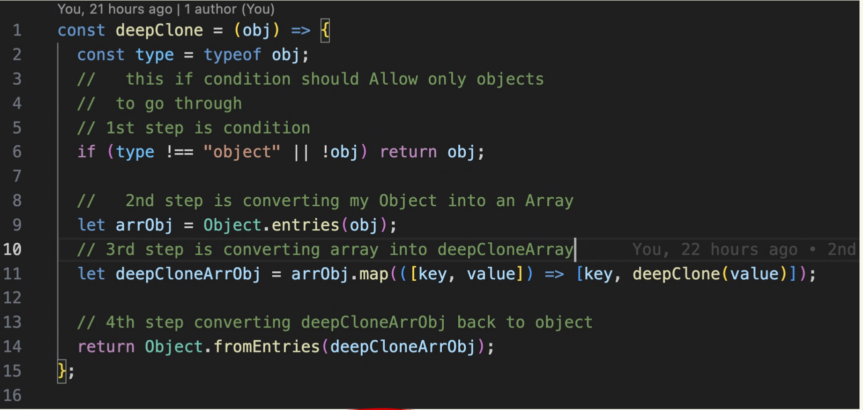


```js
const deepClone = (obj) => {
  const type = typeof obj;
  //   this if condition should Allow only objects
  //  to go through
  // 1st step is condition
  if (type !== "object" || !obj) return obj;

  //   2nd step is converting my Object into an Array
  let arrObj = Object.entries(obj);
  // 3rd step is converting array into deepCloneArray
  let deepCloneArrObj = arrObj.map(([key, value]) => [key, deepClone(value)]);

  // 4th step converting deepCloneArrObj back to object
  return Object.fromEntries(deepCloneArrObj);
};
```

let obj= { a: 10 ,
b: { c: 20 } }

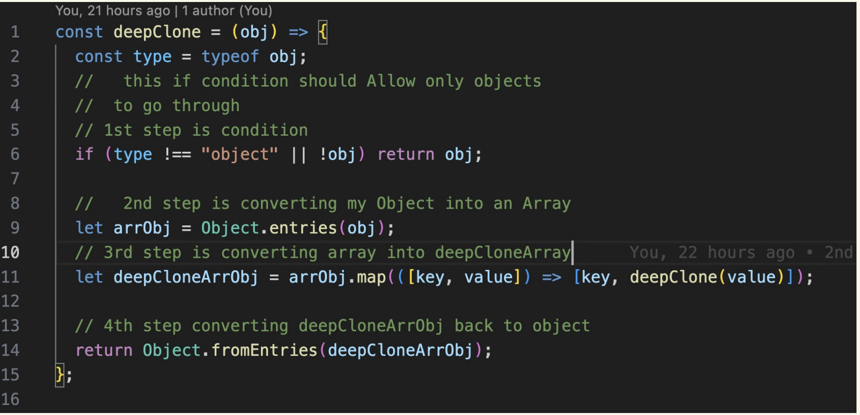
① [["a", 10], ["b", {c: 20}]] ⇒ [["a", deepClone(10)], ["b", deepClone({c:20})]]

* ["b", {c:20}] → ["b", deepClone({c:20})

③ {c:20} → [["c", 20]] ⇒ deepClone(20);

⇒ [["c", 20]]

* {c: 20} ⟵



You, 21 hours ago | 1 author (You)

```
const deepClone = (obj) => {
  const type = typeof obj;
  //   this if condition should Allow only objects
  //  to go through
  // 1st step is condition
  if (type !== "object" || !obj) return obj;

  //   2nd step is converting my Object into an Array
  let arrObj = Object.entries(obj);
  // 3rd step is converting array into deepCloneArray      You, 22 hours ago • 2nd
  let deepCloneArrObj = arrObj.map(([key, value]) => [key, deepClone(value)]);

  // 4th step converting deepCloneArrObj back to object
  return Object.fromEntries(deepCloneArrObj);
};
```

{ a: 10

b: {c: 20} }

⇒ [{"a", 10], ["b", {c: 20}]]

⇒ [["a", 10], ["b", deepClone({c:20})]]

⇒ [["c, 20]] ⟶ ["c", deepClone(20)] ⇒ {c: 20}

⇒ [["b", {c: 20}]]

↓

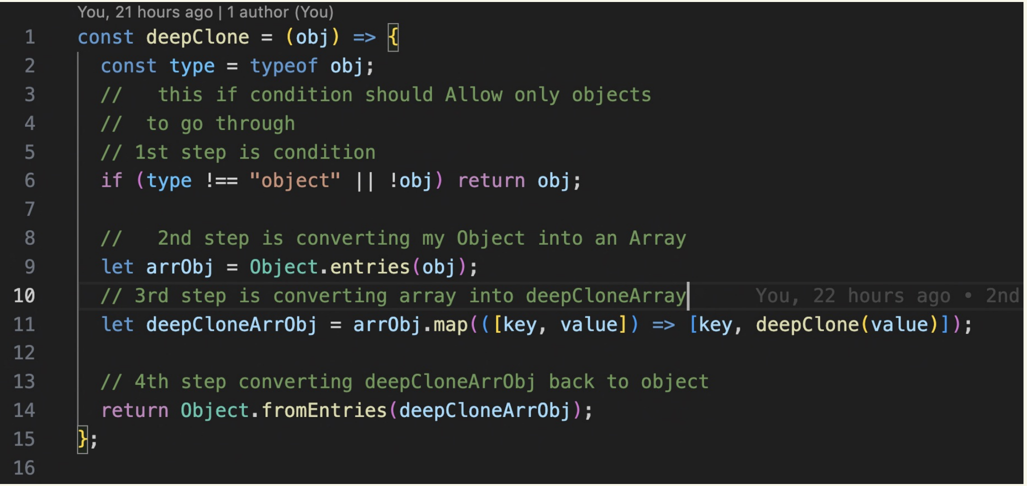
⇒ [["b", deepClone{c: 20}]]

[["c", 20]] ⇒ [["c", deepClone(20)]]

↓

[["c", 20]]

Made with Goodnotes



You, 21 hours ago | 1 author (You)

```javascript
const deepClone = (obj) => {
  const type = typeof obj;
  //   this if condition should Allow only objects
  //  to go through
  // 1st step is condition
  if (type !== "object" || !obj) return obj;

  //   2nd step is converting my Object into an Array
  let arrObj = Object.entries(obj);
  // 3rd step is converting array into deepCloneArray      You, 22 hours ago • 2nd
  let deepCloneArrObj = arrObj.map(([key, value]) => [key, deepClone(value)]);

  // 4th step converting deepCloneArrObj back to object
  return Object.fromEntries(deepCloneArrObj);
};
```

→ [{a:10}, {b:10
c:{2:100}}]

→ [deep({a:10}), deepClone({b:10
c:{2:100}})]

* [obj1, obj2, obj3] → type of arr

objec

⇒

⇒ if (it's an object or not)

⇒ [deepClone(obj1), deepClone(obj2), deepClone(obj3))

⇒ logic for Objects