1. **In your own words, explain what a Javascript module is. What problem is solved by modules? How are modules implemented in Javascript ES6?**
2. **What a Javascript module is.**

I think a module is a self-contained unit of code and it has distinct functionality that helps with the code organization of the app.

1. **What problem is solved by modules?**
2. Dependencies( lessen the dependencies on parts of the code base as much as possible and increase maintainability)
3. the risk of namespace pollution
4. use code that is already written in one place for some other purpose in another place and realize reusability.

**3) How are modules implemented in Javascript ES6?**

**ES6** introduced a model a new language keywords to support modules. The new module mechanism allows developers to create modules containing specific functions classes or objects as one or more symbols.

(1)**Exporting** ( the export keyword allows you to export symbols from your module to be available to other module consumers )

Eg:

//mathlib.js

export const sqrt = Math.sqrt;

export square = x { return x\*x ; }

export diag = ( x , y )

{return sqrt ( square(x) + square(y) );}

(2)**importing** (the import statement allows you to import all or just specific symbols for a moudle)

Eg:

//app.js

import { sqrt , diag } from mathlib;

1. **Explain the trade-off between freedom and convention in application organization? Describe the advantages and disadvantages of each approach and give your opinion about which approach your favour.**

**1) the trade-off between freedom and convention**

Developers agree on a general application structure such as where to put files and how to name folders and how to build components and how to build and share these with others by giving up some freedom.

**2) freedom ‘s advantages and disadvantages**

**advantages：**developers have complete freedom and flexibility to do

**disadvantages:** if every developer organize their react native apps differently which will lead to higher levels of entropy and complexity for the community of individuals**.**

**(** If every developer were to organize their react native apps differently no two developers could easily share the burden of supporting each other's work. At a minimum each developer would have to separately document the way they have structured their applications and be available to answer questions lest anything was not completely clear**)**

**3)convention‘s advantages and disadvantages**

**advantages:**  make it easy to change some items such as global settings, colour’s routes etc. without worrying about breaking something else elsewhere in the code.

**disadvantages:** Not very flexible and it maybe limits more possibilities

**4) my favour**

My favour approach is convention. If we all follows convention in a team and it can improve the efficiency of cooperation. And conventions help others understand our own codes and help us to solve some problems.

1. **Define and compare the concepts of:**

**(a)” write once, run everywhere”**

**(b)” learn once, write everywhere”**

**(c) what is React Native considered to the second type of development system?**

**1) Define and compare the concepts of “ write once, run everywhere” and ” learn once, write everywhere”**

**Define:**

The “write once, run everywhere” means that this app can develop all devices regardless of the hardware or operating system. The “learn once, write everywhere” means that we can create truly portable apps from a single codebase.

**Compare:**

The “write once, run everywhere” expresses support for all kinds of hardware and operating system. The “learn once, write everywhere” shows that we can use same syntax in different development environment.

**2) what is React Native considered to the second type of development system?**

React native is based on react JS framework and React JS targets a standards based browser architecture. React native is providing a web like abstraction on top of multiple platforms.

**d. Why do you think that React Native needs to provide support for platform specific components in its standard component library?**

(1) Some components are abstractions of underlying user interface functionality which is either visually or semantically different between IOS and Android.

(2) Their specific components are available for own hardware. (the Android way gives access to features which are only relevant to Android such as the permissions model.)

**e. Describe the event system in React Native and explain how to handle events in a React Native App?**

**1) Describe the event system in React Native**

The term event is used to describe something which happens asynchronously to the program execution asynchronous means that something can happen at any time during the program execution and is independent of that execution flow. (touchable presses button) And most events are delivered to javascript through callback function functions.

**2) how to handle events in a React Native App**

**(1) register an Event Handler**

In react native the component event handler is attached to an attribute of the component.

Eg: a stateful component class

export default class EventExample extends Component{

onPressButton(){

Alert.alert(‘You tapped the button!’)}

reder(){

return (

<Veiw style ={ styles.container}>

<Button onPress ={ this. onPressButton}/>

</Veiw>

)}

}

We use the “this” keyword to access the method name on the class. When the button is pressed by a user the un-press button handler will be called.

**(2) Delegating Event Handling**

Then we must consider that stateless function components perform event handling. In small apps and cases, we warp the stateless component in a stateful container component.

**(3) Handle the Event**

 Internally we can consider that the user interface logic is implemented as an event loop which waits synchronously for messages or events to arrive and then be processed in turn to completion.