4/7/25, 12:07 PM Generics

Generics

Making the component resuable is called generic

```
133
      console.log(emp1.register())
134
135

    Generics

136
      function getArray(items: any[]):any[] {
137
           return new Array().concat(items);
138
139
140
141
      let numArray = getArray([1,2,3,4,5])
142
      let strArray = getArray(['a', 'b'])
143
144
      numArray.push('Iyyappan')
      strArray.push(1)
145
```

It will accept the type dynamically and produce the result according to it

4/7/25, 12:07 PM Generics

```
console.log(emp1.register())
135
136
      //Generics
      function getAnray<T>(items: T[]):T[] {
137
          return new Array().concat(items);
138
      }
139
140
      let numArray = getArray<number>([1,2,3,4,5])
141
      let strArray = getArray<string>(['a', 'b'])
142
143
      numArray.push(1)
144
      strArray.push('sdff')
145
```

For this we have to use the <T> the T can be anything it will accept the input dynamically and provide the results