

S.No	courses	commands
1	eetyyu	ghjkklll
2	rftvgb	tyghuj

JAVA SCRIPT NOTES

JAVA SCRIPT - ARRAY,OBJECT

```
var names = [  
  {  
    name: "Ramu",  
    age: 30,  
    loc: "Chennai",  
    friends: [  
      { name: "Guna", age: 40 },  
      { name: "Sendhil", age: 42 },  
    ],  
  },  
  {  
    name: "Babu",  
    age: 40,  
    loc: "Bangalore",  
    friends: [{ name: "Rajan", age: 38 }],  
  },  
];
```

op = 42

SECOND QUESTION

```
var names = [  
  {  
    name: "Ramu",  
    age: 30,  
    loc: "Chennai",  
    friends: [  
      {  
        name: "Guna",  
        age: 40,  
        loc: { city: "Chennai", zipcode: 600001 },  
      },  
      {  
        name: "Sendhil",
```

```

age: 42,
loc: { city: "Chennai", zipcode: 600003 },
},
],
},
{
name: "Babu",
age: 40,
loc: "Bangalore",
friends: [
{
name: "Rajan",
age: 38,
loc: { city: "Chennai", zipcode: 600002 },
},
],
},
];
console.log(names[1].friends[0].loc.zipcode)

```

OP = 600002

1. THIRD QUESTION:

v

```

ar names = [
{
id: 1,
name: "Leanne Graham",
username: "Bret",
email: "Sincere@april.biz",
address: {
street: "Kulas Light",
suite: "Apt. 556",
city: "Gwenborough",
zipcode: "92998-3874",
geo: {
lat: "-37.3159",
lng: "81.1496",
},
},
phone: "1-770-736-8031 x56442",
website: "hildegard.org",
company: {
name: "Romaguera-Crona",
catchPhrase: "Multi-layered client-server neural-net",
bs: "harness real-time e-markets",
},
},
{
id: 2,
name: "Ervin Howell",
username: "Antonette",
email: "Shanna@melissa.tv",
address: {
street: "Victor Plains",
suite: "Suite 879",
city: "Wisokyburgh",

```

```
zipcode: "90566-7771",
geo: {
  lat: "-43.9509",
  lng: "-34.4618",
},
},
phone: "010-692-6593 x09125",
website: "anastasia.net",
company: {
  name: "Deckow-Crist",
  catchPhrase: "Proactive didactic contingency",
  bs: "synergize scalable supply-chains",
},
},
{
  id: 3,
  name: "Samantha",
  username: "Samantha",
  email: "Nathan@yesenia.net",
  address: {
    street: "Douglas Extension",
    suite: "Suite 847",
    city: "McKenziehaven",
    zipcode: "59590-4157",
    geo: {
      lat: "-68.6102",
      lng: "-47.0653",
    },
  },
  phone: "1-463-123-4447",
  website: "ramiro.info",
  company: {
    name: "Romaguera-Jacobson",
    catchPhrase: "Face to face bifurcated interface",
    bs: "e-enable strategic applications",
  },
},
{
  id: 4,
  name: "Patricia Lebsack",
  username: "Karianne",
  email: "Julianne.OConner@kory.org",
  address: {
    street: "Hoeger Mall",
    suite: "Apt. 692",
    city: "South Elvis",
    zipcode: "53919-4257",
    geo: {
      lat: "29.4572",
      lng: "-164.2990",
    },
  },
  phone: "493-170-9623 x156",
  website: "kale.biz",
  company: {
    name: "Robel-Corkery",
    catchPhrase: "Multi-tiered zero tolerance productivity",
    bs: "transition cutting-edge web services",
  },
},
{
  id: 5,
  name: "Chelsey Dietrich",
  username: "Kamren",
  email: "Lucio_Hettinger@annie.ca",
  address: {
    street: "Skiles Walks",
    suite: "Suite 351",
    city: "Roscoeview",
    zipcode: "33263",
```

```
geo: {
lat: "-31.8129",
lng: "62.5342",
},
},
phone: "(254)954-1289",
website: "demarco.info",
company: {
name: "Keebler LLC",
catchPhrase: "User-centric fault-tolerant solution",
bs: "revolutionize end-to-end systems",
},
},
{
id: 6,
name: "Mrs. Dennis Schulist",
username: "Leopoldo Corkery",
email: "Karley_Dach@jasper.info",
address: {
street: "Norberto Crossing",
suite: "Apt. 950",
city: "South Christy",
zipcode: "23505-1337",
geo: {
lat: "-71.4197",
lng: "71.7478",
},
},
phone: "1-477-935-8478 x6430",
website: "ola.org",
company: {
name: "Considine-Lockman",
catchPhrase: "Synchronised bottom-line interface",
bs: "e-enable innovative applications",
},
},
{
id: 7,
name: "Kurtis Weissnat",
username: "Elwyn.Skiles",
email: "Telly.Hoeger@billy.biz",
address: {
street: "Rex Trail",
suite: "Suite 280",
city: "Howemouth",
zipcode: "58804-1099",
geo: {
lat: "24.8918",
lng: "21.8984",
},
},
phone: "210.067.6132",
website: "elvis.io",
company: {
name: "Johns Group",
catchPhrase: "Configurable multimedia task-force",
bs: "generate enterprise e-tailers",
},
},
{
id: 8,
name: "Nicholas Runolfsson",
username: "Maxime Nienow",
email: "Sherwood@rosamond.me",
address: {
street: "Ellsworth Summit",
suite: "Suite 729",
city: "Aliyaview",
zipcode: "45169",
geo: {
```

```

lat: "-14.3990",
lng: "-120.7677",
},
},
phone: "586.493.6943 x140",
website: "jacynthe.com",
company: {
name: "Abernathy Group",
catchPhrase: "Implemented secondary concept",
bs: "e-enable extensible e-tailers",
},
},
{
id: 9,
name: "Glenna Reichert",
username: "Delphine",
email: "Chaime_McDermott@dana.io",
address: {
street: "Dayna Park",
suite: "Suite 449",
city: "Bartholomebury",
zipcode: "76495-3109",
geo: {
lat: "24.6463",
lng: "-168.8889",
},
},
phone: "(775)976-6794 x41206",
website: "conrad.com",
company: {
name: "Yost and Sons",
catchPhrase: "Switchable contextually-based project",
bs: "aggregate real-time technologies",
},
},
{
id: 10,
name: "Clementina DuBuque",
username: "Moriah.Stanton",
email: "Rey.Padberg@karina.biz",
address: {
street: "Kattie Turnpike",
suite: "Suite 198",
city: "Lebsackbury",
zipcode: "31428-2261",
geo: {
lat: "-38.2386",
lng: "57.2232",
},
},
phone: "024-648-3804",
website: "ambrose.net",
company: {
name: "Hoeger LLC",
catchPhrase: "Centralized empowering task-force",
bs: "target end-to-end models",
},
},
];
console.log(names[2].company.catchPhrase)

```

----->OP = Face to face bifurcated interface # SAMANTHA

ARRAY - FUNCTIONS

- reverse - reverse the given array
- join - add the element or special operators to join in between the array elements acts as an separator .
converts to string

example

```
var a=['a','s','d','f','g']
```

```
console.log(a.reverse())-----> op = [g,f,d,s,a]
```

```
console.log(a.join("-")) -----> op = a-s-d-f-g
```

- split - used to split and convert to an array

example :

```
var a = "sreem"
```

```
console.log(a.split("")) ---['s','r','e','e','m']
```

```
console.log(a.split("e")) ---["sr"," ","m"]
```

```
console.log(a.split(" ")) ---->[ "sreema" ]
```

REVERSE A STRING :

```
function strinreverse(a) {  
    var b = a.split("")  
    var c = b.reverse()  
    return c.join("")  
}  
var a = "sreema"  
var b = strinreverse(a)  
if (a == b) //or a === b  
{  
    console.log("reversed string =", b)  
    console.log("palindrome")  
}  
else {  
    console.log("reversed string =", b)  
    console.log("not an palindrome")  
}
```

```
----> reversed string = ameers
```

```
not an palindrome
```

OR (method chaining)

```
return str.split("").reverse().join("")----->ameers
```

- we can compare both number and string

```
a=1
b="1"
if(a==b){
  console.log("equal")
}
else{
  console.log("not equal")
}
```

- == - compare only the value so return equal
- === compare both value and string so not equal to
- if two non primitives(array,objects) are compared data ,datatype and the address is also checked so not equal