FIT2095 e-Business software technologies - S2 2021

Started on	Monday, 11 October 2021, 11:36 PM
State	Finished
Completed on	Tuesday, 12 October 2021, 11:58 AM
Time taken	12 hours 21 mins
Grade	**10.00** out of 10.00 (**100**%)

Print friendly format

Question **1**

Complete

Mark 10.00 out of 10.00

Question 1

```
io.sockets.emit("anotherEventName", data);
```

The param1 is the name of the event and it is a string called anotherEventName

The param2 is serialisable data structure and due to it is io.sockets.emit it will send to all the clients.

Comment:

Question **2**

Complete

Not graded

Question 2

```
io.socket.on("event1", function(data){
    console.log("got data from event 1 ");
}
```

The param1 is the name of the event and it is a string called event1

The param2 is function and it is a callback function, it will be called when the server broadcasts a notification to this socket. io.socket.on is listening to the socket events with the event name, trigger while be called

Question 3

The JSON file include privacy key and it is the password to the server connection, it likes a key to assess the specified server in google cloud

We define and export the path of the key then while we use application, the google can recognize our computer

```
export GOOGLE_APPLICATION_CREDENTIALS="/home/user/Downloads/service-account-file.json"
```

while we import google client library like : const textToSpeech = require("@google-cloud/text-to-speech");
the client library will look for credentials in the environment

Question 4

```
let express = require("express");
```

```
let path = require("path");
```

```
let app = express();
```

```
let server = require("http").Server(app);
```

```
let io = require("socket.io")(server);
```

```
let port = 9595;
```

```
app.use("/", express.static(path.join(__dirname, "dist/workshop3")));
```

```
///
```

```
io.on("connection", function (socket){
```

```
  socket.broadcast.emit("newSocketID", socket.id);
```

```
  socket.on ("values", function(data){
```

```
        let result=data.no1-data.no2;

        let Obj={

            no1: data.no1,

            no2: data.no2,

            newResult: result

            }

        io.sockets.emit("newResult", Obj);

    });

});

///

server.listen(port, () => {

    console.log("Listening on port " + port);

});
```

Question 5

Not answered

Not graded

Question 5

Question 6

Not answered

Not graded

Question 6

Question 7

Not answered

Not graded

Question 7

Question 8

Not answered

Not graded

Question 8

Jump to...