MCQS TEST-2 Modern JavaScript Essentials

1.

What would be the output of the given JavaScript code in the console?

- ["name", "players", "play", "prototype"]
- ["name", "players", "play", "prototype", "constructor"]
- ● ["name", "players", "play"]  ⊗
- ["name", "players"]  ⊘

NEXT

```javascript
1  function Sport(name, players) {
2      this.name = name;
3      this.players = players;
4  }
5
6  Sport.prototype.play = function() {
7      return "Game started";
8  };
9
10 const sport1 = new Sport("Hockey", 11);
11
12 console.log(Object.getOwnPropertyNames(sport1));
```

2.

Considering the given code snippet, which of the following statements are correct when the given JavaScript code is executed?

- ☑ Hello! text is logged first in the console  ⊗
- ☑ fetch statement executes asynchronously  ⊘
- ☑ The alert statement executes synchronously  ⊘
- ☑ Good Morning is displayed first on the screen  ⊘

```javascript
1  const url = "https://apis.ccbp.in/jokes/random";
2
3  fetch(url)
4      .then((response) => {
5          console.log("Hello!");
6      });
7
8  alert("Good Morning");
```

3.

What would be the output of the given JavaScript code in the console?

- 1
  3
  2
- ● 3
  1  ⊗
  4
- 3
  1
  2
- 1
  3  ⊘
  4

```javascript
1  const promise1 = new Promise((resolve, reject) => {
2      console.log(1);
3      reject("error");
4  });
5
6  promise1.then(() => {
7      console.log(2);
8  });
9
10 promise1.catch(() => {
11     console.log(4);
12 });
13
14 console.log(3);
```

4.

What would be the output of the given JavaScript code in the console?

○ Cat prototype is shared across all its instances

◉ Both instances refer to the same prototype  ⊗

○ Both instances refer to the same prototype
Cat prototype is shared across all its instances  ⊘

○ None of the given options

**NEXT**

**PREVIOUS**

```javascript
1  function Cat(petName, breed) {
2      this.petName = petName;
3      this.breed = breed;
4  }
5
6  Cat.prototype.eat = function() {
7      return `${this.petName} is eating`;
8  };
9
10 const cat1 = new Cat("Kitty", "Birman");
11 const cat2 = new Cat("Stella", "Turkish Angora");
12
13 if (Object.getPrototypeOf(cat1) === Object.getPrototypeOf(cat2)) {
14     console.log("Both instances refer to the same prototype");
15 }
16
17 if (Object.getPrototypeOf(cat1) === Cat.prototype) {
18     console.log("Cat prototype is shared across all its instances");
19 }
```

5.

What would be the output of the given JavaScript code in the console?

○ First task
Second task
Third task
Ohh! Error

◉ First task
Second task  ⊗
Third task

○ Ohh! Error  ⊘

**NEXT**

**PREVIOUS**

```javascript
1  function task() {
2      return new Promise(function(rej, res) {
3          res();
4      });
5  }
6
7  let promise = task();
8
9  promise
10     .then(function() {
11         console.log("First task");
12     })
13     .then(function() {
14         console.log("Second task");
15     })
16     .then(function() {
17         console.log("Third task");
18     })
19     .catch(function() {
20         console.log("Ohh! Error");
21     });
```

**6.**

What would be the output of the given JavaScript code in the console?

- ○ 345 ✓
- ○ ReferenceError
- ● 245 ✗
- ○ TypeError

**NEXT**

```javascript
1  let num = 245;
2
3  function func(){
4      num = 345;
5  }
6
7  func();
8
9  console.log(num);
```

**7.**

What would be the output of the given JavaScript code in the console?

- ○ ReferenceError
- ○ None of the given options
- ○ 2-Stroke ✓
- ● undefined ✗

**DONE**

```javascript
1  const Engine = new Function("type, cylinders",
2  `this.type = type;
3   this.cylinders = cylinders;
4   this.start = function() {
5      console.log(this.type);
6   };`
7  );
8
9  const engine1 = new Engine("2-Stroke", 4);
10 engine1.start();
```

**8.**

Refer to the given code and fill in the blank space to log the shared *prototype* of the given array in the console.

- ● Array.prototype ✗
- ○ gymEquipment.constructor
- ○ Object.getPrototypeOf(gymEquipment) ✓
- ○ Object.getOwnPropertyNames(gymEquipment)

```javascript
1  const gymEquipment = ["Treadmill", "Dumbbells", "Gym bike", "Elliptical trainer"];
2
3  console.log(___);
```

9.

What would be the output of the given JavaScript code in the console?

- ○ None of the given options
- ○ ReferenceError ⊘
- ◉ 1
  ReferenceError ⊗
- ○ 1
  3

**DONE**

**PREVIOUS**

```javascript
1   const x = 14;
2
3   if (typeof(x) === "number") {
4     const y = 1;
5   }
6
7   console.log(y);
8
9   function foo() {
10    const z = 3;
11  }
12
13  console.log(z);
```

10.

What would be the output of the given JavaScript code in the console?

- ○ constructor()
- ◉ perfume1 { ... } ⊗
- ○ f Perfume(flavour, price) { ... } ⊘

**NEXT**

```javascript
1   function Perfume(flavour, price) {
2     this.flavour = flavour;
3     this.price = price;
4   }
5
6   const perfume1 = new Perfume("Tuberose", 500);
7
8   console.log(perfume1.constructor);
```

11.

What would be the output of the given JavaScript code in the console?

- ○ Metros at undefined metro station are running
- ○ ReferenceError ⊘
- ◉ Metros at Secunderabad metro station are running ⊗

**DONE**

```javascript
1   const metroStation1 = new MetroStation("Secunderabad", true);
2   metroStation1.runMetro();
3
4   const MetroStation = function(place, isOpen) {
5     this.place = place;
6     this.isOpen = isOpen;
7     this.runMetro = () => {
8       console.log(`Metros at ${this.place} metro station are running`);
9     };
10  };
```
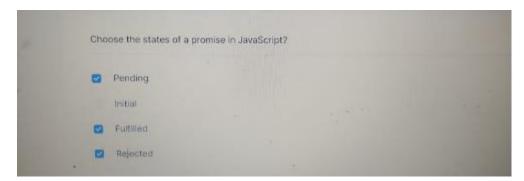
12.

What would be the output of the given JavaScript code in the console?

- true
- false

SUBMIT

SKIP

If you skip, you can attempt this question again after finishing all questions

```javascript
1  const numbers = new Array(1, 2, 3);
2  const alphabets = ["a", "b", "c"];
3
4  console.log(Object.getPrototypeOf(numbers) === Object.getPrototypeOf(alphabets));
```

13.

Choose the states of a promise in JavaScript?

- ☑ Pending
- ☐ Initial
- ☑ Fulfilled
- ☑ Rejected

14.

What would be the output of the given JavaScript code in the console?

- Person
  Person
  Person
- ● Teacher
  Person
  Person
- SyntaxError
- Teacher
  Teacher
  Teacher

```javascript
1   const alpha = "Person";
2
3-  if (alpha === "Person") {
4       const alpha = "Teacher";
5       console.log(alpha);
6   }
7
8-  if (true) {
9       console.log(alpha);
10  }
11
12  console.log(alpha);
```

15.

What would be the output of the given JavaScript code in the console?

- ● This tablet has 6GB RAM and 128GB of ROM
- This tablet has undefined RAM and undefined of ROM
- This tablet has RAM and of ROM
- ReferenceError

```javascript
1   let tablet1 = new Tablet("6GB", "128GB");
2   tablet1.describe();
3
4-  function Tablet(ram, rom) {
5       this.ram = ram;
6       this.rom = rom;
7-      this.describe = () => {
8           console.log(`This tablet has ${this.ram} RAM and ${this.rom} of ROM`);
9       };
10  }
```

**16.**

Considering the given code snippet, which of the following options are the *prototype* methods?

☐ name

☐ color

☑ describe

☑ create

**SUBMIT**

**SKIP**

If you skip, you can attempt this question again after finishing all questions

```
1  function Flower() {}
2
3 - Flower.prototype.create = function(name, color) {
4      this.name = name;
5      this.color = color;
6  };
7
8 - Flower.prototype.describe = function() {
9      console.log(`This is a ${this.color} ${this.name}`);
10 };
11
12 const flower1 = new Flower();
13 flower1.create("Rose", "Red");
14
```

**17.**

Which of the following options is correct to access the *prototype* of the Object Constructor?

○ Object.getOwnPropertyNames()

◉ Object.prototype

○ Object.getPrototypeOf()

**18.**

What would be the output of the given JavaScript code in the console?

○ jsonData.value

○ ReferenceError

◉ The robot says to the dead robot, Rust-in-peace

○ undefined

```
1  const url = "https://apis.ccbp.in/jokes/random";
2
3  fetch(url)
4  .then((response) => {   // Assume the success response to
5     return response.json();
6  })
7 - .then((jsonData) => {
8     console.log(jsonData.value);
9  });
```

**19.**

What would be the output of the given JavaScript code in the console?

○ SyntaxError

◉ ["color", "price"]

○ []

○ ["color", "price", "launch"]

**SUBMIT**

**SKIP**

If you skip, you can attempt this question again after finishing all questions

```
1  function Ferrari() {}
2
3 - Ferrari.prototype.launch = function(color, price) {
4      this.color = color;
5      this.price = price;
6  };
7
8  const ferrari1 = new Ferrari();
9  ferrari1.launch("Grey", "3.76 Cr");
10
11 console.log(Object.getOwnPropertyNames(ferrari1));
```

20.

What would be the output of the given JavaScript code in the console?

- SyntaxError
- undefined
- undefined is talking
- ● Coco is talking

**SUBMIT**

SKIP

If you skip, you can attempt this question again after finishing all questions

```javascript
1  function Parrot(petName) {
2    this.petName = petName;
3  }
4
5  Parrot.prototype.talk = function() {
6    console.log(`${this.petName} is talking`);
7  };
8
9  const parrot1 = new Parrot("Coco");
10
11 parrot1.talk();
```