

# Ignore certain properties in an interface while creating an object in typescript?



1



I have interface defined with some properties. I want to create an object from this interface, however while creating I don't want to fully initialize all the properties mentioned in the interface. I just want to initialize few. How can I achieve that? Thank you.

```
export interface Campaign {  
  id: string;  
  name: string;  
  createdOn: string;  
  lastUpdated: string;  
  createdBy: User;  
  type: CampaignType;  
  status: CampaignStatus;  
  startDate: string;  
  endDate: string;  
  budget: number;  
  description: string;  
  account: Account;  
}
```

i want to create an array of campaign objects. this is what i am trying to do .

```
let campaigns: Campaign[] = [  
  {  
    id: "1",  
    name: "test campaign"  
  }  
];
```

however i get the following error .

Type '{ id: string; name: string; }' is missing the following properties from type 'Campaign': createdOn, lastUpdated, createdBy, type, and 6 more.ts(2740)

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH





539 7 22



21 7

But why? If an object implements interface `x` then I'm expecting it to have all properties and methods defined by interface `x`. Why else would I use an interface at all? – [Andreas](#) Apr 16 at 11:16

this is just for testing . – [prasanth](#) Apr 16 at 11:23

### 3 Answers



you can use Partial -

1

```
let campaigns: Partial<Campaign>[] = [
  { id:"1",
    name: "test campaign"
  }
];
```



answered Apr 16 at 11:16

[Naor Talmor](#)

43 7

this is what i was looking for thank you so much – [prasanth](#) Apr 16 at 11:37



if you make the optional properties nullable it should work.

1

example:



```
type User = {
  firstName: string;
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

[Email Sign Up](#)

OR SIGN IN WITH

[Google](#)[Facebook](#)

```
// ... or we can explicitly assign the value undefined
let jane: User = { firstName: "Jane", lastName: undefined };

// ... or we can not define the property at all
let jake: User = { firstName: "Jake" };
```

answered Apr 16 at 11:13

[Zdravko Danev](#)

10.6k 1 26 44



You could mark the properties that you don't want to set a value on creation as optional by doing `{ propName?: propType }`

1

or if want to mark all the properties to be optional, you could use `Partial`



```
Partial<{
  prop1: string,
  prop2: number,
}>
```

answered Apr 16 at 11:18

[Rannie Aguilar Peralta](#)

632 3 12

Join **Stack Overflow** to learn, share knowledge, and build your career.

[Email Sign Up](#)

OR SIGN IN WITH

[Google](#)