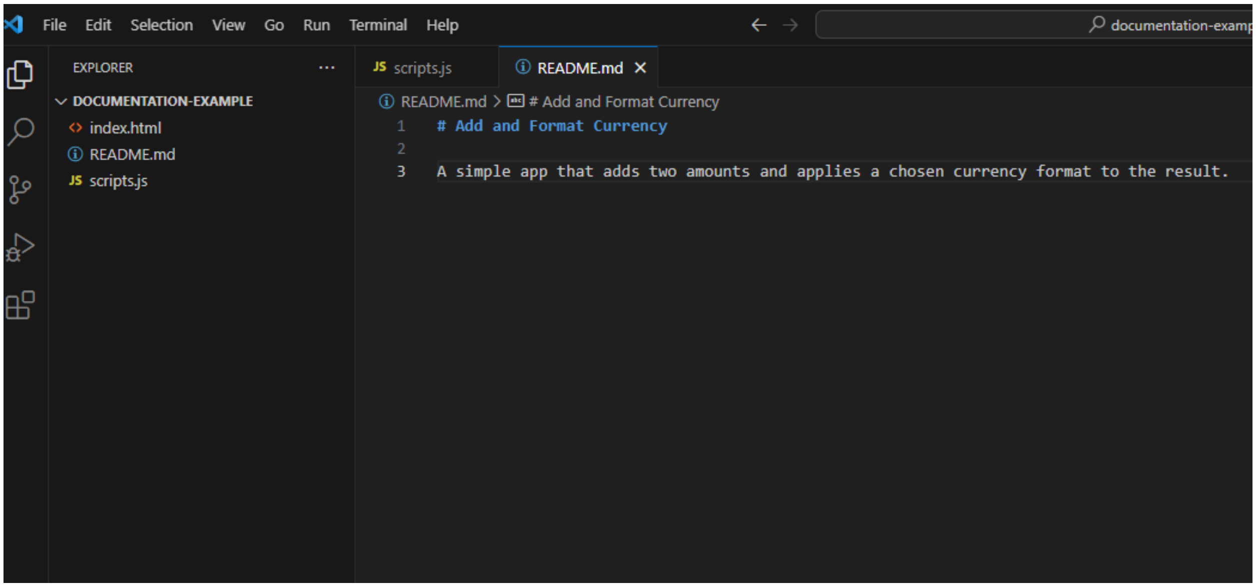
DWA_03.4 Knowledge Check_DWA3.1

---

1. Please show how you applied a Markdown File to a piece of your code.

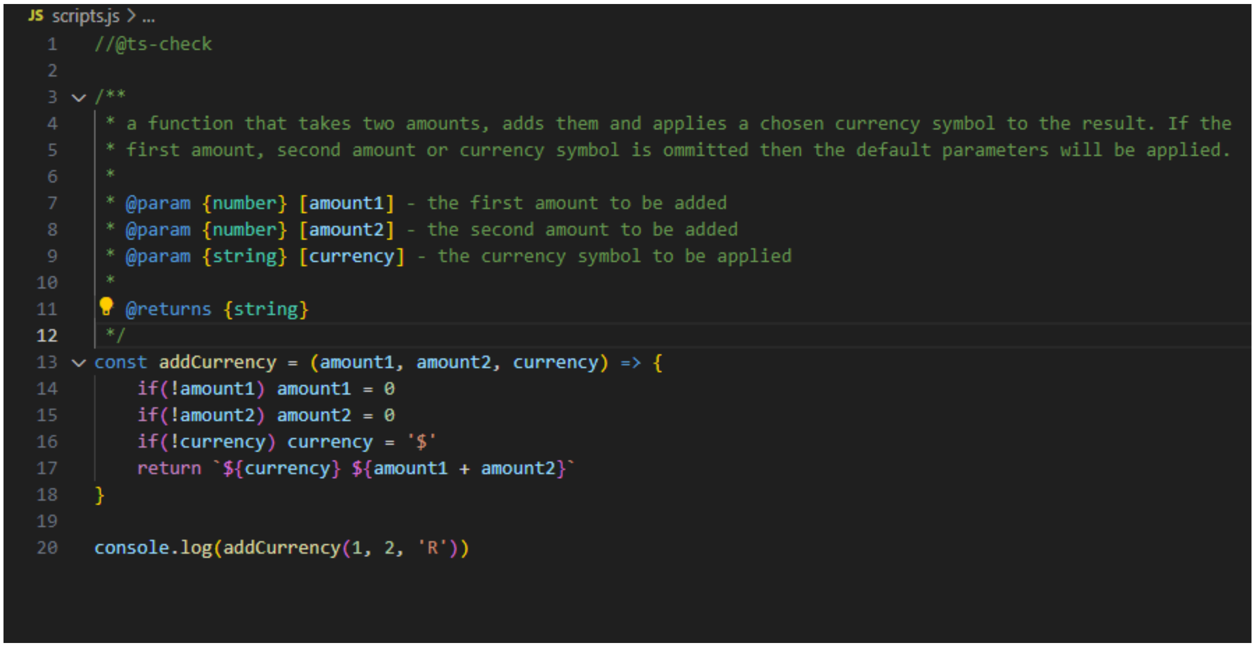


```markdown
# Add and Format Currency

A simple app that adds two amounts and applies a chosen currency format to the result.
```

A README Markdown File was added to provide context on what the app does.

---

2. Please show how you applied JSDoc Comments to a piece of your code.

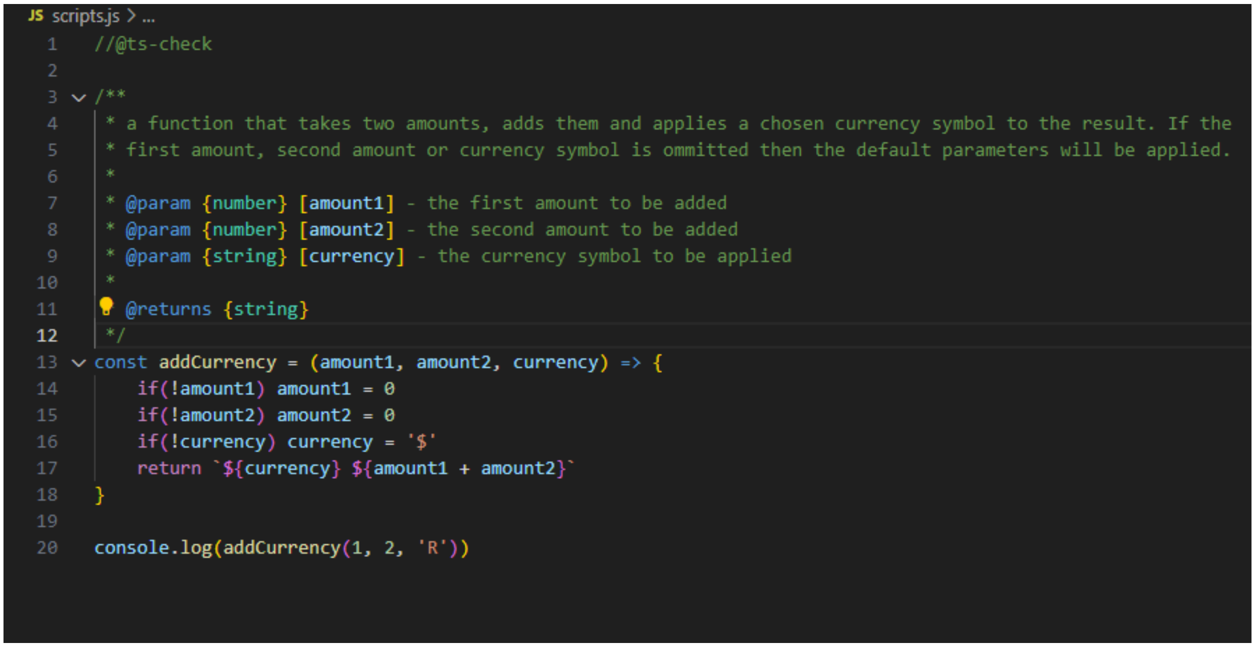


```javascript
//@ts-check

/**
 * a function that takes two amounts, adds them and applies a chosen currency symbol to the result. If the
 * first amount, second amount or currency symbol is ommitted then the default parameters will be applied.
 *
 * @param {number} [amount1] - the first amount to be added
 * @param {number} [amount2] - the second amount to be added
 * @param {string} [currency] - the currency symbol to be applied
 *
 * @returns {string}
 */
const addCurrency = (amount1, amount2, currency) => {
    if(!amount1) amount1 = 0
    if(!amount2) amount2 = 0
    if(!currency) currency = '$'
    return `${currency} ${amount1 + amount2}`
}

console.log(addCurrency(1, 2, 'R'))
```

A short description of the addCurrency function is provided along with the three parameter types and description for each.

---

3. Please show how you applied the @ts-check annotation to a piece of your code.

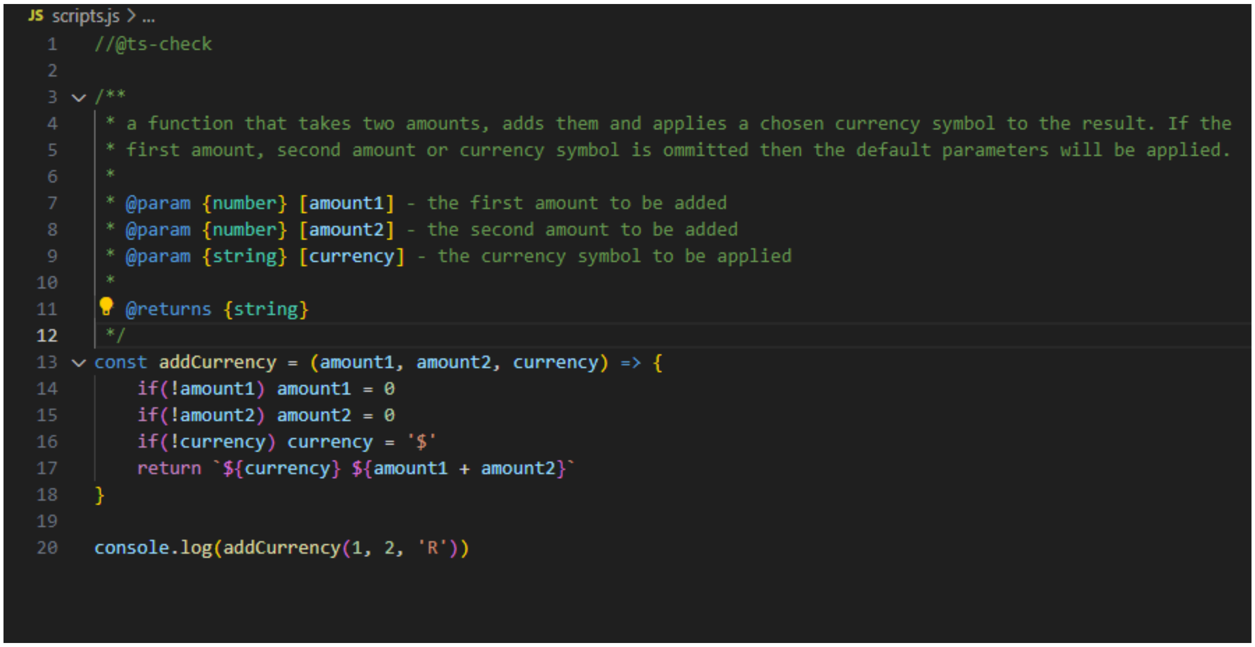


```
//@ts-check

/**
 * a function that takes two amounts, adds them and applies a chosen currency symbol to the result. If the
 * first amount, second amount or currency symbol is ommitted then the default parameters will be applied.
 *
 * @param {number} [amount1] - the first amount to be added
 * @param {number} [amount2] - the second amount to be added
 * @param {string} [currency] - the currency symbol to be applied
 *
 * @returns {string}
 */
const addCurrency = (amount1, amount2, currency) => {
    if(!amount1) amount1 = 0
    if(!amount2) amount2 = 0
    if(!currency) currency = '$'
    return `${currency} ${amount1 + amount2}`
}

console.log(addCurrency(1, 2, 'R'))
```

“//@ts-check” is added to the first line of the scripts.js file to enable type checking for every line of code within the file.

---

4. As a BONUS, please show how you applied any other concept covered in the 'Documentation' module.



```
//@ts-check

/**
 * a function that takes two amounts, adds them and applies a chosen currency symbol to the result. If the
 * first amount, second amount or currency symbol is ommitted then the default parameters will be applied.
 *
 * @param {number} [amount1] - the first amount to be added
 * @param {number} [amount2] - the second amount to be added
 * @param {string} [currency] - the currency symbol to be applied
 *
   @returns {string}
 */
const addCurrency = (amount1, amount2, currency) => {
    if(!amount1) amount1 = 0
    if(!amount2) amount2 = 0
    if(!currency) currency = '$'
    return `${currency} ${amount1 + amount2}`
}

console.log(addCurrency(1, 2, 'R'))
```

The square brackets around each parameter name within the JSDoc Comments allows each parameter to be optional when calling the addCurrency function.