

Text Areas : PlainText

Possible Answers :

7.5

AppDev 2

Number of Questions : 17

Section Marks : 50

Question Number : 142 Question Type : MCQ

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "DIPLOMA LEVEL: MODERN APPLICATION DEVELOPMENT 2"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?
CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS
REGISTERED BY YOU)

Options :

A. ✓ YES

B. ✗ NO

Question Number : 143 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following JavaScript program, and predict the output if executed.

```
new Promise((res, rej) => {
  function check(a, b){
    if (a === b){
      res("Matched");
    }
    else if (a.toString() === b){
      rej("Partial Match");
    }
    else{
      rej("Unmatched");
    }
  }

  check(9, "9");
}).then(fail => console.log("Fail Summary:", fail),
pass => console.log("Pass Summary:", pass));
```

Options :

- A. ✖ Pass Summary: Matched
- B. ✔ Pass Summary: Partial Match
- C. ✖ Fail Summary: Partial Match
- D. ✖ Fail Summary: Matched

Question Number : 144 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Which of the following statements is false, regarding REST and GraphQL?

Options :

- A. ✖ GraphQL helps to fetch exactly the same data which is needed, and avoids over fetching as well as under fetching.
- B. ✖ In general, a GraphQL response always returns 200 status code, with the "error" field containing the errors (if any).
- C. ✖ A REST API provides multiple endpoints to access multiple resources.
- D. ✔ All browsers follow cross-origin resource policy by default.

Question Number : 145 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following Vue application with markup index.html, and javascript file app.js.

index.html:

```
<div id="app">
  <router-link to="/"> Home </router-link>
  <router-link to="/set"> Change Player </router-link>
  <router-view style="padding-top:30px"></router-view>
  <script src="app.js"></script>
</div>
```

app.js:

```
const setName = {
  template:
    "<div ><button @click='setNameMethod'>set name</button></div>",
  methods: {
    setNameMethod() {
      const user = localStorage.getItem('name')
      user
        ? localStorage.removeItem('name')
        : localStorage.setItem('name', 'Rohit Sharma')
    },
  },
}

const getName = {
  template: '<div>Hello {{userName}}</div>',
  computed: {
    userName() {
      uname = localStorage.getItem('name')
      return uname ? uname : 'Virat Kohli'
    },
  },
}

const routes = [
  { path: '/', component: getName },
  { path: '/set', component: setName },
]

const router = new VueRouter({
  routes,
  base: '/',
})

const app = new Vue({
  el: '#app',
  router,
})
```

Suppose the application is running on "<http://localhost:8080>". What will be rendered inside the router-view if you visit the application for the first time?

Options :

- A. ✓ Hello Virat Kohli
- B. ✗ Hello Rohit Sharma
- C. ✗ Hello
- D. ✗ None of these

Question Number : 146 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following Vue application with markup index.html, and javascript file app.js.

index.html:

```
<div id="app">
  <router-link to="/"> Home </router-link>
  <router-link to="/user/1/home"> User </router-link>
  <router-view style="padding-top: 30px"></router-view>
  <script src="app.js"></script>
</div>
```

app.js:

```
const home = {
  template: '<div> Welcome to the home page</div>',
}

const user = {
  template: `<div><div>User ID: {{this.$route.params.id}}</div>
    <router-view></router-view></div>`,
}

const profile = {
  template: '<div> Name: Rohit Sharma </div>',
}

const error = {
  template: '<div> 404: Profile not found </div>',
}

const routes = [
  { path: '/', component: home },
  {
    path: '/user/:id',
    component: user,
    children: [
      { path: '', component: profile },
      { path: 'profile', component: error },
      { path: '*', component: profile },
    ],
  },
]

const router = new VueRouter({
  routes,
  base: '/',
})

const app = new Vue({
  el: '#app',
  router,
})
```

What will be rendered inside the router-view for 'User' page?

Options :

A. ✖ User ID: 1

User ID: 1

B. ✓ Name: Rohit Sharma

User ID: 1

C. ✗ 404: Profile not found

D. ✗ None of these

Question Number : 147 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following Vue application with markup index.html, and javascript file app.js.

index.html:

```
<div id="app">
  <input
    type="text"
    placeholder="search for fruits"
    v-model="filter_text"
  />
  <ul v-for="fruit in fruits_temp">
    <li>{{fruit}}</li>
  </ul>
</div>
```

app.js:

```
const fruits = ['Apple', 'abiu', 'Acrola', 'ackee']

const app = new Vue({
  el: '#app',
  data: {
    fruits: ['Apple', 'abiu', 'Acrola', 'ackee'],
    fruits_temp: [...fruits],
    filter_text: '',
  },
  watch: {
    filter_text() {
      this.fruits_temp = this.fruits.filter((fruit) => {
        return this.filter_text
          ? fruit.startsWith(this.filter_text == 'A' ? 'a' : 'A')
          : this.fruits
      })
    },
  },
})
```

Suppose the application is running on "<http://localhost:8080>". What will be rendered if the user types 'A' (uppercase) in the input box?

Options :

abiu

A. ✓ ackee

Apple

B. ✗ Acrola

- Apple
- abiu
- Acrola
- C. ✖ ackee
- D. ✖ None of these

Question Number : 148 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following Vue application. What will be rendered if the user types 'a' (lowercase) in the input box?

index.html:

```
<div id="app">
  <input
    type="text"
    placeholder="search for fruits"
    v-model="filter_text"
  />
  <ul v-for="fruit in fruits_temp">
    <li>{{fruit}}</li>
  </ul>
</div>
```

app.js:

```
const fruits = ['Apple', 'abiu', 'Acrola', 'ackee']

const app = new Vue({
  el: '#app',
  data: {
    fruits: ['Apple', 'abiu', 'Acrola', 'ackee'],
    fruits_temp: [...fruits],
    filter_text: '',
  },
  watch: {
    filter_text() {
      this.fruits_temp = this.fruits.filter((fruit) => {
        return this.filter_text
          ? fruit.startsWith(this.filter_text == 'A' ? 'a' : 'A')
          : this.fruits
      })
    },
  },
})
```

Options :

abiu
A. ✖ ackee

Apple
B. ✔ Acrola

- Apple
- abiu
- Acrola
- C. ✖ ackee
- D. ✖ None of these

Question Number : 149 Question Type : MCQ

Correct Marks : 3

Question Label : Multiple Choice Question

Which of the following shows the correct output after 20 seconds, if the below program is executed?

```
function resolver1() {
  return new Promise(resolve => {
    setTimeout(function() {
      resolve("mad1")
      console.log("Resolver1 promise is done")
    }, 2000)
  })
}

function resolver2() {
  return new Promise(resolve => {
    setTimeout(function() {
      resolve("mad2")
      console.log("Resolver2 promise is done")
    }, 1000)
  })
}

async function Start() {
  const p1 = resolver1();
  const p2 = resolver2();
  console.log(await p1);
  console.log(await p2);
}
Start();
```

Options :

Resolver2 promise is done
Resolver1 promise is done
mad1
mad2

A. ✓

Resolver1 promise is done
Resolver2 promise is done
mad1

B. ✗ mad2

Resolver2 promise is done
Resolver1 promise is done
mad2

C. ✗ mad1

Resolver1 promise is done
Resolver2 promise is done
mad2

D. ✗ mad1

Question Number : 150 Question Type : MCQ

Correct Marks : 2

Question Label : Multiple Choice Question

Which of the following function schemas simulates the callback-like behavior?

Options :

```
function doSomething(successCB, failureCB) {  
  let result = doSomeComputation();  
  if (result) successCB(success_message);  
  else failureCB(failure_message);  
}
```

A. ✗

B. ✗

```
function doSomething(successCB, failureCB) {  
  let result = doSomeComputation();  
  if (!result) failureCB(failure_message);  
  else successCB(success_message);  
}
```

Both

```
function doSomething(successCB, failureCB) {  
  let result = doSomeComputation();  
  if (result) successCB(success_message);  
  else failureCB(failure_message);  
}
```

and

```
function doSomething(successCB, failureCB) {  
  let result = doSomeComputation();  
  if (!result) failureCB(failure_message);  
  else successCB(success_message);  
}
```

C. ✓

D. ✗ None of these

Question Number : 151 Question Type : MCQ

Correct Marks : 2

Question Label : Multiple Choice Question

Which of the following URLs has the same origin as

["http://www.iitm.ac.in:8080/home.html"](http://www.iitm.ac.in:8080/home.html)?

Options :

A. ✗ <https://www.iitm.ac.in:8080/home.html>

B. ✗ <http://www.iitm.ac.in:5000/home.html>

C. ✓ <http://www.iitm.ac.in:8080/departments.html>

D. ✗ <http://onlinedegree.iitm.ac.in:8080/home.html>

Question Number : 152 Question Type : MCQ

Correct Marks : 4.5

Question Label : Multiple Choice Question

Consider the following Vue application with markup index.html, and javascript file app.js.

index.html:

```
<div id = "demo">
  <input v-model = "name" />
  <p :class = "[token ? 'Present' : 'Not Present']"> Present or Not? </p>
</div>
<script src = "app.js">
```

app.js:

```
const app = new Vue({
  el: '#demo',
  data: {
    name: "",
    token: true,
  },
  mounted () {
    this.token = localStorage.getItem("token");
    this.name = localStorage.getItem("token");
  },
  watch: {
    name (n) {

      if (n.length < 2 && true)
        this.token = false;
      else
        this.token = true;

      localStorage.setItem("token", this.token);
    }
  }
})
```

Say you open the "index.html" file in a browser, and write "itm" in the text input shown on screen, then force reload the page. What will be the value in the input text box and class applied to the paragraph element with text "Present or Not?", respectively?

Options :

- A. ✖ 'itm', 'Not Present'
- B. ✖ 'itm', 'Present'
- C. ✖ 'true', 'Not Present'
- D. ✔ 'true', 'Present'

Question Number : 153 Question Type : MCQ

Correct Marks : 4.5

Question Label : Multiple Choice Question

Consider the following Vue application. Suppose user visits the '/set' route, and clicks on 'set name' button for 6 times. And again visits the '/' route, then what will be rendered inside the router-view?

index.html:

```
<div id="app">
  <router-link to="/"> Home </router-link>
  <router-link to="/set"> Change Player </router-link>
  <router-view style="padding-top:30px"></router-view>
  <script src="app.js"></script>
</div>
```


app.js:

```
const setName = {
  template:
    "<div ><button @click='setNameMethod'>set name</button></div>",
  methods: {
    setNameMethod() {
      const user = localStorage.getItem('name')
      user
        ? localStorage.removeItem('name')
        : localStorage.setItem('name', 'Rohit Sharma')
    },
  },
}

const getName = {
  template: '<div>Hello {{userName}}</div>',
  computed: {
    userName() {
      uname = localStorage.getItem('name')
      return uname ? uname : 'Virat Kohli'
    },
  },
}

const routes = [
  { path: '/', component: getName },
  { path: '/set', component: setName },
]

const router = new VueRouter({
  routes,
  base: '/',
})

const app = new Vue({
  el: '#app',
  router,
})
```

Options :

- A. ✓ Hello Virat Kohli
- B. ✗ Hello Rohit Sharma
- C. ✗ Hello
- D. ✗ None of these

Question Number : 154 Question Type : MCQ

Correct Marks : 4.5

Question Label : Multiple Choice Question

Consider the following Vue application with markup index.html, and javascript file app.js.

index.html:

```
<div id="app">
  <router-link to="/"> Home </router-link>
  <router-link to="/user/1/profile"> User </router-link>
  <router-view style="padding-top: 30px"></router-view>
  <script src="app.js"></script>
</div>
```

app.js:

```
const home = {
  template: '<div> Welcome to the home page</div>',
}

const user = {
  template: `<div><div>User ID: {{this.$route.params.id}}</div>
  <router-view></router-view></div>`,
}

const profile = {
  template: '<div> Name: Rohit Sharma </div>',
}

const error = {
  template: '<div> 404: Profile not found </div>',
}

const routes = [
  { path: '/', component: home },
  {
    path: '/user/:id',
    component: user,
    children: [
      { path: '', component: profile },
      { path: 'profile', component: error },
      { path: '**', component: profile },
    ],
  },
]

const router = new VueRouter({
  routes,
  base: '/',
})

const app = new Vue({
  el: '#app',
  router,
})
```

Suppose the application is running on "<http://localhost:8080>". What will be rendered inside the router-view for "User" page?

Options :

A. ✖ User ID: 1

 User ID: 1
B. ✖ Name: Rohit Sharma

C. ✔

User ID: 1

404: Profile not found

D. ✖ None of these

Question Number : 155 Question Type : MCQ

Correct Marks : 4.5

Question Label : Multiple Choice Question

Fill in **code1** & **code2**, which can be used in Vuex Store to update the “toppers” state variable with the objects of those students who have scored more than 90 marks.

```

const store= new Vuex.Store({
  strict:true,
  state:{
    student_total:0,
    students:[
      {
        name : 'Ram',
        marks : 94
      },
      {
        name : 'Shyam',
        marks : 57
      },
      {
        name : 'Geeta',
        marks : 98
      }
    ],
    toppers:[]
  },
  getters:{
    studentTotal(state){
      return state.students.length
    },
    toppersTotal(state){
      return state.toppers.length
    }
  },
  code1:{
    topStudents(state){
      code2
    },
  }
})

```

Options :

code1: mutations

code2: students.forEach(student=>{
 if(student.marks>=90){
 toppers.push(student) } })

A. ✖

B. ✖

code1: actions

```
code2: state.students.forEach(student=>{  
    if(student.marks>=90){  
        state.toppers.push(student) } })
```

code1: actions

```
code2: context.students.forEach(student=>{  
    if(student.marks>=90){  
        context.toppers.push(student) } })
```

C. ✖

code1: mutations

```
code2: state.students.forEach(student=>{  
    if(student.marks>=90){  
        state.toppers.push(student) } })
```

D. ✔

Question Number : 156 Question Type : MSQ

Correct Marks : 3

Question Label : Multiple Select Question

Which of the following statement(s) is/are true regarding JavaScript language?

```
function rule (num, obj) {  
    // some code here  
}  
rule(num, obj);
```

Options :

- A. ✔ In the above function call, the variable "num" is passed by value, assuming it to be holding a string literal.
- B. ✔ In the above function call, the variable "obj" is passed by reference, assuming it to be holding an object.
- C. ✖ A variable declared using the keyword "let" will have the value "null", before it is initialized.
- D. ✖ All of these

Question Number : 157 Question Type : MSQ

Correct Marks : 2

Question Label : Multiple Select Question

Which of the following statement(s) is/are true regarding fetch API?

Options :

- A. ✖ A fetch API can only be used to make HTTP GET and POST requests.
- B. ✖ A fetch API call always responds with "OK" status set to true.
- C. ✔ The "credentials : 'omit'" ensures that the browser never sends the cookies with the request.
- D. ✔ A fetch API call can also have a body associated with the request.

Question Number : 158 Question Type : MSQ

Correct Marks : 2

Question Label : Multiple Select Question

Consider the following API (app.py).

```

from flask import Flask, jsonify
from flask_cors import cross_origin
app = Flask(__name__)

# home resource
@app.route('/')
@cross_origin()
def home():
    return jsonify({"resource": "home"})

# profile resource
@app.route('/profile')
def profile():
    return jsonify({"resource": "profile"})

if __name__ == "__main__":
    app.run(debug=True)

```

Suppose the API is running on 'http://localhost:5000' 'Which of the following is correct regarding the fetch call to the routes '/' and '/profile' (assuming the client to be Chrome browser)?

Options :

- A. ✖ A fetch call to get the home resource from origin "<http://localhost:8080>" will result in CORS error.
- B. ✔ A fetch call to get the home resource from origin "<http://localhost:8080>" will be successful.
- C. ✔ A fetch call to get the profile resource from origin "<http://localhost:8080>" will result in CORS error.
- D. ✖ A fetch call to get the profile resource from origin "<http://localhost:8080>" will be successful.

Java

Number of Questions :	16
Section Marks :	50