**一、 选择题（均为单选题，每小题2分，共24题， 48分）**

1. 下面代码的输出结果正确的是（  D  ）

console.log(a);

const a = 1;

console.log(a);

A. 第1行报错

B. 第2行报错

C. 第3行报错

D.  undefined     1

2. 下面代码的输出结果正确的是（  B  ）

const obj = {

a: 1,

b: 2,

};

obj.a = obj.b;

console.log(obj.a, obj.b);

A. 第5行报错

B.  2   2

C.  1   2

D. 第6行报错

3. 下面的代码输出结果正确的是（   BA ）

function combine(split=",", ...args) {

return args.join(split);

}

const str = `I love ${combine(";", "movie", "music", "sports")}`;

A. I love movie, music, sports

B. I love movie; music; sports

C. I love ;, movie, music, sports

D. 报错

4. 下面的代码输出结果是（  A  ）

const obj = {

a: 1,

b: 2,

sum: () => {

return this.a + this.b;

};

};

console.log(obj.sum());

A. 3

B. undefined

C. NaN

D. null

5. 下面的代码，输出结果正确的是（   ）

const getSum = (...args1) => (...args2) => args1.concat(args2).reduce((a, b) => a + b, 0);

const result = getSum(1, 2, 3)(4, 5, 6);

A. 0

B. NaN

C. 报错

D. 21

6. ES6中类的特点**不包含**（    ）

A. 不能当作普通函数调用

B. 类声明不会被提升

C. 类中的代码自动处于严格模式

D. 类中的实例方法可以被枚举

7. 如果A和B是两个类，A是B的父类，B是A的子类，则下面的描述正确的是（   C ）

A. 可以描述为：B是A

B. A和B拥有同一个原型

C. A的实例的原型链上有构造函数B

D. B的实例的原型链上有构造函数A

8. 下面的代码，输出结果正确的是（ C  ）

function method ({ a = 1, b = 2, c = 3 }) {

console.log(a, b, c);

}

method();

A. undefined  undefined  undefined

B. 1   2   3

C. 报错

D. null   null   null

9. 对于ES6中的符号Symbol，说法**错误**的是（   C  ）

A. 符号主要用于产生一个唯一的值

B. Symbol可以当作构造函数调用

C. 符号可以共享

D. ES6规定了一些知名符号，它们每个有着特殊的作用

10. ES6使用一种标准的模式Promise来处理异步，它将一个异步任务分为（  A   ）

A. unsettle 和 settled 两个阶段

B. pending 和 resolved 两种状态

C. resolved 和 rejected 两种状态

D. thenable 和 catchable 两个阶段

11. 如何理解Promise的构造函数参数（  A  ）

A. 描述如何进行后续处理

B. 描述settled阶段如何处理

C. 描述在unsettled阶段如何处理

D. 如何通向settled阶段不由该参数负责

12. 如何理解Promise对象的then方法（  C  ）

A. 通过then方法可以让Promise通向settled阶段

B. 通过then方法可以让Promise通向resolved状态

C. 通过then方法可以向Promise注册thenable或catchable处理函数

D. 通过then方法可以向状态传递数据

13. 下面哪种做法**不能**让Promise进入rejected状态（   D ）

A. 在Promise构造函数的参数方法中抛出错误

B. 在Promise构造函数的参数方法中调用reject()

C. 在Promise构造函数的参数方法中代码报错

D. 调用Promise的catch方法

14. 当你需要同时发送多个请求，并把所有响应的结果进行合并，你应该选择（  A  ）

A. Promise的串联

B. Promise.all

C. Promise.race

D. 使用async和await

15. 关于async和await说法正确的是（    ）

A. async修饰的方法内必须使用await

B. async修饰的方法必定返回Promise对象

C. await必须等待一个Promise对象

D. await语句之后的代码可能仍然是同步代码

16. Fetch Api相比于XHR的优势包括（    ）

A. 功能分割更加精细

B. 摒弃了ajax模式

C. 使用的是回调模式

D. 使用的是事件模式

17. 对于文件上传表单字段，说法正确的是（  D   ）

A. 表单字段名必须是file

B. 表单字段名需要和服务器进行约定

C. 表单字段内容是文件名

D. 表单格式必须是multipart/form-data

18. 下面的哪种对象**不是**可迭代对象（  D   ）

A. Object

B. Array

C. NodeList

D. String

19. 对于yield关键字理解正确的是（  D   ）

A. yield后面的表达式表示迭代的结果

B. yield会导致浏览器运行阻塞

C. yield语句的返回结果是后面表达式的结果

D. yield语句可以用到普通函数中

20. 关于生成器函数理解正确的是（    ）

A. 生成器函数调用后会运行到第一个yield位置停止

B. 生成器函数本质上就是迭代器

C. 生成器函数返回的是一个Promise对象

D. 生成器函数返回的一定是一个可迭代对象

21. 关于set和map集合的说法**正确**的是（  D  ）

A. set集合没有filter方法

B. map集合的值（value）是唯一的

C. map集合在迭代时，每次迭代得到的是一个普通对象

D. 用set构造函数初始化集合内容时，参数必须是一个数组

22. 关于Reflect的说法，正确的是（   ）

A. Reflect只能用于代理中

B. Reflect 出现后，过去的相关语法已被淘汰

C. Reflect使用api的形式暴露JS底层实现

D. Reflect.defineProperty和Object.defineProperty完全一样

23. 如果要从一个学生数组中找到姓名为“李华”的学生对象，最合适的API是（    ）

A. includes

B. indexOf

C. find

D. findIndex

24. 关于ArrayBuffer和类型化数组，说法正确的是（    ）

A. ArrayBuffer可以存储任何类型的数据

B. 类型化数组可以存储任何类型的数据

C. ArrayBuffer只能通过类型化数组操作数据

D. 类型化数组可以降低内存消耗

### 二、填空题（每空1分，共12空， 12分）

1. 过去，一个函数总是可以有两种使用方式：直接调用和使用关键字new，这模糊了函数的功能，ES6之后，我们可以通过               来判断使用者使用了何种方式调用函数。

2. 在使用解构时，我们可以通过符号   …   来修改解构到变量的名称

3. 将下面的代码补充完整

/\*\*

\* 请求指定的地址，得到服务器返回的json对象

\* @param {\*} url

\*/

const req = \_\_\_\_\_\_\_ => {

return await fetch(url).then(\_\_\_\_\_\_\_\_=> resp.json());

}

//请求students.json，输出得到的结果

req("students.json").\_\_\_\_\_\_\_\_\_(resp => console.log(resp))

4. ES6对函数做了大量改进，其中，          解决了参数默认值的问题，           解决了参数数量不固定的问题。

5. Promise可以从unsettled阶段推向settled阶段，调用resolve函数可以进入      resolve    状态，调用reject函数或        可以进入      reject   状态。

6. 一个可迭代对象，必定满足               ，即包含一个知名符号Symbol.\_\_\_\_\_\_\_\_\_\_。

### 三、简答题（共2题，每题10分， 20分）

1. 阐述Promise解决了什么问题，它如何解决的，async和await又解决了什么问题？

2. 阐述什么是迭代器、可迭代协议、可迭代对象、生成器

### 四、编程题（共2题，每题10分，20分）

1. 有一个字符串数组urls，里面存放了很多的请求地址，现在需要同时向这些地址发送ajax请求，并把每次请求的JSON结果按照响应的先后顺序依次保存到一个新数组results里面，当所有请求完成后，把新数组results输出

results数组中每一项的对象格式如下：

{

requestTime: 日期对象, //开始请求的时间

data: 对象, //服务器响应的结果

responseTime: 日期对象 //响应的时间

}

2. 编写一个函数createDomProxy，该函数返回一个DOM对象的代理，该代理可以将DOM对象以on开头的属性（即事件）变为Promise对象，可以让后续的开发者这样使用这个函数createDomProxy：

async function test() {

    const div = document.getElementById("test"); //得到某个div dom

    const divProxy = createDomProxy(div); //创建它的代理

    while (true) { //使用一个死循环，是为了不断的监听它被点击

        const e = await divProxy.onclick //等待它被点击，e是事件参数

        console.log("div被点击了", e) //被点击后运行的代码

    }

}

test();