📖 Set 2: React Functions & Components

📖 1. JavaScript Functions

✅ Important Notes

> ◈ A function is a reusable block of code used to perform a single task. ◈ Functions can take parameters and return a value using the `return` keyword. ◈ Declared using `function` keyword or arrow syntax.

📝 MCQs

Q1. How do you declare a named function? a) `function greet() {}` b) `const greet = () ⇒ {}` c) `def greet()` d) `greet() := {}`

Q2. What is the purpose of parameters in a function? a) To style output b) To loop data c) To pass inputs d) To store return values

Q3. What will this return?

```
function sum(a, b) {
  return a + b;
}
console.log(sum(2, 3));
```

a) 5 b) 23 c) undefined d) Error

Q4. What happens if a function doesn't return anything? a) Returns `null` b) Returns `undefined` c) Returns `0` d) Crashes

Q5. What is the default return value of a function without return? a) `"null"` b) `"0"` c) `undefined` d) `false`

💡 Challenge Style

> ❓ Write a function `multiplyByTwo` that accepts a number and returns it multiplied by 2.
>
> ❓ Write a function `isEven(num)` that returns `"Even"` if number is even, `"Odd"` otherwise.

📑 2. Return Statement

✅ Important Notes

> ◈ `return` exits a function and sends a value back. ◈ Code after `return` is never executed. ◈ Used in both `function` and `arrow functions`.

📝 MCQs

Q1. Which line is unreachable?

```
function test() {
    return "Done";
    console.log("Hello");
}
```

a) `"Done"` b) `console.log("Hello");` c) `function test()` d) None

Q2. What happens if `return` is on one line and value on next?

```
function a() {
    return
```

```
    5;
  }
```

a) Returns 5 b) Returns undefined c) Error d) Crashes

Q3. How can you return a string? a) `return string;` b) `return "text";` c) `return text();` d) `send "text"`

Q4. Can you return multiple values directly? a) Yes b) No c) Only strings d) Using array/object

Q5. What is output?

```
function check() {
    return;
}
console.log(check());
```

a) `undefined` b) `null` c) `false` d) Error

💡 Challenge Style

> 🧩 Create a function `fullName(first, last)` that returns full name as a string.

> 🧩 Create a function that returns the square of a number.

🏹 3. Arrow Functions

✅ Important Notes

> ◈ Introduced in ES6. Shorter syntax for functions. ◈ Inherits `this` from outer context (important in React). ◈ For single expressions: parentheses and `return` are optional.

```
const add = (a, b) ⇒ a + b;
```

📝 MCQs

Q1. Which is a valid arrow function? a) `let x ⇒ () {}` b) `const x = () ⇒ {}` c) `⇒ const x()` d) `const ⇒ () x`

Q2. Arrow function with 1 param doesn't need: a) Braces b) `return` c) Parentheses d) All of the above

Q3. What does this return?

```
const msg = () ⇒ "Hi";
console.log(msg());
```

a) `"Hi"` b) `undefined` c) `msg` d) `null`

Q4. Output of this arrow function?

```
const square = x ⇒ x * x;
console.log(square(3));
```

a) 6 b) 9 c) undefined d) Error

Q5. Which arrow function syntax is incorrect? a) `x ⇒ x * 2` b) `(x, y) ⇒ x + y` c) `() ⇒ { return 5; }` d) `() ⇒ return 5;`

💡 Challenge Style

> 🔖 Convert this to arrow function:

```
function sayHello(name) {
    return "Hi " + name;
}
```

> 🔖 Write an arrow function to check if number is positive.

📦 4. Modules – `export`, `export default`

✓ Important Notes

> ◈ JavaScript modules allow splitting code into files. ◈ Use `export` for named exports. ◈ Use `export default` for one main export per file. ◈ `import` is used to bring code from other files.

📝 MCQs

Q1. Which is a named export? a) `export default` b) `export function greet() {}` c) `default greet()` d) `module.export = greet`

Q2. How to import a default export? a) `import { greet } from './file.js'` b) `import greet from './file.js'` c) `require greet` d) `use greet`

Q3. Can you have multiple `export` in one file? a) Yes b) No c) Only in Node d) Only one

Q4. What happens if both `export` and `export default` exist? a) Error b) Only default works c) Both can be used d) Ignores default

Q5. What does this do?

```
export default function main() {}
```

a) Exports multiple functions b) Exports main as default c) Imports another file d) Causes error

💡 Challenge Style

> ❔ Create a file `math.js` that exports `add`, `subtract` using named exports.
>
> ❔ In another file, import and use `add`.

🧩 5. Components (React)

✔️ Important Notes

> ◈ A component is a reusable UI block. ◈ Must start with uppercase. ◈ Returns JSX (HTML-like syntax). ◈ Can be written as:

```
function Hello() {
  return <h1>Hello</h1>;
}
```

or

```
const Hello = () ⟹ <h1>Hello</h1>;
```

📝 MCQs

Q1. What is required for a React component to render JSX? a) It must return a string b) It must return JSX c) It must use `console.log` d) It must end with `.jsx`

Q2. Which component is correctly named? a) `function header()` b) `function Header()` c) `component header()` d) `Header = () ⟹ {}`

Q3. Can a React component return multiple tags? a) No b) Yes, wrapped in one element c) Only in fragments d) Only if exported

Q4. Which of the following is a valid functional component? a) `function App()
{ return <h1>App</h1>; }` b) `component App() ⇒ <h1>App</h1>` c)
`return App() ⇒ <h1>App</h1>` d) `App = function: JSX`

Q5. How are components reused? a) Calling as functions b) JSX tags like `<Header />`
c) Using `require` d) Global scope

💡 Challenge Style

> ❓ Create a functional component `Welcome` that returns `<h1>Welcome to
> MyApp</h1>`
>
> ❓ Create a component `User` that takes `props.name` and shows:

```
<h2>Hello, {props.name}</h2>
```