

CHALMERS

EXAMINATION / TENTAMEN

Course code/kurskod		Course name/kursnamn		
DET 342		web development		
Anonymous code Anonym kod		Examination date Tentamensdatum	Number of pages Antal blad	Grade Betyg
DET 342 - 0002 - 0LF		2025-08-20	9	

* I confirm that I've no mobile or other similar electronic equipment available during the examination.
Jag intygar att jag inte har mobiltelefon eller annan liknande elektronisk utrustning tillgänglig under
examinationen.

Solved task Behandlade uppgifter No/nr	Points per task Poäng på uppgiften	Observe: Areas with bold contour are to completed by the teacher. Anmärkning: Rutor inom bred kontur ifylles av lärare.
1	X 75,5	
2	X 75,5	
3	X 6	
4	X 23	
5	X 9,5	
6	X 5	
7	0	
8	X 4	
9	X 4	
10		
11		
12		
13		
14		
15		
16		
17		
Bonus poäng		

Question 1

```

1-1 app.put1('/routes/:id', function2(req, res) {
  const id = req.params.id0,5; Number0,5(req.params.id);
  const route = routes.find1(function(r) { r.id0,5 === id });

```

```

  Object.assign0,5(id, req.body);
  res.status1(200).json1(route);

```

```

});

```

```

2 app.post1('/routes/:id/stop', function7,5(req, res) {

```

```

  const id = req.params0,5.id;
  const route = routes.find1(function(r) { r.id0,5 === id });

```

```

  const newStop = req.body.stop;

```

```

  routes.steps[id].push1(newStop);

```

```

  res.status1(201).json0,5({stop: newStop});

```

```

});

```

15,5

Question 2

2.1 Post /routes → creates new routes

→ 201 created status code will be received

→ then a new routes will be created

Put /routes → modify/update all routes

→ 200 OK status code will be received

so that we know that all routes have been modified.

2.2 GET /route? delete (:route) → Get is normally a secure

method, so it should not modify the server, is also idempotent so that

every request with get ~~see~~ should result in the same response

In this case we getting and deleting a specific route, according to this method. If we intend to delete a specific route we should use DELETE /routes/:route, so that we respect the RESTful API design by using the correct method.

GET /route /:route/ getAll Stops

Here the issue is with the getAllStops, that is a function i would assume and will not allowed the method to get the request of the client, 400 code could be shown.

Could use GET /routes/:route/stops, for an accept request, more readable, and so that we get all stops from a specific route.

DET 342-0002 - DCF

Question 2

2.4.) GET /route / 64026 / stops / stop? platform = "B" (2,5)

This method should return all stops ~~that~~ in platform B in route id 64026

2.5.) CRUD | ~~function~~ Method

Create → POST¹

Read → GET¹

Update → PUT/PATCH¹

Delete → DELETE¹

Question 3

14 - Black¹15 - Orange¹17 - ~~Blue~~ Black ~~Blue~~ Black¹

18 - Purple

19 - yellow¹20 - ~~blue~~ Black ~~Blue~~ Black¹

21 - purple

22 - red¹23 - blue¹24 - green¹

(6)

Question 4

6- <input type="text" v-model="newStop"/>
 7- <input type="text" v-model="newDuration"/>
 8- <button @click="addToStop"> Add intermediate stop </button>

11- <li v-for="stop in stops">
 <p> {{ stop.newStop }}
 {{ stop.newDuration }}
 <button @click="increase(newDuration)"> + </button>
 <button @click="decrease(newDuration)"> - </button>
 </p>

26- addToStop: function() {
 stops.push({~~newStop: this.newStop~~, ~~newDuration: this.newDuration~~
 newDuration: this.newDuration});
 this.newStop = "";
 this.newDuration = 10;
 }

increase: function (stops [newDuration]) {
 stops = indexOf(stops [newDuration]);
 stop [newDuration] + 1;
 }

decrease: function (stops [newDuration]) {
 if (! stops [newDuration] <= 1) { stops = indexOf(stops [newDuration]);
 stop [newDuration] - 1;
 }

Question 5

Div 1	Div 1	Div 1	Div 1	Div 2	Div 2	Div 3						
Div 4	Div 4	Div 4	Div 4	Div 5	Div 5	Div 5	Div 5	Div 6	Div 6	Div 6	Div 6	

| | → border

9,5

large browser window 1300px

Div 1	Div 1	Div 1	Div 1	Div 1	Div 1	Div 1	Div 1	Div 2	Div 2	Div 2		
Div 3	Div 3											
Div 4	Div 4	Div 4	Div 4	Div 4	Div 5	Div 5						
Div 6	Div 6	Div 6	Div 6	Div 6								

| | → border

small browser window 530px

9,5

Question 6

A → http method¹

B → path¹

5

C → http version¹

D → host, web page name¹

E → (optional) body, accept¹, cookie etc

Question 8

Client side → is the front-end part, where we take care of ~~the~~ what the user sees, ~~the~~ UI, HTML, CSS JavaScript are the most ² ~~caroused~~ used in front end development. Execution happens in the browser

→ if

4

server side → is the back-end part, here we take care of the data the client stores or request from server, and we send ² it in a response. Here we use databases as MongoDB, JSON to format the, ~~AJAX~~ for ~~Apross~~. Here ~~reaction~~ happens in the server.

Depending on how do you design your page as an SPA or a more traditional web site ~~meaning that~~ In a SPA single-page-application, thanks to AJAX, an asynchronous request request is made ~~so that~~ to server, then a response is send, then the part that is selected is the only part that is going to update. On the other hand with a more traditional design the reload is made on the entire page making it ~~more~~ slower.

6

Question 9

```
constructor : function (namevar, stopvar) {
```

```
  const stop = new Stop();
```

```
  this.name = name1,5;
```

```
  this.stop = stop1,5;
```

```
  name = " ";
```

```
  stop = " ";
```

```
}
```

④

4