. `passport` לא מעביר לנו מידע נוסף על המשתמש

שאלה 10

מה נכון לגבי משתני `let`, `const`, `var`

1. למשתנה `const` אפשר לתת אינסוף השמות, למשתנה `let` השמה אחת, למשתנה `var` אינסוף השמות
2. ה- `scope` של משתנה `let` הוא פונקציונלי
3. ה- `scope` של משתנה `const` הוא פונקציונלי
4. למשתנה `const` חייב לתת השמה כאשר מגדירים אותו ואסור לתת יותר מהשמה אחת.