German International University
Faculty of Informatics and Computer Science
Dr. Iman Awaad
Eng. Ahmed Sherif
Amir Haythem
Mohamed Essam
Yassein Eldamasy

Software Engineering, Winter 2025
Practice Assignment 5

Exercise 5-1

```
//main.js
function getReminder(x,y){
return x % y;
}

function isEven(x){
return getReminder(x,2) === 0;
}

function getEven(arr){
return arr.filter( num  => num % 2 === 0  );
}

function getOdd(arr){
return arr.filter( num  => num % 2 === 1  );
}
```

Assume that main.js and app.js are in the same directory

How can we import function isEven from main.js to app.js ?
How can we import function getEven and getOdd from main.js to app.js ?

Exercise 5-2

What is the output for the given code ?

```
const firstWeek = () => { console.log( "Week 1" ); }
const secondWeek = () => { console.log( "Week 2" ); }

setTimeout( firstWeek, 5000  ); // async Function
setTimeout( secondWeek, 4000  );

console.log("Week 3");
console.log("Week 4");
```

Exercise 5-3

What is the output for the given code ?

```
const getReminder = ( x,y ) => { return x % y;   }

const getTypeNumber =  (x, callback) => {

const reminder = callback(x,2);
if( reminder == 0){
console.log(`${x} is an Even Number`);
}else{
console.log(`${x} is an odd Number`);
}

}

getTypeNumber(7,getReminder);
```

Exercise 5-4

What is the output for the given code ?

```
async function callTimer(){

setTimeout( () => {
    console.log("First Timer: 7:45");
      setTimeout (  () => {
          console.log("Second Timer 7:55");
              setTimeout( () => {
              console.log("End");
                  setTimeout(  () => {
                         console.log("Third Timer 8:05");
                },3000 )
           },2000  )
       },3000 )
   },2000)
}

callTimer();
console.log("Start Timer");
```

Exercise 5-5

What is the output for the given code ?

```
const isAppointment = false;

const appointment = (appointmentFlag) => { return new Promise( (resolve , reject) =>
{

if( appointmentFlag ){
reject( new Error("cannot schedule new appointment") );
} else {
const newAppointment = { name : "office hours" , timing : "8pm"};
resolve( newAppointment );
}

})}

const addDefaultDetails = (res) => {
res.location="Zoom";
return Promise.resolve(res);
}



appointment(isAppointment).then( res => { return addDefaultDetails(res); })
.then( res =>
console.log(`Appointment ${res.name} on ${res.location} at ${res.timing}`) )
.catch(err => console.log(err.message))
```

Assume that the value of isAppointment changed to true
What will be the ouptut for the above code ?

Exercise 5-6

New Function myAppointment was added to the code in Ex 5.5
What is the output for the given code ?

```
async function myAppointment(appointmentFlag){
try{
const appointmentDetails =  await appointment(appointmentFlag);
const res =  await addDefaultDetails(appointmentDetails);
console.log( `Appointment ${res.name} on ${res.location} at ${res.timing}` );
} catch(e) {
console.log("error message: ",e.message)
}
}

const flag = true;
myAppointment( flag );
```

Assume that the value of flag changed to false
What will be the ouptut for the above code ?

Exercise 5-7

```
Given the following asynchronous function called getStudents
which takes one argument :
1. studentId : string

getStudents return a promise object with the following properties:
1. studentId : string
2. GPA: Number
3. email : string
4. name : string


function getStudents(studentId){
    let studentObj = {name : "notFound"};
    switch(studentId){
        case "1002345": studentObj = {GPA : 0.7,
        email :"mo.ihab@student.giu-uni.de", name : "Mohamed Ihab"};break;
        case "1002340": studentObj = {GPA : 0.75,
        email :"nada.yasser@student.giu-uni.de", name : "Nada Yasser"};break;
        case "1002346": studentObj = {GPA : 0.79,
        email :"mostafa.saeed@student.giu-uni.de", name : "Mostafa Saeed"};break;
    }
    return studentObj.name!="notFound"?Promise.resolve(studentObj)
    :Promise.reject(new Error(‘student id ${studentId} doesn’t exists‘));

}
```

Write a function called displayStudentInfo which takes one argument :
1. students : array of string

returns array of string where each item of the output array should adhere to the following format :
name:email:GPA:id
Implement function displayStudentInfo twice
The first implementation using async/await approach
you are not allowed to use async/await in the second implementation
Note that students is array of students id