**Exercise 1**

1. **Scope and context in JS**
2. First console **this** will refer to **window** and

Second console **this** will refer to Drinks {}

1. 10

0

1. Console A 🡺 **Oops ReferenceError: guessMe2 is not defined**, it is due to let variable definition are not hoisted to top of block unlike var

Console B 🡺 **1,3**

Console C 🡺 **1,2**

1. **Arrow Function**
2. **Uncaught ReferenceError: arguments is not defined //** we cannot access arguments object inside arrow function and this one of the case when we should not use arrow function.
3. **2**
4. **Undefined, window ///**Inside arrow function **this** parent scope and it is window currently

10 {i: 10, b: ƒ, c: ƒ} /// As it is not arrow function **this** refers to the current object

1. Uncaught TypeError: Foo is not a constructor ///One other case where we should not use arrow function.Arrow function should not be used as constructor
2. Uncaught TypeError: Cannot set property 'name' of undefined //Arrow function should not be used with prototype
3. const name ="Mike";

const age = "20";

const profile=(name , age)=>({name,age})

console.log(profile(name , age)) //{name:”Mike”,age:”20”}

1. const myEle=[5,7,8];

const total=myEle.reduce((total , number) => total+number);

console.log(`Sum of the integers: ${ total }`)

1. Default Arguments
2. 120 //to take default parameter we have to pass **undefined**
3. String
4. const callbackdelay = (callback , delay=1000) => {setTimeout( () => {callback()} ,delay)}
5. let defaultValue=1;

function printComment( comment , line=defaultValue ) {

console. log( line, comment ) ;

defaultValue+=1;

}

printComment(“hello”); // 1 “hello”

printComment(“hello”); // 2 “hello”

printComment(“hello”); // 3 “hello”

1. Same as 3 iii.
2. Currying
3. 12
4. const greetings = toname => msg => fromname => `Dear ${toname}

${msg}

From,

${ fromname }`

greetings('Mike')('Wish you Happy Birthday!')('Steve');