**Exercise 2**

1. **Spread Operators**
2. const sortletters = myString => [...myString].sort().join('');

console.log( sortletters(‘zxa’));

1. const concatenateString = (seperator,...arrOfString) => arrOfString.join(seperator);

console.log(concatenateString(",", "Hello Sir", "How are you", "My name is Mike"))

//"Hello Sir,How are you,My name is Mike"

1. Same as 1 ii.
2. SyntaxError: Rest parameter must be last formal parameter
3. SyntaxError: Unexpected token ...
4. const funcReverse2= (inp) => parseInt([...inp.toString()].reverse().join(''));

console.log(funcReverse2(123456));

1. [1, 2, Array(3)] ie [1,2[3,4,5]]
2. const mergeArray = (fArr,sArr,tArr) => {

fArr.push.apply(fArr,sArr);

fArr.push.apply(fArr,tArr);

return fArr

}

mergeArray([1,2],[3,4],[5,6]);

// [1, 2, 3, 4, 5, 6]

1. Same as viii.
2. **Destructuring**
3. **2**

**5**

1. **const {a,b}=obj**
2. **const {**

**company:organisation="Sapient",**

**personal: {**

**name:firstName,**

**age:aged,**

**state:region="Andhra Pradesh",**

**spouse: {**

**name:spouseName**

**}**

**}**

**}=data;**

**console.log(comp,firstName,spouseName,region,aged)**

1. **const mapList=[];**

**for(let {name,company} of arr) {**

**mapList.push({name,company});**

**}**

**console.log(mapList);**

1. **const {name:childrenName}=data2.personal.children[0];**

**console.log(childrenName);**

1. **const myDestruct = (obj) => {**

**const {name,id}=obj;**

**console.log(`My name is ${name} and id ${id}`)**

**}**

**myDestruct({name:”naveed”,id:12345})**

1. **Object destructuring**

const myData = {

name:”Naveed”,

id:12345,

company1:”cts”,

company2:”itc”,

company3:”sapient”

}

const {name,id,…companies}=myData

console.log(comapnies)

// {company1:”cts”,company2:”itc”,company3:”sapient”}

**Array Destructuring**

const fruits=["mango","orange","jackfruit"];

const [fruit1,…restfruit]=fruits

console.log(restFruit) // ["orange", "jackfruit"]