FIT2095 e-Business software technologies - S2 2021

Started on	Sunday, 22 August 2021, 4:34 PM
State	Finished
Completed on	Sunday, 22 August 2021, 4:39 PM
Time taken	4 mins 53 secs
Grade	**10.00** out of 10.00 (**100**%)

With respect to MongoDB features, which of the following is false?

Select one:

○ a.　MongoDB supports Ad-Hoc Queries

◉ b.　only primary key can be used as an index　　　　　✔

○ c.　document data structure is composed of field and value pairs.

○ d.　MongoDB high availability is achieved by the replication facility, which is called replica set

Your answer is correct.

The correct answer is: only primary key can be used as an index

The table of relational database management system is called ------ in MongoDB

Select one:

○ a.　Table

◉ b.　Collection　　　　　✔

○ c.　Document

○ d.　Book

○ e.　Record

Your answer is correct.

The correct answer is: Collection

With respect to the only requirement in MongoDB collection, which of the following is true?

Select one:

○ a.　require to define a schema for a collection

◉ b.　each document must have a unique _id　　　　　✔

○ c.　binary data can not be stored as a value of a field

○ d.　documents can not be empty

Your answer is correct.

The correct answer is: each document must have a unique _id

---

assume you have this fragment of code:

```
let query = { address: /n$/ };
db.collection("warehouse").find(query).toArray(function (err, result) {
 if (err) throw err;
 console.log(result);
});
```

The result will be:

Select one:

- a.   all the documents with address field that ends with the suffix '/n$/'
- ◉ b.   all the documents with address field that end with the letter 'n'   ✔
- c.   the last document with address field that ends with the letter 'n'
- d.   all the documents with address field that starts with the letter 'n'
- e.   the first document with address field that ends with the letter 'n'

Your answer is correct.

The correct answer is: all the documents with address field that end with the letter 'n'

---

assume you have this fragment of code:

```
let query = { quantity: { $gte: 25 } };
db.collection("warehouse").find(query).toArray(function (err, result) {
 if (err) throw err;
 console.log(result);
});
```

The result will be:

Select one:

- a.   all the documents with field quantity that has the object { $gte: 25 } in its value
- ◉ b.   all the documents with field quantity that has value greater than or equal to 25   ✔
- c.   all the documents with field quantity that has value not greater than or equal to 25
- d.   the first document with field quantity that has value greater than or equal to 25
- e.   all the documents with field quantity that has value greater 25

Your answer is correct.

The correct answer is: all the documents with field quantity that has value greater than or equal to 25

Assume you have this fragment of code:

```
let query = { quantity: { $lte: 25 } };
let sortBy = { price: -1 }
db.collection("warehouse").find(query).sort(sortBy).limit(5).toArray(function (err, result) {
 if (err) throw err;
 console.log(result);
});
```

The result will be:

Select one:

○ a. the first 5 documents of a list sorted in descending order by attribute price with quantity attribute less than or equal 25 ✔

○ b. the first 5 documents of a list sorted in ascending order by attribute price with quantity attribute less than or equal 25

○ c. the first document of a list sorted in descending order by attribute price with quantity attribute less than or equal 25

○ d. the first 5 documents of a list sorted in descending order by attribute price with quantity attribute greater than or equal 25

○ e. all the documents of a list sorted in descending order by attribute price with quantity attribute less than or equal 25

Your answer is correct.

The correct answer is: the first 5 documents of a list sorted in descending order by attribute price with quantity attribute less than or equal 25

To delete all documents with price attribute does not equal to 350 from a collection named 'warehouse', we should use:

Select one:

○ a.
```
db.collection().deleteMany({price: { $ne:  350}}, function (err, obj) {
  console.log(obj.result);
});
```

○ b.
```
db.collection("warehouse").deleteMany({price: { $gte:  350}}, function (err, obj) {
  console.log(obj.result);
});
```

○ c.
```
db.collection("warehouse").deleteOne({price: { $ne:  350}}, function (err, obj) {
  console.log(obj.result);
});
```

○ d.
```
db.collection("warehouse").deleteAll({price: { $ne:  350}}, function (err, obj) {
  console.log(obj.result);
});
```

◉ e.
```
db.collection("warehouse").deleteMany({price: { $ne:  350}}, function (err, obj) {
  console.log(obj.result);
});
```  ✔

Your answer is correct.

The correct answer is:
```
db.collection("warehouse").deleteMany({price: { $ne:  350}}, function (err, obj) {
  console.log(obj.result);
});
```

Assume you have this fragment of code:

```
db.collection("warehouse").updateMany({ itemName: /x$/ }, { $inc: { quantity: 2 } }, { upsert: false}, function
(err, result) {
});
```

What does this code do?

Select one:

○ a. increments the quantity by 2 for all documents with field itemName ends by letter 'x' in a collection named 'warehouse'

◉ b. increments the quantity by 2 for all documents with field itemName starts by letter 'x' in a collection named 'warehouse' ✔

    ○ c.   sets the itemName field to '/x$/' for all documents with quantity field equal to 2 in a collection named 'warehouse'

    ○ d.   increments the quantity by 2 for all documents with field itemName ends by letter 'x' in a collection named 'warehouse' OR insert a new document if there is no match.

Your answer is correct.

The correct answers are: increments the quantity by 2 for all documents with field itemName ends by letter 'x' in a collection named 'warehouse', increments the quantity by 2 for all documents with field itemName starts by letter 'x' in a collection named 'warehouse'

With respect to MongoDB Compass Community software, which of the following is false?

Select one:

    ○ a.   Visually explore your data

    ○ b.   Interact with your data with full CRUD functionality

○ c.   it works as a driver for Node.js application to connect to MongoDB server   ✔

○ d.   Available on Windows, Mac and Linux

○ e.   View and optimize your query performance

Your answer is correct.

The correct answer is: it works as a driver for Node.js application to connect to MongoDB server

---

```
MongoClient.connect("http://localhost:8080/", {useNewUrlParser: true}, function (err, client) {
  if (err) {
      console.log('Err ', err);
  } else {
      console.log("Connected successfully to server");
      db = client.db('warehouse');
    }
});
```

With respect to the above code, which of the following is true?

Select one:

○ a.   The code has no issue

○ b.   The code has an issue: the URL protocol should be 'https:' instead of 'http:' and the port number should be 27017

○ c.   The code has an issue: the URL protocol should be 'mongodb:' instead of 'http:' and the port number should be 27017

○ d.   The code has an issue: the port number should be 27017

○ e.   The code has an issue: the URL protocol should be 'mongodb:' instead of 'http:'   ✔

Your answer is correct.

The correct answer is: The code has an issue: the URL protocol should be 'mongodb:' instead of 'http:'

◀ Week 4 Pre-Reading Quiz

Jump to...   ◆

Week 6 Pre-Reading Quiz ▶